



UNIVERSIDAD DE OVIEDO

ESCUELA DE INGENIERÍA INFORMÁTICA

MIGRACIÓN Y REINGENIERÍA DE UNA
APLICACIÓN WEB DIRIGIDA AL ESTUDIO DE
LA ACCESIBILIDAD

TRABAJO FIN DE MÁSTER

DIRECTOR: AQUILINO ADOLFO JUAN FUENTE

CODIRECTORA: MARÍA DEL CARMEN SUÁREZ TORRENTE

AUTOR: CARMEN PEÑALVER SAN CRISTÓBAL

Agradecimientos

Me gustaría agradecer y dedicar este documento a todas las personas que me han ayudado y han hecho posible, junto con mi dedicación y tesón, a finalizar este proyecto y, por consecuente, mi máster.

En primer lugar, agradecer a mi familia: mis padres, hermanas, hermano y Paquita, la paciencia con la que me han tratado. Ellos han soportado mi malhumorado y cambiante humor, mis lloros, mis “no se hacer esto” o mis “pues no lo hago” y, a pesar de todo, han estado ahí ayudándome y animándome para poder concluir estos estudios.

De igual forma agradecer a mi amiga y compañera de estudios, Liliana Villar Iglesias, por pasar, con su eterna paciencia durante estos dos años, días enteros acompañándome, enseñándome y animándome a que si uno quiere todo es posible.

Gracias a mi tutor, Aquilino A. Juan Fuentes, por darme la oportunidad de volver a realizar un proyecto con él, así como enseñarme a que un buen profesional del sector, no solamente se dedica a crear un programa desde cero, sino que también es necesario saber trabajar sobre un código desarrollador por otro compañero.

Finalmente, agradecer a todas las personas y compañeros que durante estos años han ido trabajando sobre el proyecto Atenea

Resumen

El presente proyecto está dirigido a retomar la implementación de un sistema ya existente, denominado Atenea. Esta es una aplicación dirigida al estudio cuantitativo de la web, atendiendo a aspectos tanto de accesibilidad como usabilidad.

Actualmente, ya cuenta con una cierta trayectoria de utilidad, sobre todo en el ámbito de la universidad de Oviedo y, durante este periodo de tiempo, se han podido detectar ciertas deficiencias o problemas que se desean solventar. Además, debido al avance de las nuevas tecnologías y las nuevas exigencias de la sociedad, es necesario actualizar el sistema y añadirle nueva funcionalidad, por haberse quedado ligeramente anticuado.

Cabe descartar que, la mayor y principal deficiencia que acusa retomar el desarrollo de esta aplicación es su dependencia con los sistemas operativos Windows, lo que impide ejecutar Atenea en servidores con un sistema operativo distinto al citado.

Por todas estas razones, es necesario iniciar un proceso tanto de migración como de reingeniería, con el fin de solventar las deficiencias que presente Atenea, así como mejorar la funcionalidad existente e incluso añadir nuevas funciones. Y de esta forma poder conseguir un sistema Atenea mejorado, actualizado y sin fisuras.

Palabras Clave

Migración, Reingeniería, Accesibilidad, Usabilidad, Métrica, Crawler, Análisis

Abstract

This project is aimed at retaking the implementation of an existing system, called Athena. This is an application aimed at the quantitative study of the web, addressing aspects of both accessibility and usability.

Currently, it already has a certain track record of usefulness, especially in the field of the University of Oviedo and, during this period of time, it has been possible to detect certain deficiencies or problems that it is desired to solve. In addition, due to the advancement of new technologies and the new demands of society, it is necessary to update the system and add new functionality, as it has been slightly outdated.

It is discarded that, the biggest and main deficiency that accuses resuming the development of this application is its dependence on Windows operating systems, which prevents Athena from running on servers with an operating system different from that mentioned.

For all these reasons, it is necessary to start a process of both migration and reengineering, in order to solve the deficient Athena present, as well as improve existing functionality and even add new functions. And in this way to get an improved Athena system, updated and seamless.

Keywords

Migration, Re-engineering, Accessibility, Usability, Metric, Crawler, Analysis

Índice General

CAPÍTULO 1. INTRODUCCIÓN.....	25
1.1 JUSTIFICACIÓN DEL PROYECTO	25
1.2 OBJETIVOS DEL PROYECTO	26
1.3 ESTUDIO DE LA SITUACIÓN ACTUAL	27
1.3.1 <i>Evaluación de Alternativas</i>	27
CAPÍTULO 2. ASPECTOS TEÓRICOS.....	29
2.1 ACCESIBILIDAD EN LA WEB	29
2.1.1 <i>Pautas de accesibilidad en la web</i>	29
2.1.2 <i>Evaluación de la accesibilidad en la web</i>	30
2.2 USABILIDAD EN LA WEB.....	33
2.2.1 <i>Evaluación de la usabilidad en la web</i>	34
2.3 CRAWLER O ARAÑA WEB	35
2.3.1 <i>Funcionamiento</i>	35
CAPÍTULO 3. PLANIFICACIÓN DEL PROYECTO Y PRESUPUESTOS	37
3.1 PLANIFICACIÓN.....	37
3.1.1 <i>Estudio de la situación actual</i>	38
3.1.2 <i>Análisis del proyecto</i>	39
3.1.3 <i>Implementación</i>	39
3.1.4 <i>Implantación en producción</i>	40
3.1.5 <i>Cierre del proyecto</i>	41
3.1.6 <i>Diagrama de Gantt</i>	43
3.2 PRESUPUESTO	45
3.2.1 <i>Costes indirectos</i>	45
3.2.2 <i>Costes de desarrollo</i>	45
3.2.3 <i>Resumen del presupuesto</i>	46
3.2.4 <i>Presupuesto indicativo para el cliente</i>	47
CAPÍTULO 4. REINGENIERÍA DEL SISTEMA.....	48
4.1 CONTEXTO DEL PROYECTO	49
4.2 CREACIÓN DEL PROYECTO	50
4.3 MODIFICACIONES INTERNAS.....	52
4.3.1 <i>Paquete main</i>	52
4.3.2 <i>Paquete presentation</i>	52
4.3.3 <i>Paquete model</i>	57
<i>Paquete dao</i>	61
4.3.4 <i>Paquete utilities</i>	63
4.3.5 <i>Paquete comands</i>	64
4.3.6 <i>Paquete services</i>	65
4.4 MODIFICACIONES EXTERNAS	67
4.4.1 <i>Archivos JAR</i>	67
4.4.2 <i>Web para configurar los crawlers</i>	68
4.5 ADICIONES AL PROYECTO.....	71
4.5.1 <i>Internacionalización</i>	71
4.5.2 <i>Aspecto de listas y “cargando”</i>	71

4.5.3	<i>Servicio de notificaciones</i>	72
4.5.4	<i>Control de sesiones</i>	73
4.6	LISTADO COMPLETO DE APORTACIONES.....	76
CAPÍTULO 5.	ANÁLISIS	77
5.1	DEFINICIÓN DEL SISTEMA.....	77
5.1.1	<i>Determinación del Alcance del Sistema</i>	77
5.2	REQUISITOS DEL SISTEMA.....	78
5.2.1	<i>Obtención de los Requisitos del Sistema</i>	78
5.2.2	<i>Identificación de Actores del Sistema</i>	92
5.2.3	<i>Especificación de Casos de Uso</i>	93
5.3	IDENTIFICACIÓN DE LOS SUBSISTEMAS EN LA FASE DE ANÁLISIS.....	99
5.3.1	<i>Descripción de los Subsistemas</i>	99
5.3.2	<i>Descripción de los Interfaces entre Subsistemas</i>	100
5.4	DIAGRAMA DE CLASES PRELIMINAR DEL ANÁLISIS	100
5.4.1	<i>Diagrama de Clases</i>	100
5.4.2	<i>Descripción de las Clases</i>	101
5.5	ANÁLISIS DE CASOS DE USO Y ESCENARIOS	105
5.5.1	<i>Caso de uso análisis público</i>	105
5.5.2	<i>Caso de uso consultar información pública</i>	106
5.5.3	<i>Caso de uso iniciar sesión en el sistema</i>	106
5.5.4	<i>Caso de uso modificar idioma</i>	107
5.5.5	<i>Caso de uso actualizar configuración básica</i>	107
5.5.6	<i>Caso de uso añadir una métrica</i>	108
5.5.7	<i>Caso de uso añadir un crawler</i>	108
5.5.8	<i>Caso de uso gestión de usuarios</i>	109
5.5.9	<i>Caso de uso configurar notificaciones</i>	110
5.5.10	<i>Caso de uso gestión de proyectos</i>	110
5.5.11	<i>Caso de uso gestión de consultas</i>	113
5.5.12	<i>Caso de uso consultar análisis en curso</i>	116
5.5.13	<i>Caso de uso gestionar análisis</i>	117
5.5.14	<i>Caso de uso consultar resultados</i>	119
5.5.15	<i>Caso de uso consultar errores</i>	119
5.5.16	<i>Caso de uso consultar ayuda</i>	120
5.5.17	<i>Caso de uso cerrar sesión</i>	120
5.6	ANÁLISIS DE INTERFACES DE USUARIO	121
5.6.1	<i>Descripción de la Interfaz</i>	121
5.6.2	<i>Descripción del Comportamiento de la Interfaz</i>	123
5.6.3	<i>Diagrama de Navegabilidad</i>	124
5.7	ESPECIFICACIÓN DEL PLAN DE PRUEBAS	125
CAPÍTULO 6.	DISEÑO DEL SISTEMA	133
6.1	ARQUITECTURA DEL SISTEMA.....	133
6.1.1	<i>Diagramas de Paquetes</i>	133
6.1.2	<i>Diagramas de Componentes</i>	136
6.1.3	<i>Diagramas de Despliegue</i>	137
6.2	DISEÑO DE CLASES	139
6.2.1	<i>Diagrama de clases del paquete model</i>	139
6.2.2	<i>Diagrama de clases del paquete utilities</i>	140
6.2.3	<i>Diagrama de clases del paquete main</i>	141
6.2.4	<i>Diagrama de clases del paquete engine</i>	142

6.2.5	<i>Diagrama de clases del paquete engine_threads</i>	143
6.2.6	<i>Diagrama de clases de los paquetes metric y crawler</i>	144
6.2.7	<i>Diagrama de clases del paquete services</i>	145
6.2.8	<i>Diagrama de clases del paquete dao</i>	146
6.2.9	<i>Diagrama de clases del paquete presentation</i>	148
6.3	DIAGRAMAS DE INTERACCIÓN Y ESTADOS	150
6.3.1	<i>Zona pública: Analizar un sitio web</i>	150
6.3.2	<i>Zona privada: Analizar un único sitio web</i>	152
6.3.3	<i>Zona privada: Analizar un conjunto de URLs</i>	154
6.3.4	<i>Crear un nuevo proyecto</i>	156
6.4	DIAGRAMAS DE ACTIVIDADES.....	158
6.5	DISEÑO DE LA BASE DE DATOS	159
6.5.1	<i>Descripción del SGBD Usado</i>	159
6.5.2	<i>Integración del SGBD en Nuestro Sistema</i>	159
6.5.3	<i>Diagrama E-R</i>	160
6.6	DISEÑO DE LA INTERFAZ	161
6.6.1	<i>Zona pública</i>	161
6.6.2	<i>Zona privada</i>	164
6.7	ESPECIFICACIÓN TÉCNICA DEL PLAN DE PRUEBAS	172
6.7.1	<i>Pruebas de Integración y del Sistema</i>	172
6.7.2	<i>Pruebas de Usabilidad y Accesibilidad</i>	189
6.7.3	<i>Pruebas de Rendimiento</i>	191
CAPÍTULO 7. IMPLEMENTACIÓN DEL SISTEMA		193
7.1	ESTÁNDARES Y NORMAS SEGUIDOS.....	193
7.2	PATRONES DE DISEÑO.....	193
7.2.1	<i>Patrón Command</i>	194
7.2.2	<i>Patrón Singleton</i>	195
7.2.3	<i>Patrón Facade</i>	196
7.2.4	<i>Patrón Abstract Factory</i>	197
7.2.5	<i>Patrón DAO</i>	198
7.2.6	<i>Patrón Factory Method</i>	199
7.2.7	<i>Patrón Observer</i>	201
7.2.8	<i>Patrón MVC</i>	202
7.3	Lenguajes de Programación	204
7.3.1	<i>Java</i>	204
7.3.2	<i>JPQL y SQL</i>	204
7.3.3	<i>JavaScript y jQuery</i>	204
7.4	HERRAMIENTAS Y PROGRAMAS USADOS PARA EL DESARROLLO.....	205
7.4.1	<i>Herramientas de ofimática</i>	205
7.4.2	<i>Herramienta CASE</i>	205
7.4.3	<i>Herramientas para el desarrollo del sistema</i>	205
7.5	CREACIÓN DEL SISTEMA	207
7.5.1	<i>Problemas Encontrados</i>	207
7.5.2	<i>Descripción Detallada de las Clases</i>	207
CAPÍTULO 8. DESARROLLO DE LAS PRUEBAS		212
8.1	PRUEBAS DE INTEGRACIÓN Y DEL SISTEMA	212
8.2	PRUEBAS DE USABILIDAD Y ACCESIBILIDAD.....	233
8.2.1	<i>Identificación del sitio web</i>	233
8.2.2	<i>Usabilidad</i>	233

8.2.3	Accesibilidad	241
8.2.4	Funcionalidad.....	245
8.2.5	Contenidos	246
8.2.6	Confiabilidad	247
8.3	PRUEBAS DE RENDIMIENTO	248
8.3.1	Comparativa de rendimiento: Zona pública.....	248
8.3.2	Comparativa de rendimiento: Zona privada	249
CAPÍTULO 9. ESTUDIO DE LA ACCESIBILIDAD Y USABILIDAD DE LAS UNIVERSIDADES ESPAÑOLAS UTILIZANDO ATENEA		250
CAPÍTULO 10. MANUALES DEL SISTEMA		263
10.1	MANUAL DE INSTALACIÓN	263
10.1.1	Instalar Tomcat.....	266
10.1.2	Instalar sistema de gestión de bases de datos	269
10.1.3	Instalar Atenea	270
10.2	MANUAL DE EJECUCIÓN	271
10.3	MANUAL DE USUARIO.....	272
10.3.1	Zona pública	272
10.3.2	Zona privada.....	275
10.4	MANUAL DEL PROGRAMADOR	287
10.4.1	Manual para añadir nuevas métricas.....	287
10.4.2	Manual para añadir nuevos crawlers	290
10.4.3	Configuración de un crawler.....	292
CAPÍTULO 11. CONCLUSIONES Y AMPLIACIONES		293
11.1	CONCLUSIONES.....	293
11.2	AMPLIACIONES	294
11.2.1	Crawlers con interfaz de usuario	294
11.2.2	Reutilización de hilos	294
11.2.3	Prioridades entre análisis en curso	294
11.2.4	Opción para analizar un sitio web completo	294
11.2.5	Creación de una app de apoyo	294
11.2.6	Gestión de usuarios avanzada.....	295
CAPÍTULO 12. BIBLIOGRAFÍA.....		297
12.1	LIBROS Y ARTÍCULOS	297
12.2	REFERENCIAS EN INTERNET	298
12.3	OTROS TRABAJOS FIN DE MÁSTER	299
CAPÍTULO 13. APÉNDICES		300
13.1	GLOSARIO Y DICCIONARIO DE DATOS	300

Índice de Figuras

Imagen 1. Funcionamiento básico de un crawler	36
Imagen 2. Estructura general de la planificación	37
Imagen 3. Estructura de la fase de estudio	38
Imagen 4. Estructura de la fase de análisis	39
Imagen 5. Estructura de la fase de implementación	40
Imagen 6. Estructura de la fase de implantación	40
Imagen 7. Estructura de la fase de cierre del proyecto	41
Imagen 8. Diagrama de Gantt	43
Imagen 9. Diagrama de paquetes Atenea 3.0	49
Imagen 10. Popularidad frameworks web	50
Imagen 11. Asistente Spring Boot	51
Imagen 12. Patrón MVC	52
Imagen 13. Paquete presentation Atenea 3.0	53
Imagen 14. Paquete presentation Atenea 4.0	54
Imagen 15. Diagrama de clases comparativa Struts 2 y Spring	56
Imagen 16. Modelo de dominio Atenea 3.0	58
Imagen 17. Modelo de dominio Atenea 4.0	59
Imagen 18. Diagrama de paquete DAO	62
Imagen 19. Persistence manager	63
Imagen 20. Diagrama de paquete comands	64
Imagen 21. Dependencia entre Atenea y JARs	67
Imagen 22. Diagrama web complementaria configuración de crawler	69
Imagen 23. Diagrama de paquetes CrawlerConfig	70
Imagen 24. Ciclo Spring Security	73
Imagen 25. Diagrama Spring Security	74
Imagen 26. Estructura, información y formato al exportar un proyecto	83
Imagen 27. Estructura, información y formato para importar un proyecto	83
Imagen 28. Clasificación de los requisitos no funcionales según Ian Sommerville	90
Imagen 29. Casos de uso de un usuario anónimo	93
Imagen 30. Casos de uso de un usuario registrado	95
Imagen 31. Diagrama de clases preliminar	100
Imagen 32. Diagrama de robustez para análisis público	105
Imagen 33. Diagrama de robustez para inicio de sesión	106
Imagen 34. Diagrama de robustez abrir un proyecto	111
Imagen 35. Diagrama de robustez ejecutar consulta	113
Imagen 36. Diagrama de robustez análisis URL	117
Imagen 37. Prototipo de interfaz para configurar notificaciones	121
Imagen 38. Prototipo de interfaz para consultar errores	122
Imagen 39. Prototipo de interfaz para los listados	122
Imagen 40. Diagrama de paquetes	133
Imagen 41. Diagrama de componentes	136
Imagen 42. Diagrama de despliegue	137
Imagen 43. Diagrama de clases del paquete model	139
Imagen 44. Diagrama de clases del paquete utilities	140
Imagen 45. Diagrama de clases del paquete main	141
Imagen 46. Diagrama de clases del paquete engine	142

Imagen 47. Diagrama de clases del paquete engine_threads.....	143
Imagen 48. Diagrama de clases de los paquetes metric y crawler.....	144
Imagen 49. Diagrama de clases del paquete services.....	145
Imagen 50. Diagrama de clases de los paquetes dao y comand.....	146
Imagen 51. Diagrama de clases del paquete daos.....	147
Imagen 52. Arquitectura de Spring MVC.....	148
Imagen 53. Diagrama de clases del paquete presentation.....	148
Imagen 54. Diagrama de clases de la acción añadir consulta.....	149
Imagen 55. Diagrama de interacción para un análisis público.....	150
Imagen 56. Diagrama de interacción para analizar una única URL.....	152
Imagen 57. Diagrama de interacción para analizar un conjunto de URLs.....	154
Imagen 58. Diagrama de interacción para crear un proyecto.....	156
Imagen 59. Diagrama de actividad.....	158
Imagen 60. Diagrama Entidad-Relación.....	160
Imagen 61. Página principal zona pública.....	161
Imagen 62. Cargando análisis zona pública.....	162
Imagen 63. Página de resultados zona pública.....	162
Imagen 64. Página de inicio de sesión.....	163
Imagen 65. Página de ayuda.....	163
Imagen 66. Pantalla principal zona privada.....	164
Imagen 67. Pantalla nuevo proyecto.....	165
Imagen 68. Pantalla importar proyecto.....	165
Imagen 69. Pantalla abrir proyectos.....	166
Imagen 70. Pantalla añadir consulta.....	167
Imagen 71. Pantalla abrir consulta.....	168
Imagen 72. Pantalla análisis actual.....	168
Imagen 73. Pantalla realizar análisis desde fichero.....	169
Imagen 74. Pantalla de resultados.....	169
Imagen 75. Pantalla de errores.....	170
Imagen 76. Pantalla de configuración de notificaciones.....	170
Imagen 77. Pantalla de verificación.....	171
Imagen 78. Estructura del patrón Command.....	194
Imagen 79. Estructura del patrón Singleton.....	195
Imagen 80. Estructura del patrón Facade.....	196
Imagen 81. Estructura del patrón Abstract Factory.....	198
Imagen 82. Estructura del patrón DAO.....	199
Imagen 83. Estructura del patrón Factory Method.....	200
Imagen 84. Estructura del patrón observer.....	201
Imagen 85. Estructura patrón MVC.....	202
Imagen 86. Menú privado de Atenea.....	233
Imagen 87. Pie de página Atenea.....	234
Imagen 88. Calidad de etiquetado.....	234
Imagen 89. Página principal pública.....	235
Imagen 90. Atenea sin imágenes.....	235
Imagen 91. Navegación consistente.....	236
Imagen 92. Especificación JoBo.....	236
Imagen 93. Ayuda en la zona privada.....	237
Imagen 94. Información de contacto.....	237
Imagen 95. Atenea en español.....	238
Imagen 96. Descarga de contenidos.....	238
Imagen 97. Test adaptabilidad I.....	239

Imagen 98. Test de adaptabilidad II	239
Imagen 99. Sección de proyectos	240
Imagen 100. Sección de consultas	240
Imagen 101. Aumento del tamaño de letra	241
Imagen 102. Contraste azul y blanco	242
Imagen 103. Contraste rojo y blanco	242
Imagen 104. Atenea desde enfermedades visuales comunes	243
Imagen 105. Atenea desde navegador de texto Elinks	243
Imagen 106. Atenea desde navegador de texto Lynx	244
Imagen 107. Test accesibilidad TAW	244
Imagen 108. Test accesibilidad AChecker	245
Imagen 109. Test accesibilidad WAVE	245
Imagen 110. Mecanismos de búsqueda y filtrado	246
Imagen 111. Posición actual	246
Imagen 112. XHTML válido	247
Imagen 113. CSS válido	247
Imagen 114. Comparativa tiempos de respuesta zona pública	248
Imagen 115. Comparativa de rendimiento: zona privada	249
Imagen 116. Comparativa tiempos de respuesta zona privada	249
Imagen 117. Gráfico resultados métrica WAB	259
Imagen 118. Instalar XenServer	263
Imagen 119. Administración de XenServer	264
Imagen 120. Nuevo repositorio de ISOs	264
Imagen 121. Nueva máquina virtual	265
Imagen 122. Instalar Debian Stretch	265
Imagen 123. Debian Stretch	266
Imagen 124. Instalando JDK	266
Imagen 125. Instalando Tomcat	266
Imagen 126. Página inicial Tomcat	267
Imagen 127. Configurando usuarios	267
Imagen 128. Reiniciando Tomcat	268
Imagen 129. Accediendo a Tomcat	268
Imagen 130. Tomcat instalado	268
Imagen 131. Modificando el puerto	269
Imagen 132. Instalando MySQL	269
Imagen 133. Accediendo a MySQL	269
Imagen 134. Subiendo WAR	270
Imagen 135. Aumentando tamaño de subida	270
Imagen 136. Creando usuarios para Atenea	270
Imagen 137. Manual de usuario: Pantalla de inicio	272
Imagen 138. Manual de usuario: Cargando análisis público	273
Imagen 139. Manual de usuario: Pantalla de resultados públicos	273
Imagen 140. Manual de usuario: Pantalla de ayuda pública	274
Imagen 141. Manual de usuario: Modificar idioma	274
Imagen 142. Manual de usuario: Más información	274
Imagen 143. Manual de usuario: Inicio de sesión	275
Imagen 144. Manual de usuario: Página de inicio privada	275
Imagen 145. Manual de usuario: Añadir proyecto	276
Imagen 146. Manual de usuario: Pantalla de configuración de crawler	277
Imagen 147. Manual de usuario: Pantalla proyecto abierto	277
Imagen 148. Manual de usuario: Pantalla acciones proyectos	277

Imagen 149. Manual de usuario: Pantalla importar proyecto	278
Imagen 150. Manual de usuario: Pantalla guardar consulta	279
Imagen 151. Manual de usuario: Pantalla abrir consulta	279
Imagen 152. Manual de usuario: Pantalla de análisis actual	280
Imagen 153. Manual de usuario: Pantalla analiza una URL	281
Imagen 154. Manual de usuario: Pantalla análisis de URLs	281
Imagen 155. Manual de usuario: Pantalla de exportar ficheros	282
Imagen 156. Manual de usuario: Pantalla de resultados.....	282
Imagen 157. Manual de usuario: Pantalla de errores.....	283
Imagen 158. Manual de usuario: Configuración básica	283
Imagen 159. Manual de usuario: Pantalla de adición de métricas	284
Imagen 160. Manual de usuario: Pantalla de añadir un usuario	284
Imagen 161. Manual de usuario: Pantalla de configuración de notificaciones	285
Imagen 162. Manual de usuario: Aspecto de un correo	285
Imagen 163. Manual de usuario: Menú superior derecho.....	286
Imagen 164. Diagrama de clases Checkpoints.....	288
Imagen 165. Diagrama de clases Métrica.....	289
Imagen 166. Diagrama de clases Adaptador	291
Imagen 167. Métodos y atributos de crawleradapter	292

Índice de Tablas

Tabla 1. Estudio de alternativas	27
Tabla 2. Métricas cuantitativas	31
Tabla 3. Costes indirectos: Hardware	45
Tabla 4. Costes indirectos: Oficina.....	45
Tabla 5. Coste de desarrollo	46
Tabla 6. Resumen del presupuesto	46
Tabla 7. Resumen del presupuesto para el cliente	47
Tabla 8. Estudio frameworks web	50
Tabla 9. Comparativa Struts 2 y Spring.....	56
Tabla 10. Comparativa modelo de dominio.....	60
Tabla 11. Modelo E-R Atenea 4.0	61
Tabla 12. Caso de uso de análisis público	93
Tabla 13. Caso de uso consultar resultado público	93
Tabla 14. Caso de uso consultar información pública	94
Tabla 15. Caso de uso modificar idioma zona pública.....	94
Tabla 16. Caso de uso inicio de sesión	94
Tabla 17. Caso de uso gestión de proyectos	95
Tabla 18. Caso de uso gestión de consultas.....	95
Tabla 19. Caso de uso consultar análisis en curso.....	96
Tabla 20. Caso de uso gestión de análisis	96
Tabla 21. Caso de uso consultar resultados.....	96
Tabla 22. Caso de uso consultar errores	96
Tabla 23. Caso de uso actualizar configuración básica.....	96
Tabla 24. Caso de uso gestionar usuarios	97
Tabla 25. Caso de uso añadir métrica	97
Tabla 26. Caso de uso añadir crawler	97
Tabla 27. Caso de uso configurar notificaciones	97
Tabla 28. Caso de uso consultar ayuda.....	97
Tabla 29. Caso de uso modificar idioma II	98
Tabla 30. Caso de uso cerrar sesión.....	98
Tabla 31. Descripción de la clase MetricManager.....	101
Tabla 32. Descripción de la clase Metric.....	101
Tabla 33. Descripción de la clase MetricReport	101
Tabla 34. Descripción de la clase CrawlerManager	102
Tabla 35. Descripción de la clase Crawler	102
Tabla 36. Descripción de la clase CrawlerConfiguration	102
Tabla 37. Descripción de la clase Evaluate.....	103
Tabla 38. Descripción de la clase Error	103
Tabla 39. Descripción de la clase Analyzer	103
Tabla 40. Descripción de la clase Project	103
Tabla 41. Descripción de la clase Coordinator	104
Tabla 42. Descripción de la clase GenericThread	104
Tabla 43. Descripción de la clase NotificationServices.....	104
Tabla 44. Escenario análisis público	105
Tabla 45. Escenario consultar información pública.....	106
Tabla 46. Escenario iniciar sesión	106
Tabla 47. Escenario modificar idioma.....	107

Tabla 48. Escenario actualizar configuración básica	107
Tabla 49. Escenario añadir una métrica	108
Tabla 50. Escenario añadir un crawler	108
Tabla 51. Escenario añadir usuario	109
Tabla 52. Escenario eliminar usuario	110
Tabla 53. Escenario configurar notificaciones	110
Tabla 54. Diagrama de robustez crear proyecto.....	110
Tabla 55. Escenario crear un proyecto	111
Tabla 56. Escenario abrir proyecto.....	111
Tabla 57. Escenario eliminar un proyecto	112
Tabla 58. Escenario importar un proyecto	112
Tabla 59. Escenario exportar un proyecto.....	113
Tabla 60. Escenario cerrar proyecto.....	113
Tabla 61. Escenario ejecutar consulta	114
Tabla 62. Escenario añadir consulta.....	114
Tabla 63. Escenario abrir consulta	115
Tabla 64. Escenario eliminar una consulta	115
Tabla 65. Escenario ver estado de los análisis en curso.....	116
Tabla 66. Escenario detener análisis en curso	116
Tabla 67. Escenario analizar una URL.....	117
Tabla 68. Escenario analizar una lista de URLs	118
Tabla 69. Escenario exportar análisis	118
Tabla 70. Escenario consultar resultados	119
Tabla 71. Escenario consultar errores	119
Tabla 72. Escenario consultar ayuda.....	120
Tabla 73. Escenario cerrar sesión	120
Tabla 74. Plan de pruebas para análisis público	125
Tabla 75. Plan de pruebas para inicio de sesión	125
Tabla 76. Plan de pruebas para configuración básica	126
Tabla 77. Plan de pruebas para añadir una métrica	126
Tabla 78. Plan de pruebas para añadir un crawler	126
Tabla 79. Plan de pruebas para añadir usuarios.....	127
Tabla 80. Plan de pruebas para eliminar usuario	127
Tabla 81. Plan de pruebas para configurar notificaciones	127
Tabla 82. Plan de pruebas para crear un proyecto	127
Tabla 83. Plan de pruebas para abrir un proyecto	128
Tabla 84. Plan de pruebas para eliminar un proyecto	128
Tabla 85. Plan de pruebas para importar un proyecto	128
Tabla 86. Plan de pruebas para exportar un proyecto	128
Tabla 87. Plan de pruebas para cerrar un proyecto.....	129
Tabla 88. Plan de pruebas para ejecutar una consulta	129
Tabla 89. Plan de pruebas para añadir una consulta	129
Tabla 90. Plan de pruebas para abrir una consulta	129
Tabla 91. Plan de pruebas para eliminar consulta.....	130
Tabla 92. Plan de pruebas para análisis en curso	130
Tabla 93. Plan de pruebas para detener análisis	130
Tabla 94. Plan de pruebas para analizar una URL.....	131
Tabla 95. Plan de pruebas para analizar un conjunto de URLs	131
Tabla 96. Plan de pruebas para exportar análisis.....	131
Tabla 97. Plan de pruebas para consultar resultados.....	132
Tabla 98. Plan de pruebas consultar errores	132

Tabla 99. Plan de pruebas cerrar sesión	132
Tabla 100. Especificación técnica del plan de pruebas para análisis público.....	172
Tabla 101. Especificación técnica del plan de pruebas para inicio de sesión.....	173
Tabla 102. Especificación técnica del plan de pruebas para configuración básica.....	174
Tabla 103. Especificación técnica del plan de pruebas para añadir una métrica	174
Tabla 104. Especificación técnica del plan de pruebas para añadir un crawler	175
Tabla 105. Especificación técnica del plan de pruebas para añadir usuarios.....	176
Tabla 106. Especificación técnica del plan de pruebas para eliminar usuario	176
Tabla 107. Especificación técnica del plan de pruebas para configurar notificaciones.....	177
Tabla 108. Especificación técnica del plan de pruebas para crear un proyecto.....	177
Tabla 109. Especificación técnica del plan de pruebas para abrir un proyecto	178
Tabla 110. Especificación técnica del plan de pruebas para eliminar un proyecto.....	178
Tabla 111. Especificación técnica del plan de pruebas para importar un proyecto	179
Tabla 112. Especificación técnica del plan de pruebas para exportar un proyecto	179
Tabla 113. Especificación técnica del plan de pruebas para cerrar un proyecto.....	180
Tabla 114. Especificación técnica del plan de pruebas para ejecutar una consulta.....	180
Tabla 115. Especificación técnica del plan de pruebas para añadir una consulta.....	181
Tabla 116. Especificación técnica del plan de pruebas para abrir una consulta	181
Tabla 117. Especificación técnica del plan de pruebas para eliminar consulta.....	182
Tabla 118. Especificación técnica del plan de pruebas para análisis en curso	182
Tabla 119. Especificación técnica del plan de pruebas para detener análisis	182
Tabla 120. Especificación técnica del plan de pruebas para analizar una URL.....	184
Tabla 121. Especificación técnica del plan de pruebas para analizar un conjunto de URLs	186
Tabla 122. Especificación técnica del plan de pruebas para exportar análisis.....	186
Tabla 123. Especificación técnica del plan de pruebas para consultar resultados.....	187
Tabla 124. Especificación técnica del plan de pruebas consultar errores.....	188
Tabla 125. Especificación técnica del plan de pruebas cerrar sesión.....	188
Tabla 126. Clase HibernateComandExecutor	207
Tabla 127. Clase HibernateDAOFactory.....	208
Tabla 128. Clase Analyzer.....	208
Tabla 129. Clase Coordinator	209
Tabla 130. Clase ProjectOperation	209
Tabla 131. Clase PublicOperation.....	210
Tabla 132. Clase MetricCrawlerManager.....	211
Tabla 133. Clase ServicesFactory	211
Tabla 134. Clase ObservableTrace	211
Tabla 135. Desarrollo plan de pruebas para análisis público	212
Tabla 136. Desarrollo plan de pruebas para inicio de sesión	213
Tabla 137. Desarrollo plan de pruebas para configuración básica	214
Tabla 138. Desarrollo plan de pruebas para añadir una métrica.....	215
Tabla 139. Desarrollo plan de pruebas para añadir un crawler.....	216
Tabla 140. Desarrollo plan de pruebas para añadir usuarios	217
Tabla 141. Desarrollo plan de pruebas para eliminar usuario.....	218
Tabla 142. Desarrollo plan de pruebas para configurar notificaciones	218
Tabla 143. Desarrollo plan de pruebas para crear un proyecto	219
Tabla 144. Desarrollo plan de pruebas para abrir un proyecto	219
Tabla 145. Desarrollo plan de pruebas para eliminar un proyecto	220
Tabla 146. Desarrollo plan de pruebas para importar un proyecto	221
Tabla 147. Desarrollo plan de pruebas para exportar un proyecto.....	222
Tabla 148. Desarrollo plan de pruebas para cerrar un proyecto	222
Tabla 149. Desarrollo plan de pruebas para ejecutar una consulta	223

Tabla 150. Desarrollo plan de pruebas para añadir una consulta.....	224
Tabla 151. Desarrollo plan de pruebas para abrir una consulta	224
Tabla 152. Desarrollo plan de pruebas para eliminar consulta.....	225
Tabla 153. Desarrollo plan de pruebas para análisis en curso	225
Tabla 154. Desarrollo plan de pruebas para detener análisis	226
Tabla 155. Desarrollo plan de pruebas para analizar una URL.....	227
Tabla 156. Desarrollo plan de pruebas para analizar un conjunto de URLs	230
Tabla 157. Desarrollo plan de pruebas para exportar análisis.....	230
Tabla 158. Desarrollo plan de pruebas para consultar resultados.....	231
Tabla 159. Desarrollo plan de pruebas consultar errores.....	232
Tabla 160. Desarrollo plan de pruebas cerrar sesión.....	232
Tabla 161. Datos para el informe de auditoría	233
Tabla 162. Comparativa de rendimiento: zona pública	248
Tabla 163. Conjunto de enlaces a analizar	251
Tabla 164. Resultados al analizar los enlaces de las universidades de España	255
Tabla 165. Sistema de notificaciones	257
Tabla 166. Resultados de la métrica WAB.....	258

Capítulo 1. Introducción

1.1 Justificación del Proyecto

El actual auge de las nuevas tecnologías y la repercusión de éstas en la sociedad hacen que estemos continuamente conectados a internet, ya sea a través de nuestros ordenadores o gracias a la utilización de dispositivos móviles.

Los potenciales usuarios pueden ser muy dispares y no todos cuentan con la misma capacidad para hacer uso de estas tecnologías en constante cambio, debido a su edad, cultura, localización geográfica o incluso por padecer algún tipo de discapacidad que dificulte dicha tarea. Por esto, a la hora de crear nuevos sistemas web o de escritorio es necesario reflexionar y crear interfaces de usuario que sigan una serie de pautas con el fin de que un gran número de usuarios pueda disfrutar de ellas sin ningún inconveniente y con todas las facilidades posibles.

Estas pautas que se deben seguir en la creación de un sistema para conseguir que sea accesible y, fácilmente usable, han sido descritas y normalizadas por la organización W3C (World Wide Web Consortium s.f.).

Actualmente, existen varias aplicaciones que permiten medir un sistema según su nivel de accesibilidad y usabilidad. En la Universidad de Oviedo, el Grupo de Investigación en Arquitectura del Software Empresarial (GIASE) desarrolló la aplicación de Atenea.

Atenea es una herramienta que proporciona la capacidad de evaluar de manera cuantitativa uno o más sitios web atendiendo a su nivel de accesibilidad y usabilidad, usando diferentes métricas y permitiendo así modificar e incluso mejorar aquellos aspectos en los que la web no cumple ciertos criterios de accesibilidad.

En la actualidad, Atenea cuenta con una versión tanto de escritorio como web y ambas hacen uso de distintas métricas y crawlers o arañas web que se pueden emplear en la realización de estudios sobre una web dada.

El proyecto actual tiene por objetivo una reingeniería de diversos aspectos de la aplicación Atenea y se va a centrar en la versión web de Atenea, ya que la tendencia actual es utilizar este tipo de aplicaciones dejando de lado las antiguas aplicaciones de escritorio. La actual versión web de Atenea tiene con un gran inconveniente, puesto que solo puede ser ejecutada sobre un servidor instalado en un sistema operativo Windows, lo que hace que su administración se haga mucho más tediosa.

Además, esta aplicación ya tiene algunos años y, por lo tanto, las librerías, frameworks o incluso la forma de acceder a la base de datos se han quedado anticuadas e incluso en algunos casos, obsoletas.

Por tanto, la dependencia de Atenea a Windows, así como su antigüedad son la principal motivación que lleva a la realización de este nuevo sistema Atenea por medio de un proceso de reingeniería.

1.2 Objetivos del Proyecto

Actualmente, Atenea es un proyecto que se encuentra en producción y se utiliza para realizar análisis tanto de accesibilidad como de usabilidad sobre determinados sitios webs.

Aunque este sistema esté en producción, cuenta con algunas deficiencias o errores que a lo largo del desarrollo del presente proyecto se van a subsanar.

La primera deficiencia clara, que urge de una solución rápida y estudiada, es la dependencia que este sistema presenta con los sistemas operativos Windows. Esto es, actualmente, el proyecto Atenea presenta una estrecha relación con el sistema de directorios de Windows, ya que la mayoría de los recursos que necesita para ser una solución integral como analizador, los obtiene del conocido directorio "C:\". Esta dependencia se traduce en el principal objetivo por el cual es necesario evolucionar mediante este proceso de reingeniería, la aplicación Atenea. La actual relación impide el uso de otros sistemas operativos capaces de servir la presente aplicación de una forma más transparente.

Cabe destacar, que Atenea cuenta con una cierta trayectoria en su campo, lo que conlleva la desactualización y desuso de muchas de las tecnologías con las que hasta el momento cuenta para su correcto funcionamiento. Esto se debe a que el mundo de la informática está en continuo cambio. Por todo esto, otro objetivo derivado es la necesidad de actualizar todas las posibles librerías de terceros, así como el framework utilizado para la implementación de su capa de presentación.

El tiempo que Atenea lleva en producción también conlleva la necesidad de añadir nuevas funciones al nuevo sistema, con el fin de adaptarse a los nuevos avances tecnológicos y mantener este sistema actualizado. Por esto, otro objetivo será añadir un sistema de avisos que permita a los usuarios conocer, en tiempo real y sin necesidad de acceder a la aplicación, el progreso de sus análisis en ejecución.

Es probable que a medida que se van realizando cambios y se va estudiando el sistema de partida, aparezcan nuevas cuestiones que se deban mejorar o transformar, por lo que otro objetivo es una posible reingeniería de estos aspectos que posiblemente merezcan mejoras.

Es necesario conseguir estos objetivos sin modificar la esencia de la aplicación, es decir, la aplicación siempre debe ir a mejor y nunca perder de vista su función básica, su correcto diseño o la línea base de arquitectura sobre la que está cimentada.

Finalmente, surge la necesidad de mejorar, actualizar y, finalmente, integrar la documentación de ambos proyectos en un único documento, atendiendo exclusivamente a la aplicación web y con el fin de facilitar sus posteriores modificaciones o adiciones, así como su estudio.

1.3 Estudio de la Situación Actual

1.3.1 Evaluación de Alternativas

Como ya se ha mencionado, el asunto de la accesibilidad y la usabilidad en la web cada vez está tomando más importancia, ya que es necesario que los contenidos web lleguen al mayor número de usuarios posibles.

Por esto, en la actualidad, existen aplicaciones tanto de escritorio como para la web, que se dedican a analizar y evaluar la accesibilidad, es decir, existen herramientas que miden de manera cualitativa o cuantitativa este aspecto, siendo la segunda opción menos popular.

En cuanto a la usabilidad, también existen herramientas que miden este tema, aunque con cierta monotonía debido a las exigencias de la sociedad. Esto es, están muy centradas en el estudio del diseño responsivo, es decir, la capacidad que tiene un sitio web para adaptarse a los diferentes tamaños de pantalla de los posibles dispositivos móviles que lo sirven.

A continuación, se presenta una comparativa de las herramientas estudiadas, atendiendo a una serie de pautas. Estas herramientas estudiadas son alternativas a la aplicación desarrollada en el presente proyecto.

	MEDIDA	MÉTRICAS CRAWLERS	ASPECTOS MEDIDOS	ANÁLISIS MASIVO	OTROS FORMATOS	INFORME
TAW	Cualitativa	No	Accesibilidad	No	Escritorio	Sí
ACHECKER	Cualitativa	No	Accesibilidad	Sí	Escritorio	Sí
WAVE	Cualitativa	No	Accesibilidad	No	Extensión Web	Sí
ATENEA	Cuantitativa	Sí	Accesibilidad y Usabilidad	Sí	Escritorio	Sí

Tabla 1. Estudio de alternativas

Según la anterior comparativa, se puede concluir que la herramienta más completa y diferente es Atenea. De todas formas, es necesario destacar que, aunque mide aspectos de usabilidad, no permite comprobar si una página web cuenta con un diseño adaptable.

Capítulo 2. Aspectos Teóricos

2.1 Accesibilidad en la Web

Una de las definiciones más aceptadas de accesibilidad web es la proporcionada por (Slatin y Rush 2003) (Association, IEEE-Standards 2000) “web sites are accessible when individuals with disabilities can access and use them as effectively as people who don’t have disabilities”. El éxito de la web radica, básicamente, en su alta disponibilidad y facilidad de acceso a la información. Sin embargo, muchas personas con restricciones sensoriales (principalmente visuales y auditivas), motoras (problemas de movilidad principalmente en los miembros superiores) o cognitivas (problemas a la hora de leer y comprender), que pueden ser permanentes o como resultado de una situación transitoria, suelen encontrar serios problemas en el acceso. Aunque las personas con discapacidad son un claro ejemplo de la exclusión por falta de accesibilidad, no son las únicas que la experimentan. Similares restricciones encuentran quienes usan equipos con visualizadores de pequeño tamaño o poca definición, conexiones de bajo ancho de banda, navegadores antiguos o especiales, etc.

2.1.1 Pautas de accesibilidad en la web

En los últimos años han surgido diferentes iniciativas con el objetivo de fomentar y garantizar la accesibilidad en la Web. Una de las más significativas es la Iniciativa de Accesibilidad a la Web (Web Accessibility Initiative s.f.) creada por el Consorcio W3C (World Wide Web Consortium s.f.) que proporciona un soporte a los desarrolladores de contenidos y a los desarrolladores de herramientas de autor.

Estas pautas se dividen en tres bloques fundamentales:

- **Pautas de Accesibilidad al Contenido en la Web (WCAG):** dirigidas principalmente a desarrolladores de páginas web. Proporcionan las indicaciones necesarias para hacer que una web sea accesible.
- **Pautas de Accesibilidad para Herramientas de Autor (ATAG):** dirigidas principalmente a desarrolladores de herramientas de autor. Éstas tratan de definir una serie de normas tanto para las herramientas de autor como para que los contenidos que generen éstas sean accesibles.
- **Pautas de Accesibilidad para Agentes de Usuario (UAAG):** dirigidas principalmente a los desarrolladores de agentes de usuario. Entre los agentes de usuario se incluyen navegadores, reproductores multimedia y tecnologías asistidas, software que algunas personas con discapacidad utilizan para interactuar con los dispositivos.

De los tres conjuntos de pautas, las pautas de la WCAG (Web Content Accessibility Guidelines s.f.) son las que están siendo utilizadas como referentes principales a la hora de analizar la accesibilidad de un sitio web. Existen dos versiones de esta pauta: WCAG 1.0 y la WCAG 2.0.

2.1.1.1 WCAG 1.0

La WCAG 1.0 está organizada en 14 pautas estructuradas en diferentes checkpoints con tres niveles de prioridad:

- La prioridad 1 se refiere a los checkpoints que el desarrollador tiene que satisfacer para que no existan grupos de personas incapaces de acceder a la información del sitio.
- La prioridad 2 es para los checkpoints que el desarrollador debe satisfacer, ya que si no lo hace algunas personas tendrán dificultades para acceder a la información.
- La prioridad 3 es para los checkpoints que el desarrollador puede satisfacer, ya que si no se satisfacen algunas personas encontrarán problemas para acceder a la información.

2.1.1.2 WCAG 2.0

La WCAG 2.0 es la evolución de su análoga y fue publicada por la W3C en el 2008. Esta pauta se rige principalmente por cuatro pilares fundamentales:

- Perceptible.
- Operable.
- Comprensible.
- Robusto.

Además, estos principios se organizan en 12 pautas y distintos criterios de éxito asociados al nivel de conformidad.

2.1.2 Evaluación de la accesibilidad en la web

Para evaluar la accesibilidad en la web existen métricas, mediante las que es posible realizar mediciones tanto cualitativas como cuantitativas.

Este apartado se va a centrar en describir las métricas que realizan mediciones cuantitativas, ya que con las cualitativas no se puede determinar de forma precisa el nivel de accesibilidad de una aplicación web.

A continuación, se detallan las principales métricas cuantitativas. Además, alguna de ellas es utilizada en el presente proyecto, para realizar las pertinentes mediciones que permitan probar el software.

NOMBRE	DESCRIPCIÓN	FÓRMULA	EXPLICACIÓN FÓRMULA	RESULTADO
TE	Métrica basada en la evaluación de 8 checkpoints de prioridad 1.	$\frac{e_reales}{e_potenciales}$	Siendo e_reales el número de casos donde no se cumplen los puntos pautados y e_potenciales el número total de casos donde sí se cumplen esos puntos.	Es un valor comprendido entre 0 y 1. Un valor cercano a 0 indica que la web es accesible y, por el contrario, un valor cercano a 1 indica que la web no lo es.
WAB	Métrica que mide las violaciones a las pautas de la WCAG 1.0.	$\frac{\sum \rho \sum v (\eta v / N v) W v}{N \rho}$	Siendo p el número total de páginas que componen el sitio, v el número total de errores en una página, nv el número de errores, Nv el número de errores potenciales, Wv el peso de cada error y Np el número total de páginas evaluadas.	Un valor WAB=0 significa que el sitio evaluado tiene un nivel de accesibilidad aceptable. Un valor WAB<=5.5 significa que el sitio presenta pocas barreras de accesibilidad. Por el contrario, un valor superior a 5.5 implica que el sitio presenta muchas barreras de accesibilidad.
UWEM	Métrica que mide la probabilidad de encontrar barreras en la página.	$\frac{\sum i B \rho i}{\sum i N \rho i}$	Siendo Bpi el número de fallos producidos como resultado de aplicar los test en la página <i>i</i> y Npi el número de veces que se han aplicado los test dentro de la página <i>i</i> .	Es un valor comprendido entre 0 y 1. Un valor cercano a 0 indica que la web es accesible y, por el contrario, un valor cercano a 1 indica que la web no lo es.
WAB*			Esta métrica está compuesta por la colaboración de las dos métricas anteriores.	

Tabla 2. Métricas cuantitativas

2.2 Usabilidad en la web

Jacob Nielsen es considerado el padre de la usabilidad y la definió como el atributo de calidad, que mide lo fáciles de usar que son las interfaces web. Esto es, un sitio web usable es aquél en el que los usuarios pueden interactuar de la forma más fácil, cómoda, segura e inteligente posible.

No sólo la tecnología y el aspecto gráfico son factores determinantes para hacer un sitio web llamativo. Es importante que cumpla con las siguientes características:

- Entendible
- Novedoso
- Comprensible
- Inteligente
- Atractivo

Es decir, la finalidad, en este caso de un sitio web, es lograr que el usuario encuentre lo que busca en el menor tiempo posible.

La usabilidad de un sitio web está determinada por sus contenidos, cuanto más cercanos estén al usuario, mejor es la navegación por el mismo y más acertada será la experiencia al enfrentarse a la aplicación.

Aunque no hay estándares definidos para la usabilidad, depende en cierta forma del espacio donde se desenvuelve el navegante. Pero lo importante en este caso es que el usuario no se deje consumir ni dominar por el sitio, es decir que sea él mismo quien tome el control de la navegación por medio de un aprendizaje sencillo y el dominio de los elementos necesarios, para encontrar finalmente y en el menor tiempo posible, lo que busca.

Un buen sitio web debe responder a las necesidades del usuario. En una comunidad virtual donde confluyen diferentes culturas e intereses, el contexto en el que se desenvuelven los miembros de un grupo virtual, o comunidad, no puede generar molestias en el momento de la navegación.

Un error recurrente de los creadores y diseñadores de sitios web es querer imponer sus decisiones y criterios sin pensar en el usuario. Por eso en el momento de diseñar el sitio e introducir contenidos, siendo esta última labor de los editores y no de los diseñadores, es importante pensar en el otro.

2.2.1 Evaluación de la usabilidad en la web

En este proyecto se utilizan una serie de métricas descritas en el trabajo llamado “*Empirically Validated Web Page Design Metrics*” (Ivory, Sinha y Hearst 2001), para evaluar de forma cuantitativa la usabilidad, proporcionando como resultado un valor numérico. Se mencionan a continuación:

- **TOTAL WORDS:** Esta métrica cuenta el número de palabras que contienen los documentos HTML de un determinado sitio Web. Está compuesta por un único checkpoint y su evaluación es automática.
- **TOTAL IMAGES:** Esta métrica cuenta el número de imágenes que contiene un sitio Web. Está compuesta por un único checkpoint y su evaluación es automática.
- **TOTAL FONTS:** Esta métrica cuenta el número de fuentes que se aplican en un sitio Web. Está compuesta por un único checkpoint y su evaluación es automática.
- **TEXT CLUSTER:** Esta métrica cuenta el número de porciones de texto destacadas con bordes, colores, reglas o listas, de los documentos HTML de un determinado sitio Web. Está compuesta por un único checkpoint y su evaluación es automática.
- **EMPHASIZED TEXT:** Esta métrica calcula el porcentaje de palabras pertenecientes al cuerpo de los documentos HTML que están enfatizadas (negrita, cursiva, etc.), del total de palabras de los mismos, de un determinado sitio Web. Está compuesta por un único checkpoint y su evaluación es automática.
- **IVORY:** La métrica Ivory está compuesta por 5 checkpoints y su evaluación es automática. Evalúa una serie de criterios que aparecen reflejados en la siguiente fórmula que calcula el valor devuelto por la métrica:

$$Ivory = 0.129 + 0.002 * Link\ Count - 0.011 * Text\ Positioning\ Count + 0.020 * Color\ Count + 8.734 * 10^{-7} * Page\ Size + 0.002 * Body\ Text\ %$$

Donde *Link Count*, *Text Positioning Count*, *Color Count*, *Page Size* y *Body Text %* son los valores calculados automáticamente por sus respectivos checkpoints.

El valor resultante de aplicar esta fórmula tiene que ser un valor entre 0 y 1. Si el valor está cerca de 0, la página será "no buena"; si está cerca de 1, será "buena".

2.3 Crawler o araña web

Una araña web es un pequeño programa diseñado para atravesar la red de una manera mecánica y automatizada. Existen diversos términos para mencionar a este tipo de software como por ejemplo *spiders*, *crawlers*, *bots*... y al proceso de búsqueda se le llama *Web crawling* o *spidering*.

En Internet se maneja una grandísima cantidad de información y, en particular, en la *World Wide Web* existen billones de páginas con ingente cantidad de información asociada a ellas. Realizar estudios para determinar la estructura y comportamiento de las webs y las relaciones existentes entre éstas implica extraer una gran cantidad de información de las propias páginas. Recoger toda esta información de manera manual sería demasiado costoso y, prácticamente imposible, es aquí donde entran los crawlers jugando un papel fundamental para la recopilación de esta información de manera fiable y automática.

El principal uso de las arañas web es en los motores de búsqueda como *Google* o *Yahoo!* que las emplean para recopilar la información de los sitios web y así poder indexar sus contenidos.

Pero éste no es el único uso que se le puede dar a este tipo de software. Como se mencionó anteriormente se pueden usar para extraer información de la estructura y contenidos web, verificar la disponibilidad de enlaces, avisar de cambios en las páginas, estudiar el uso de la web, analizar el acceso a cierta información alojada en diferentes sedes web, realizar búsquedas...

En función de la forma en la que rastrean la red se pueden clasificar en:

- **Broad crawler:** recolectan la mayor cantidad de información posible, dentro de un cierto intervalo de tiempo.
- **Incremental crawlers:** vuelven a visitar las páginas que ya han bajado en busca de modificaciones.
- **Focused crawlers:** buscan información sobre un tema específico.
- **Deep o hidden web crawlers:** Tienen la capacidad de rellenar informes para conseguir páginas dinámicas, por ejemplo, acceso a bases de datos con clave.

2.3.1 Funcionamiento

El funcionamiento básico de una araña web o crawler consiste en que a partir de una o varias URLs iniciales (semillas) la araña va descargando de forma local las páginas web asociadas a éstas y buscando dentro de ellas otras nuevas URLs. Cada nueva URL que se va encontrando se almacena en una lista para que posteriormente la araña pueda visitarla y así de forma recursiva.

Finalmente, cuando un determinado número de páginas son descargadas éstas se analizan y se procesa la información obtenida se procesa aplicando un determinado algoritmo para

conseguir un objetivo previamente establecido. La siguiente figura muestra el funcionamiento básico de un crawler o araña web.

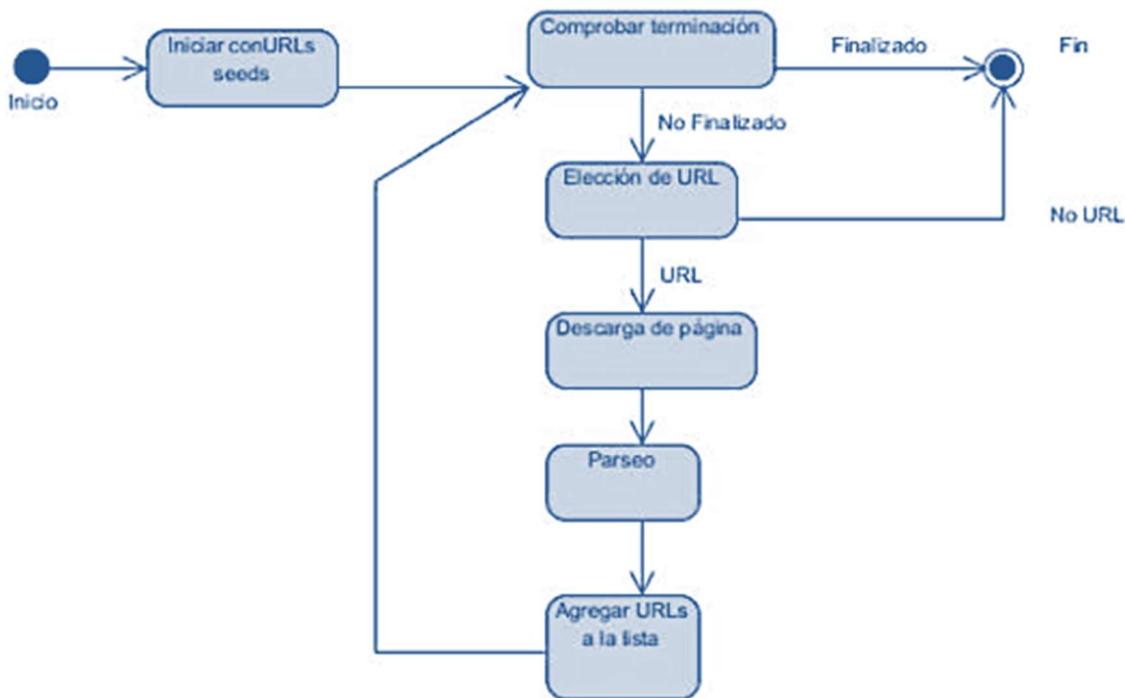


Imagen 1. Funcionamiento básico de un crawler

El recorrido que el crawler realiza sobre las páginas web puede ser de dos tipos:

- **Recorrido primero en anchura:** se siguen los enlaces que lleven a otras páginas dentro del mismo nivel de profundidad de la web en el orden en que son descubiertas.
- **Recorrido primero en profundidad:** el crawler sigue los enlaces que le dirigen a otras páginas de niveles inferiores dentro de la estructura jerárquica de la web.

Los múltiples problemas a los que se enfrenta un crawler son, principalmente, el manejo de enormes cantidades de páginas a recorrer, el elevado número de actualizaciones de páginas existentes que se realizan en cada momento, páginas que crean su contenido de forma dinámica, redireccionamientos, etc.

Algunos de los ajustes que se pueden aplicar al crawler para minimizar el efecto de los problemas anteriores son:

- Acotar el procesamiento del contenido de las páginas web a tipos de objetos concretos, como imágenes.
- Acotar el rastreo a ciertos dominios o subdominios.
- Limitar el número de páginas a explorar, los niveles en una URL o el tamaño de los ficheros a descargar.

Capítulo 3. Planificación del Proyecto y Presupuestos

3.1 Planificación

La planificación temporal tiene como objetivo principal mostrar el plazo de ejecución del proyecto, de esta manera, el cliente puede hacerse una idea de cuándo va a recibir el producto final.

Para el desarrollo del presente proyecto, se ha creado un calendario personalizado, ya que, durante el mes de marzo, por motivos profesionales, solamente se contaba con 4 horas al día de trabajo. En cuanto a los demás meses de ejecución del proyecto, se mantenía un horario laboral de 8 horas de trabajo al día, con un periodo de descanso.

A continuación, se presenta la estructura de la descomposición del trabajo, agrupada según la categoría de las distintas tareas y reflejada en forma de árbol.

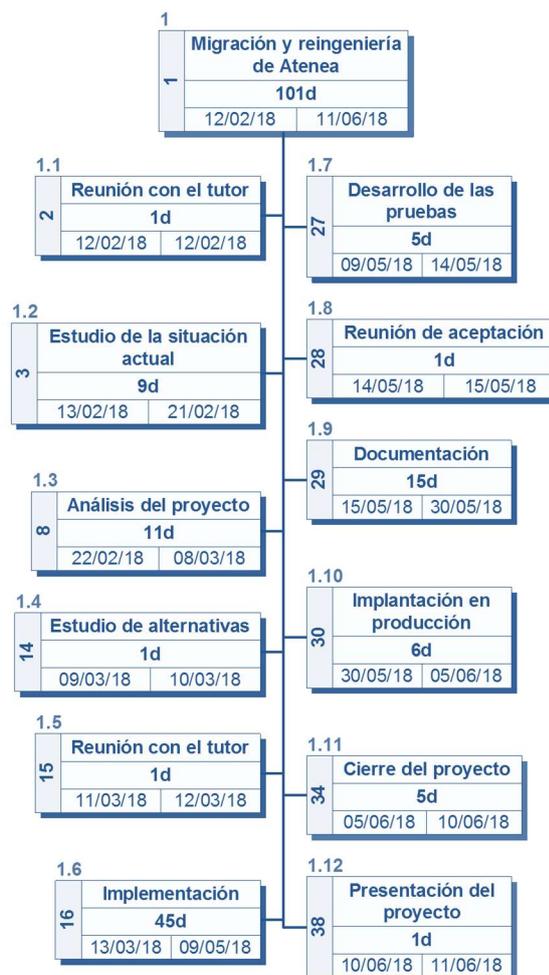


Imagen 2. Estructura general de la planificación

La ejecución del proyecto está condicionada principalmente por cinco grandes fases que, a su vez, engloban todas las tareas que deben llevarse a cabo para poder producir el sistema final. Entre estas fases, y como se puede observar en la imagen anterior, se encuentra una dirigida al estudio y conocimiento de la situación actual, otra destinada a la reflexión y documentación del análisis del presente proyecto, otra dedicada a la implementación, una cuarta para a la implantación en producción y, por último, una quinta fase que desencadena el cierre del proyecto.

Además, a lo largo del proyecto, se realizarán distintas reuniones, en las que se pondrá en conocimiento la idea principal del proyecto, los cambios, los progresos y las decisiones.

Aunque durante el proceso se hacen paradas intermedias para la realización de pruebas, el tiempo empleado en estas se contabilizará todo como una sola fase.

Finalmente, cabe destacar que aparecen dos tareas dedicadas a la documentación final del proyecto, así como a la preparación de su presentación.

3.1.1 Estudio de la situación actual

Como ya se ha reflejado, este proyecto surge de la necesidad de modificar y actualizar un proyecto ya existente, con el fin de adaptarse a las nuevas tecnologías y mejorar algunas deficiencias que este presenta. Por esto, es necesario dedicar un tiempo al análisis exhaustivo del proyecto de partida, es decir, se estudiará la documentación, el código y se someterá el sistema a diversas pruebas para conocer su funcionamiento integral.

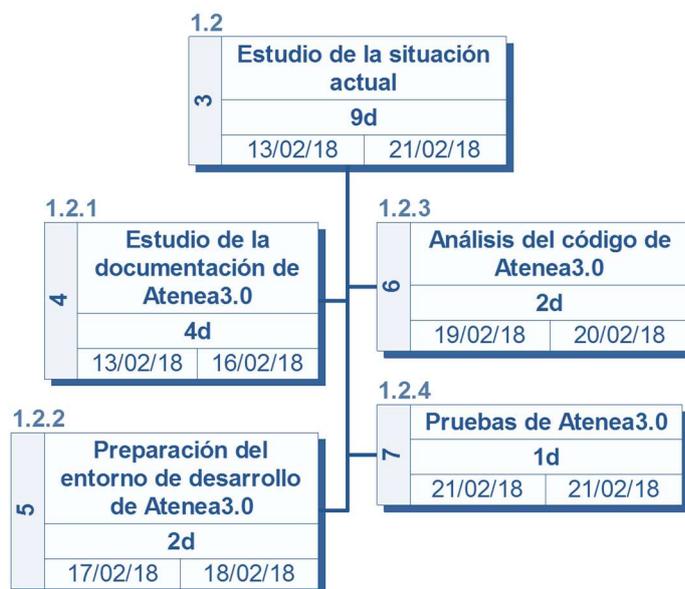


Imagen 3. Estructura de la fase de estudio

3.1.2 Análisis del proyecto

En esta fase se encapsulan tareas previas al desarrollo dedicadas al diseño tanto del modelo E-R como del modelo de clases, la definición de los diversos casos de uso, el diseño de prototipos o las especificaciones del plan de pruebas que se realizarán tras la implementación.

Además, tras estudiar el proyecto de partida, se tienen claras sus deficiencias y se puede definir el alcance del nuevo sistema.

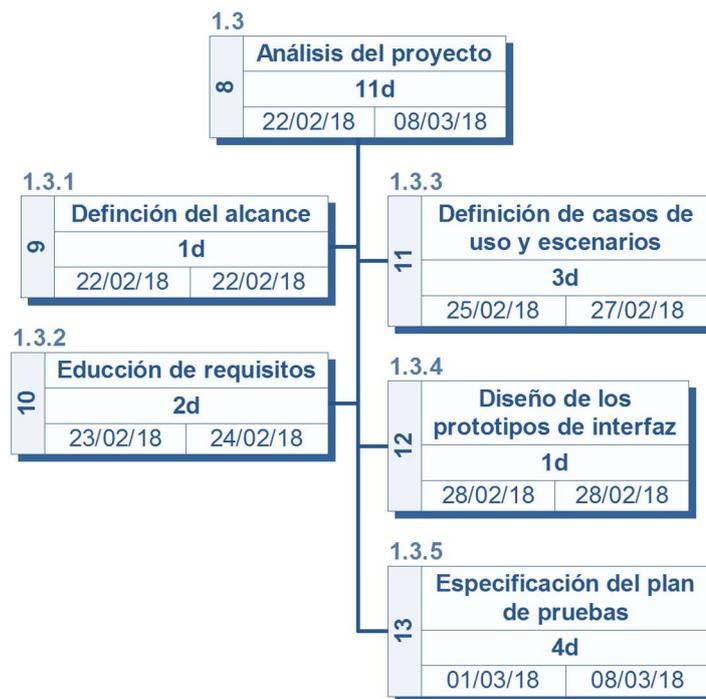


Imagen 4. Estructura de la fase de análisis

3.1.3 Implementación

Tras el estudio de la documentación comenzará la fase de implementación, que durará aproximadamente un mes y medio. El adecuado transcurso de este mes será primordial para que el producto salga adelante, ya que esta fase se centra en la implementación del sistema integral.

La fase de implementación, al ser tan extensa, se ha dividido a su vez en distintos conjuntos de tareas relacionadas. Esto es, aparecen dos fases, una dedicada a la reingeniería del proyecto y otra a la adición de nuevas funcionalidades.

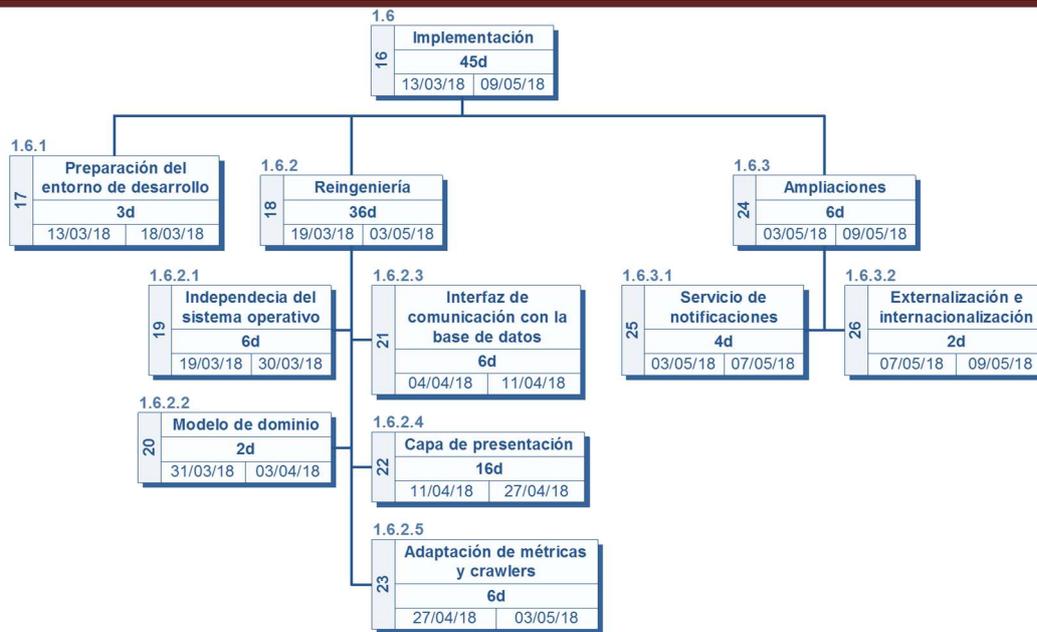


Imagen 5. Estructura de la fase de implementación

Durante la fase de reingeniería se realizarán las tareas necesarias, con el fin de darle al proyecto una independencia del sistema operativo, actualizar sus librerías y frameworks, modificar el modelo de dominio o realizar diversas transformaciones para poder integrar las anteriores modificaciones.

En la fase de ampliaciones, se añade, principalmente, la nueva funcionalidad de enviar y recibir notificaciones de la ejecución del análisis, mediante correos electrónicos.

3.1.4 Implantación en producción

Una vez se ha desarrollado y finalizado el producto, será necesario llevar a cabo una serie de actividades que permitan poner en producción y a disposición de los potenciales usuarios las mejoras que este nuevo sistema les aportará.

Entre estas tareas se encuentra la instalación del sistema operativo que hará las veces de servidor, así como de los programas que, en conjunto, servirán la funcionalidad completa del sistema. Además, se desplegará el sistema.

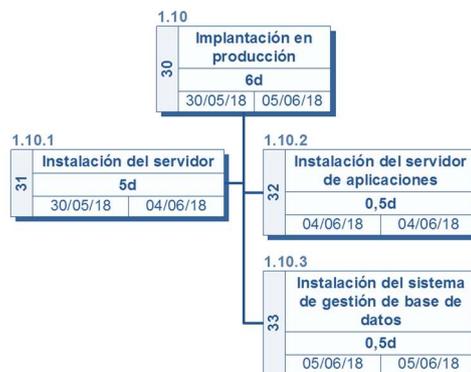


Imagen 6. Estructura de la fase de implantación

3.1.5 Cierre del proyecto

En último lugar, para finalizar la ejecución del proyecto con éxito, será necesario realizar unas últimas tareas meramente administrativas, como son la elaboración de los manuales de usuario, o también, la recopilación y archivado de toda la documentación generada y utilizada en la ejecución del proyecto, así como su código fuente.

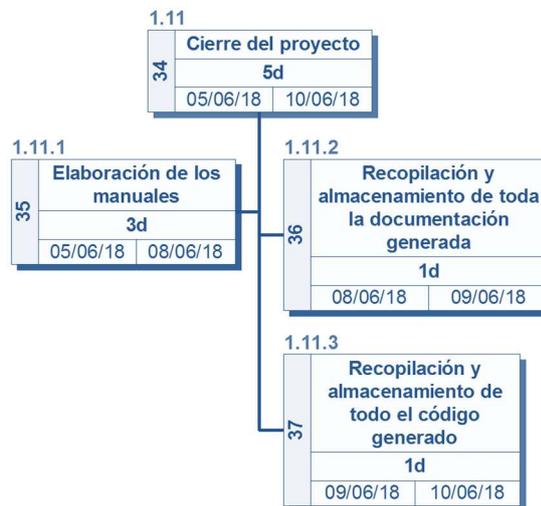


Imagen 7. Estructura de la fase de cierre del proyecto

3.1.6 Diagrama de Gantt

A continuación, se presenta el diagrama de Gantt que representa el desarrollo del proyecto, con el fin de dar una mayor comodidad al lector, se han simplificado las tareas.

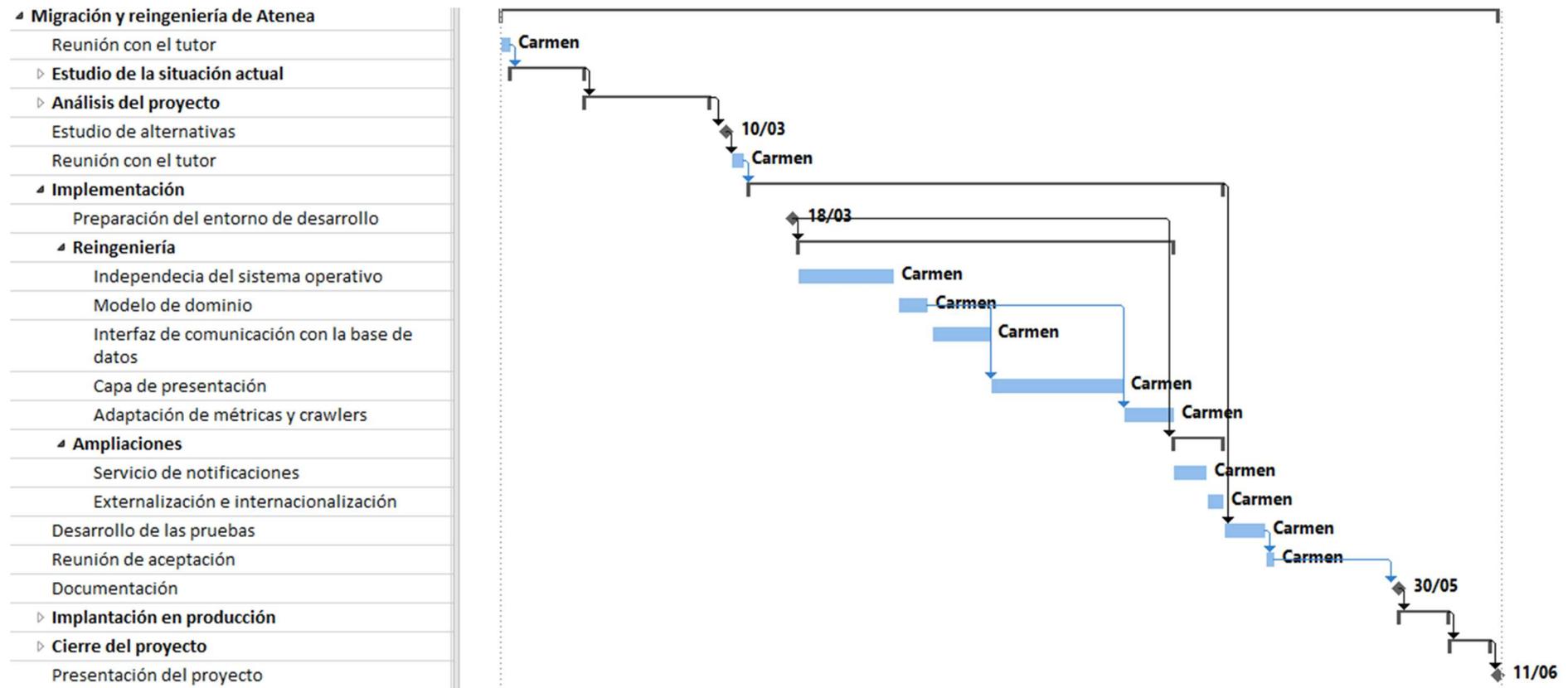


Imagen 8. Diagrama de Gantt

3.2 Presupuesto

La realización de todo proyecto lleva consigo la necesidad de crear un presupuesto que permita dar una visión del coste económico que conllevará la creación de dicho proyecto.

3.2.1 Costes indirectos

Para la ejecución del proyecto se necesita utilizar tanto un ordenador de desarrollo como un servidor para alojar la aplicación. Con intención de calcular el coste de estos equipos, se calcula el factor de amortización de ambos, atendiendo a la duración del proyecto, en este caso, 101 días.

Ordenador de desarrollo

$$\text{Factor de amortización} = \frac{101 \text{ días}}{4 \text{ años} \times 365 \text{ días}} \approx 7\%$$

Servidor

$$\text{Factor de amortización} = \frac{101 \text{ días}}{10 \text{ años} \times 365 \text{ días}} \approx 3\%$$

Una vez se han calculado las amortizaciones, se puede calcular el precio total de utilización unitario de ambos equipos.

HARDWARE	PRECIO	FACTOR DE AMORTIZACIÓN	TOTAL
ORDENADOR DE DESARROLLO	1.000,00 €	7 %	70,00 €
SERVIDOR	1.500,00 €	3 %	45,00 €
		TOTAL	115,00 €

Tabla 3. Costes indirectos: Hardware

Dentro de los costes indirectos es necesario contemplar los gastos provenientes de la oficina.

OFICINA	MESES	PRECIO/MES	TOTAL
ALQUILER	4	300,00 €	1.200,00 €
MATERIAL DE OFICINA	4	55,00 €	220,00 €
INTERNET/TELÉFONO	4	62,00 €	248,00 €
AGUA	4	17,70 €	70,80 €
ELECTRICIDAD	4	156,00 €	624,00 €
		TOTAL	2.362,80 €

Tabla 4. Costes indirectos: Oficina

3.2.2 Costes de desarrollo

Gracias a la planificación del desarrollo del proyecto, se pueden obtener las horas que llevará la elaboración de cada una de las tareas. Teniendo en cuenta un salario de 8 euros la hora para el alumno, el precio total del proyecto sería el siguiente:

TAREAS	HORAS	PRECIO/HORA	TOTAL
ESTUDIO DE LA SITUACIÓN ACTUAL	80	25,00 €	2.000,00 €
ANÁLISIS	104	25,00 €	2.600,00 €
IMPLEMENTACIÓN	360	20,00 €	7.200,00€
DESARROLLO DE LAS PRUEBAS	48	15,00 €	720,00 €
DOCUMENTACIÓN	120	25,00 €	3.000,00 €
IMPLANTACIÓN	48	15,00 €	720,00 €
CIERRE DEL PROYECTO	48	25,00 €	1.200,00 €
	TOTAL		17.440,00 €

Tabla 5. Coste de desarrollo

3.2.3 Resumen del presupuesto

Finalmente, al coste de desarrollo se le incrementan los gastos indirectos, así como el beneficio (25%) que le corresponde a la empresa. Además, se asigna un total del 15% del presupuesto estimado como reserva para poder solventar algún tipo de imprevisto.

CONCEPTO	TOTAL
COSTE DE DESARROLLO	17.440,00 €
GASTOS INDIRECTOS	2.477,80 €
PLAN DE CONTINGENCIAS (15%)	2.616,00 €
BENEFICIO (25%)	4.360,00 €
PRESTUPUESTO TOTAL ESTIMADO	26.839,80 €

Tabla 6. Resumen del presupuesto

3.2.4 Presupuesto indicativo para el cliente

Finalmente se presenta un presupuesto dirigido exclusivamente al cliente, que le permita valorar el coste de una forma rápida y clara y poder así tomar la decisión de aceptar o no la ejecución del proyecto.

TAREAS	TOTAL
ESTUDIO DE LA SITUACIÓN ACTUAL	3.084,15 €
ANÁLISIS	4.009,40 €
IMPLEMENTACIÓN	11.102,94 €
DESARROLLO DE LAS PRUEBAS	1.110,29 €
DOCUMENTACIÓN	4.626,23 €
IMPLANTACIÓN	1.110,29 €
CIERRE DEL PROYECTO	1.850,49 €
PRESUPUESTO ESTIMADO ANTES DE IMPUESTOS	26.839,80 €
IVA (21%)	5.647,70 €
PRESUPUESTO TOTAL ESTIMADO	32.541,50 €

Tabla 7. Resumen del presupuesto para el cliente

El presupuesto indicativo para el cliente integra, en cada una de las unidades en las que se desglosa el proyecto, tanto los costes indirectos como el beneficio y el dinero asociado al plan de contingencias. También incluye, finalmente, el impuesto del valor añadido, IVA, del 21%.

Capítulo 4. Reingeniería del sistema

Atenea es un proyecto que ha pasado por muchas fases durante su vida como sistema para la evaluación cuantitativa de la accesibilidad de sitios web.

Comenzó siendo una simple aplicación de escritorio que permitía realizar análisis de sitios web individuales y basándose únicamente en un tipo determinado de métricas adaptas a la WCAG 1.0.

A lo largo de los años, se han ido introduciendo mejoras para llegar a una aplicación web que, a grandes rasgos, permite:

- Introducir en tiempo de ejecución diferentes tipos de métricas en tiempo de ejecución, que miden aspectos tanto de accesibilidad como de usabilidad.
- Analizar varios sitios web simultáneamente, gracias a la incorporación de hilos que permiten trabajar en paralelo.
- Introducir, en tiempo de ejecución, nuevos crawlers o arañas web que permitan analizar, de diferentes maneras, todas las webs asociadas a una dada inicialmente.
- Visualizar la traza de una evaluación, es decir, mostrar que sitios webs han sido analizados hasta el momento y cuáles no. Así como un sistema de log que facilita en mayor medida, al desarrollador, encontrar errores para posteriormente solventarlos.

Aunque a priori, parece que esta última versión web de Atenea 3.0 se acerca mucho a la idea original e incluso compite, llegando a mejorar la funcionalidad de muchas alternativas existentes en el mercado, presenta algunos fallos o deficiencias que, en este proyecto, se van a mejorar.

Una de las principales deficiencias que justifica la necesidad de retomar el desarrollo de Atenea, es que dicho proyecto depende completamente de la unidad "C:\\" del sistema de archivos de Windows. Lo que limita, al administrador de la aplicación, modificar el sistema operativo bajo el que se sirve dicha aplicación.

A medida que se fue realizando un estudio exhaustivo de la documentación y del propio código de la aplicación, para poder solventar el defecto anteriormente citado, se fueron encontrando nuevos errores e incluso nuevas necesidades que, respectivamente, se solventaron e implementaron.

Además, cabe destacar que la última versión de Atenea 3.0 cuenta ya con algunos años y, por lo tanto, se han actualizado librerías y frameworks externos que la aplicación utiliza para su correcto funcionamiento.

A lo largo de este capítulo, se mostrará el proceso de reingeniería que se ha llevado a cabo para llegar a un nuevo Atenea 4.0, actualizado, mejorado y sin fisuras.

4.1 Contexto del proyecto

A continuación, se muestra el diagrama de paquetes de la versión web de Atenea 3.0 que servirá de punto de partida y, a lo largo del presente apartado, se harán continuas referencias a él, mostrando las diferentes transformaciones que sufrirá durante el desarrollo del nuevo Atenea 4.0, así como las nuevas aportaciones que se incorporarán.

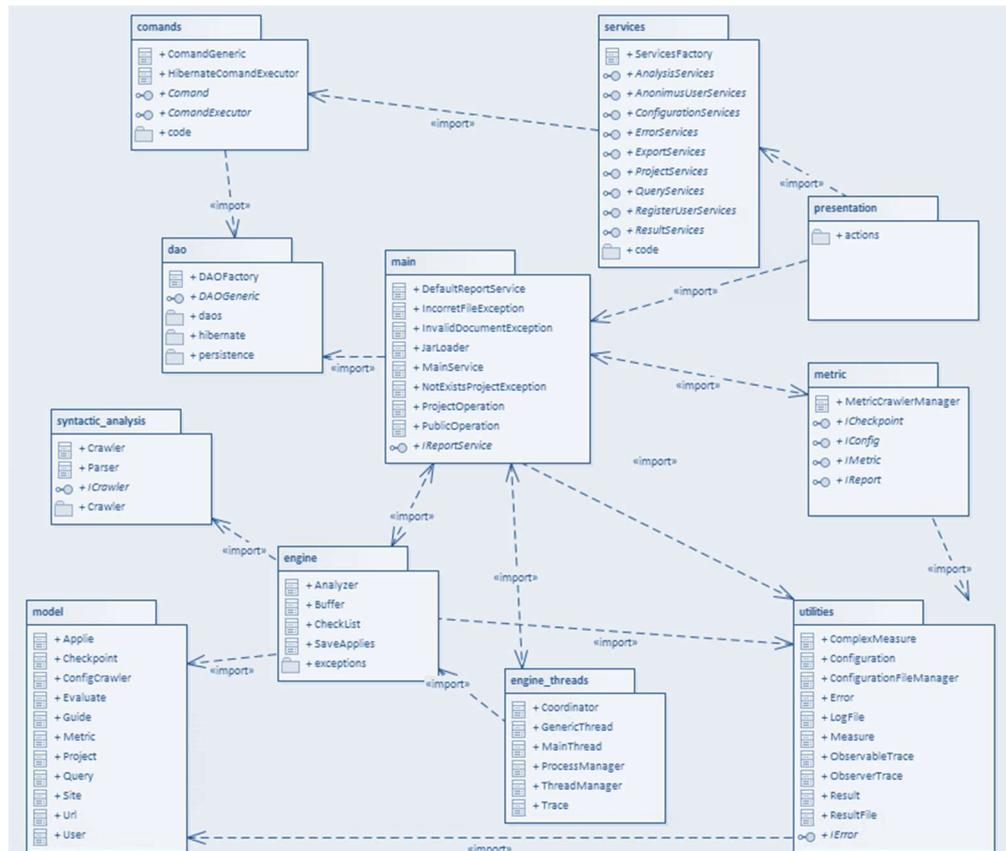


Imagen 9. Diagrama de paquetes Atenea 3.0

Cabe destacar que el diagrama anterior difiere levemente de la realidad, se omiten ciertos subpaquetes y clases irrelevantes, con el fin de clarificar y simplificar dicho diagrama.

Atenea 3.0 es una aplicación web implementada en el lenguaje de programación Java, que utiliza software de terceros para su desarrollo, entre estos, se encuentran:

- *Hibernate* como herramienta de mapeo objeto-relacional, que facilita el mapeo de atributos entre una base de datos relacional y el modelo de objetos de una aplicación. Para ello se emplean archivos declarativos (XML) que permiten establecer estas relaciones.
- *Struts 2* como framework web para implementar el patrón de diseño MVC que formará la capa de presentación junto con las correspondientes vistas JSP.

Estos, junto a las demás librerías, métricas y crawlers utilizados en el proyecto, se cargan desde el directorio "C:\\" del sistema operativo Windows.

4.2 Creación del proyecto

Tras haber probado Atenea 3.0 y un largo periodo de reflexión, se decide modificar el framework web *Struts 2*, utilizado hasta el momento. Para ello, se realiza un estudio exhaustivo de las tendencias actuales en el desarrollo de aplicaciones webs.

A la hora de seleccionar el nuevo framework para implementar la capa de presentación del nuevo y mejorado sistema Atenea 4.0, se planteaban tres posibles competidores, además de la opción de mantener *Struts 2*. Entre estos competidores se encuentran: *Spring*, *Play* y *JSF*.

Tras analizar estos frameworks según una serie de criterios, se elige *Spring*, atendiendo al resultado obtenido. Dicho resultado queda reflejado en la siguiente tabla comparativa.

	 SPRING	 PLAY	 JSF	 STRUTS
DESARROLLO RÁPIDO	✓	✗	✗	✓
CÓDIGO PRE-CONSTRUIDO	✓	✗	✓	✗
INTEGRACIÓN FÁCIL CON TERCEROS	✓	✗	✗	✗
RESTRICCIONES EN LA INTERFAZ DE USUARIO	✓	✓	✓	✓
RENDIMIENTO EFICIENTE	✓	✓	✗	✓

Tabla 8. Estudio frameworks web

Además, cuando se trata de popularidad, *Spring Framework* también se encuentra en la cima de la comparativa.

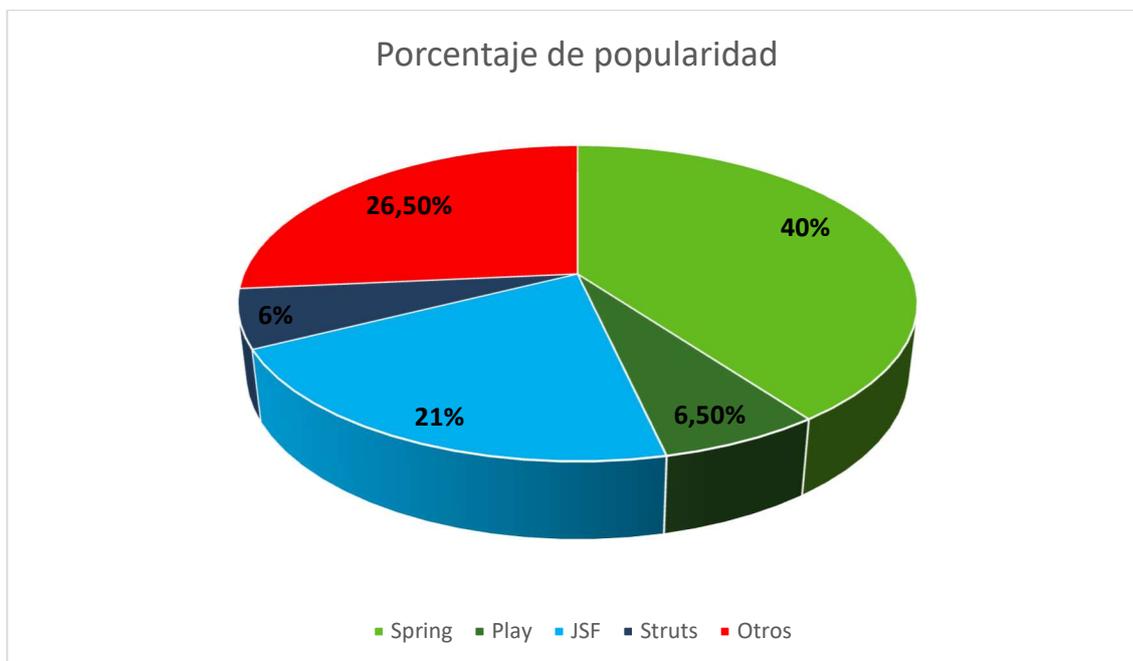


Imagen 10. Popularidad frameworks web

Spring es un framework muy consolidado y no solo aporta ventajas a la hora de implementar la capa de presentación, sino que también da soporte a las demás capas necesarias en una aplicación.

Cabe destacar que a la hora de adaptar la aplicación Atenea 3.0 para llevar a cabo la migración entre *Struts 2* y *Spring*, se va a prescindir de esta capacidad ventajosa que el framework proporciona. Si bien se hace uso de algunas de sus capacidades como *Spring Security*, para crear un sistema de inicio de sesión seguro, aunque siempre creando todas las clases necesarias que permitan seguir la arquitectura dada.

Esto se debe a la necesidad y requisito obligatorio de que el sistema tiene mantenga una arquitectura sólida e independiente de cualquier cambio que se pueda producir a lo largo de su existencia. Es decir, se mantendrá un esqueleto consolidado que permita, en un futuro, como ahora, poder modificar la capa de presentación fácilmente y sin tener que modificar el núcleo lógico de la aplicación.

Finalmente, se ha selecciona *Spring* ya que va de la mano de una nueva tecnología denominada *Spring Boot*. Para explicar *Spring Boot*, hay que tener en cuenta todos los pasos que se deben llevar a cabo para crear un proyecto con *Spring*. El primer paso es crear un proyecto Maven e incluir en el fichero de descripción del proyecto, todas las dependencias necesarias. En segundo lugar, se desarrolla la aplicación y, por último, se despliega la misma.

De estos pasos, tanto el primero como el tercero tratan aspectos de infraestructura y no se centran en el desarrollo de la aplicación. Por esto, surge la idea de *Spring Boot*, ya que simplifica dichos pasos y pone a disposición del desarrollador un asistente que crea y permite desplegar el proyecto sin muchos esfuerzos y permitiendo, siempre, centrarse en la implementación del producto deseado.

Generate a

Maven Project

with

Java

and Spring Boot

2.0.2

Project Metadata

Artifact coordinates

Group

miw.tfm

Artifact

atenea

Dependencies

Add Spring Boot Starters and dependencies to your application

Search for dependencies

Web, Security, JPA, MySQL

Selected Dependencies

Generate Project

Imagen 11. Asistente Spring Boot

Con la creación de un proyecto Maven, que permite mantener y organizar todas y cada una de las librerías externas, se soluciona parte de los problemas que provienen de obtener algunos ficheros JAR del directorio “C:\” de Windows.

Aprovechando la integración de Maven en el proyecto, se actualizan, en la medida de lo posible, las librerías necesarias para el correcto funcionamiento del sistema y se realizan las pertinentes modificaciones en la lógica, para adaptarse a los cambios sufridos, por estas, a lo largo del tiempo. Aunque esto solventa algunos problemas también hace que aparezcan otros que se comentarán a lo largo de los próximos apartados.

Con el proyecto creado se comienza a reproducir el esqueleto de paquetes y clases de Atenea 3.0, para ir modificándolo y añadiendo nuevos paquetes o clases a petición de las necesidades.

4.3 Modificaciones internas

A continuación, se desarrollarán todas las modificaciones que ha sufrido Atenea 3.0 para evolucionar a Atenea 4.0, atendiendo a la estructura interna de paquetes.

4.3.1 Paquete main

Dentro del paquete main se encuentra la clase *JarLoader*, cuyo cometido es cargar tanto las métricas como los crawlers desde el directorio “C:\” de Windows.

Para eliminar esta limitación, el primer paso es crear un paquete *resources* relativo al proyecto, donde se almacenarán todos los recursos que la aplicación demanda, entre ellos, las métricas y crawlers. El siguiente paso es modificar, en la presente clase, la ruta para hacer referencia a dos nuevas carpetas que se han creado en dicho paquete, metrics y crawlers, con sus ficheros JAR correspondientes.

4.3.2 Paquete presentation

Como se ha mencionado, hasta el momento, Atenea 3.0 trabaja con *Struts 2* para el desarrollo de la capa de presentación. La decisión de migrar de *Struts 2* a *Spring*, hace que este paquete *presentation* sea el que sufre el cambio más drástico de toda la aplicación.

Ambos frameworks trabajan sobre una arquitectura Modelo-Vista-Controlador (MVC), pero difieren en muchos aspectos en el momento de la implementación.

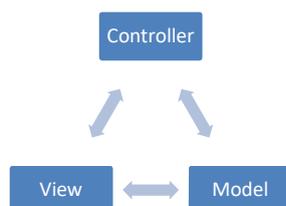


Imagen 12. Patrón MVC

4.3.2.1 Struts 2

En Atenea 3.0 el paquete presentación se encuentra dividido en diferentes subpaquetes categorizados según su funcionalidad.

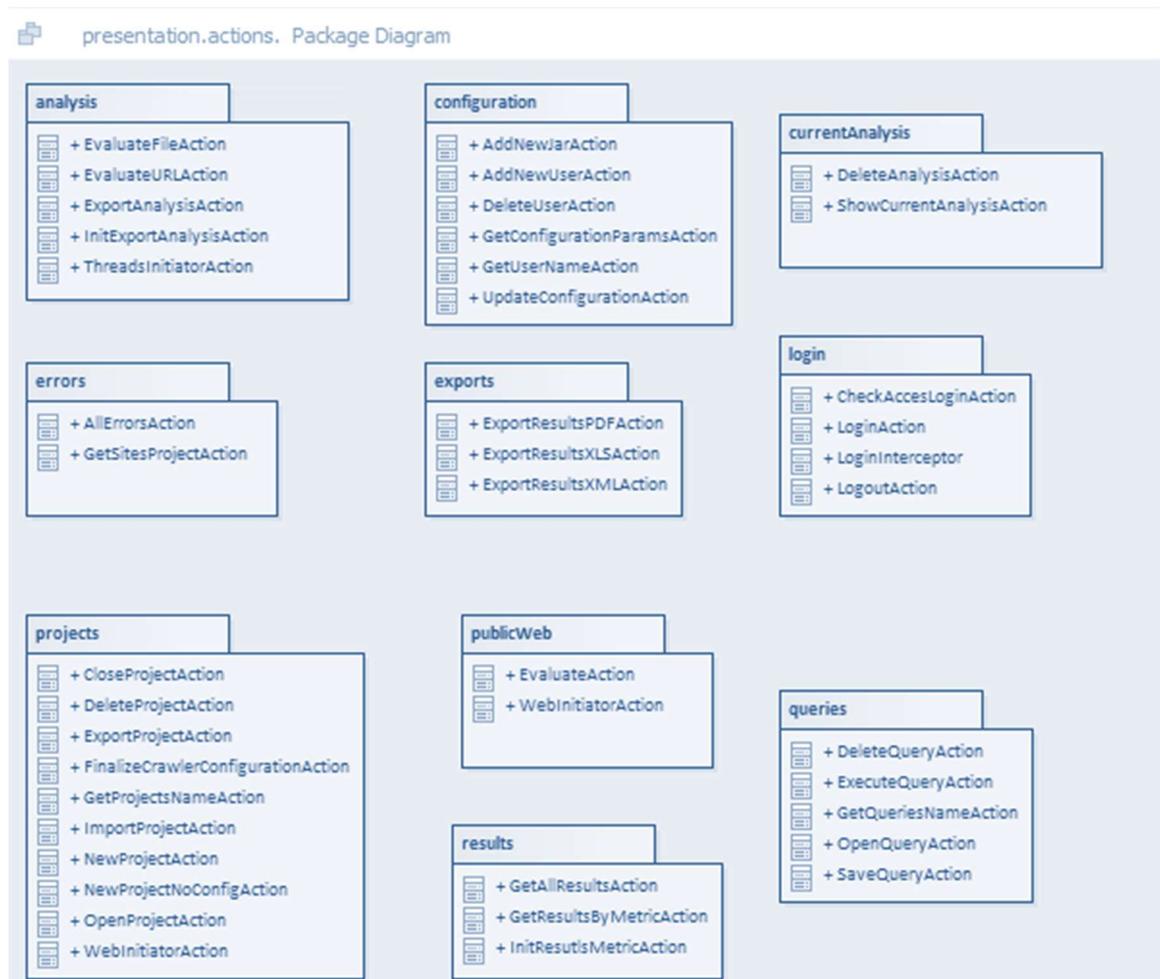


Imagen 13. Paquete presentation Atenea 3.0

Cada uno de estos subpaquetes está formado por distintas clases que representan las diferentes funcionalidades que el usuario puede realizar en el sistema.

Todas estas clases heredan de una clase abstracta, *ActionSupport*, que junto a un fichero XML, permiten la navegabilidad del sistema. También implementan una clase, *ServletRequestAware*, que permite recuperar el contexto de la aplicación.

El fichero XML no solo sirve para dar navegabilidad al sistema, sino que permite declarar los denominados *interceptors* que se ejecutan antes y después de la llamada a estas acciones para recoger y devolver datos a la vista, entre otras cosas.

Para poder recoger la información, estas clases, están dotadas de una serie de atributos correspondientes a la información que el usuario introduce a través de una vista determinada. Será cada una de estas acciones la que se encargue de validar los campos introducidos por el usuario y redirigir a una nueva vista de error.

Para devolver o mostrar información al usuario, se deben crear unas acciones previas que recojan la información que se desea mostrar y, ayudándose de la sesión, presentarlas por pantalla al usuario.

4.3.2.2 Spring

La disposición de paquetes en *Spring* difiere ligeramente de la anteriormente comentada, ya que además de un paquete *Controller*, subdividido en diferentes paquetes categorizados según su función, cuenta con nuevos paquetes formados por clases que servirán para recoger la información del usuario y validarla.

Spring, principalmente, trabaja mediante el uso de anotaciones.

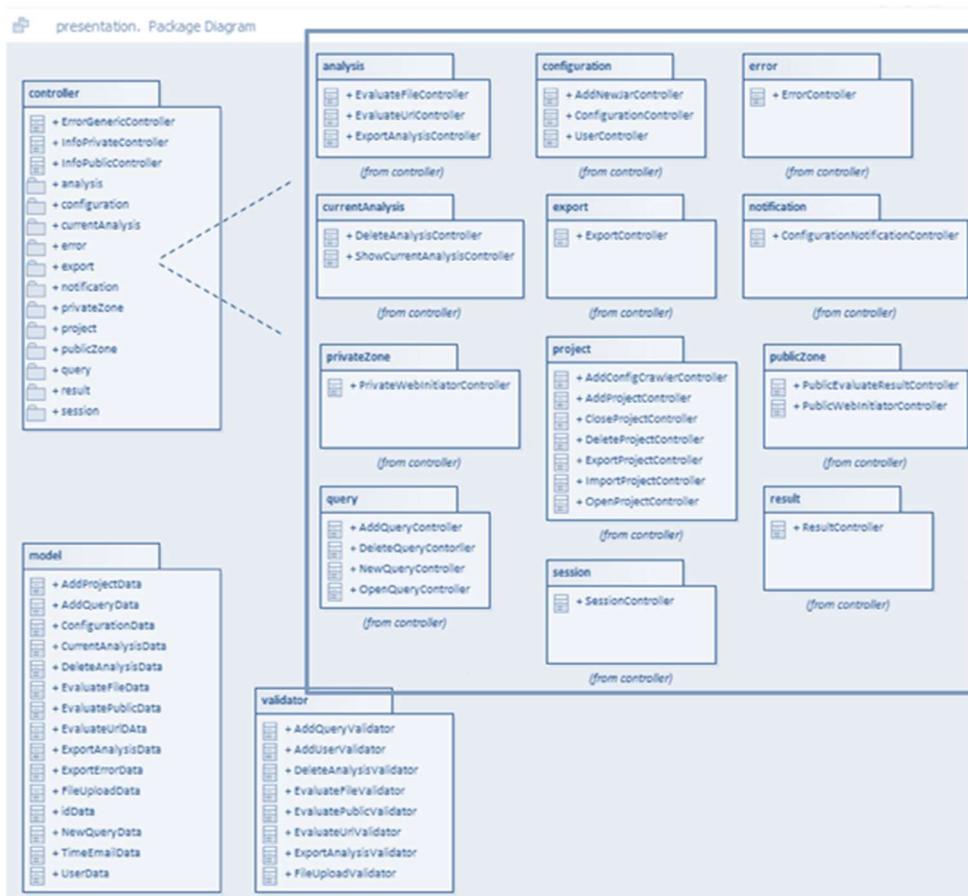


Imagen 14. Paquete presentation Atenea 4.0

Para trabajar con Spring, por cada acción que se quiera llevar a cabo, es necesario crear una nueva clase y anotarla con `@Controller`. En Spring, los controladores son los encargados de proporcionar el acceso a las distintas funcionalidades de la aplicación, son capaces de interpretar los datos introducidos por los usuarios y transformarlos en un modelo.

Cada una de estas clases contará, normalmente, con dos métodos anotados con `@GetMapping` y `@PostMapping`, recibiendo un parámetro que corresponderá con la URL a mapear y devolviendo la dirección de la JSP a la que se dirigirán al terminar sus correspondientes acciones. De esta forma se obtiene la navegabilidad del sistema.

Para recoger y devolver información de la vista, se hace uso de la anotación `@ModelAttribute`. Todo método con esta anotación se invocará siempre antes que cualquier `@<HTTPMETHOD>Mapping`, con el fin de preparar el modelo que va a ser utilizado durante la acción.

El modelo trabaja con objetos, es decir, se crea un modelo asociado a la capa de presentación, que servirá para recoger los datos introducidos por el usuario a través de las vistas, así como para presentar los datos en estas.

Existen dos objetos con unas características particulares, que nos permiten acceder al contexto de la acción que se está realizando. En primer lugar, el objeto `session` nos permite acceder al contexto de la aplicación y, análogamente, el objeto `request`, se utiliza para tener acceso al contexto de cada petición. Para acceder a estos objetos desde un controlador, simplemente tenemos que declararlos como parámetros del método que los quiera utilizar.

Otras de las funcionalidades necesarias en la capa de presentación, es la validación de los datos procedentes de los usuarios, puesto que estos pueden ser o no correctos. Se pueden diferenciar dos tipos de validaciones:

- Las validaciones más sencillas, se realizan directamente sobre los atributos del modelo asociado a la capa de presentación. Pueden ser, por ejemplo, `@NotNull`, `@NotEmpty`, etc.
- En el caso de validaciones más complejas, donde sea necesario especificar la lógica de validación, se deben crear clases de validación propias que implementen la interfaz `Validator` y sobrescriban el método `validate` adecuadamente. Estos validadores son utilizados en los controladores para validar el modelo.

Los errores procedentes de las validaciones se muestran por pantalla indicando qué campo falló y por qué, gracias a un componente de Spring.

4.3.2.3 Comparativa

	STRUTS 2	SPRING
FUNCIONALIDAD	Clases que extienden <code>ActionSupport</code> .	Clases anotadas con <code>@Controller</code> .
NAVEGABILIDAD	Fichero XML.	Anotaciones <code>@GetMapping</code> y <code>@PostMapping</code> .
RECOGER INFORMACIÓN	Las clases <code>Action</code> cuentan con atributos específicos para recoger la información procedente de la vista.	Los controladores trabajan con un nuevo modelo asociado a la capa de presentación para recoger la información. Método anotado con <code>@ModelAttribute</code> .
DEVOLVER INFORMACIÓN	Nuevas clases <code>Action</code> que a través del contexto muestran una información concreta mediante una vista.	Método anotado con <code>@ModelAttribute</code> .

	STRUTS 2	SPRING
VALIDACIONES	Las clases Action comprueban que la información sea correcta. Si no lo es, muestran un mensaje de error en otra vista diferente y terminan la acción.	Se crean unas clases específicas para validar la información procedente del usuario. Si la información es incorrecta, los mensajes se muestran en la misma vista facilitando al usuario su corrección.

Tabla 9. Comparativa Struts 2 y Spring

El cambio comenzó de forma paulatina, es decir, se decidió comenzar migrando una única funcionalidad, en este caso, la evaluación pública de un sitio web, ya que se trataba de un ciclo cerrado que permitiría probar y verificar su correcto funcionamiento. De igual forma, se van a mostrar los cambios realizados, mediante un diagrama de clases, con el fin de mostrar todas las comparaciones citadas.

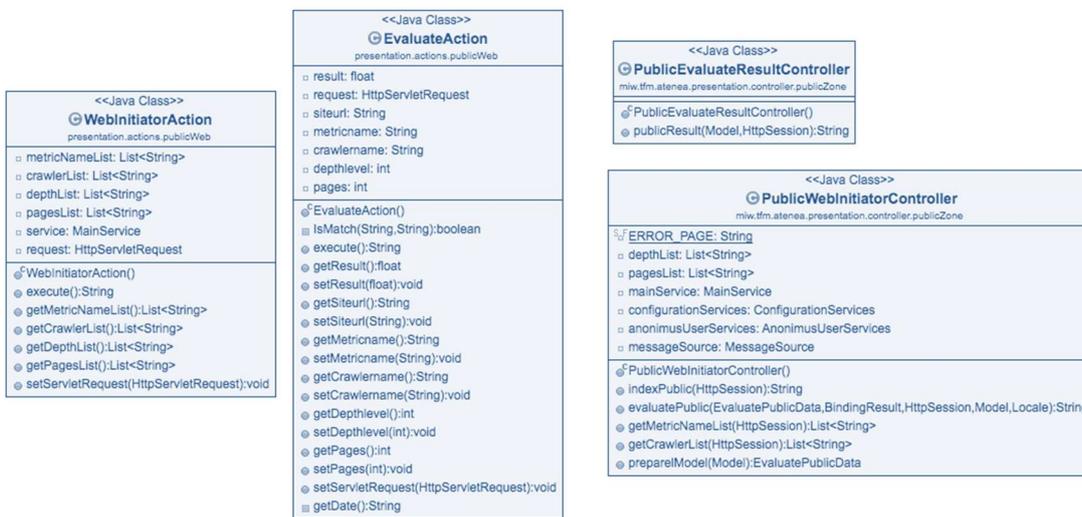


Imagen 15. Diagrama de clases comparativa Struts 2 y Spring

Todas las modificaciones se han ido propagando hasta migrar por completo la aplicación de *Struts 2* a *Spring*, transformando todas y cada una de las funciones individuales, que en conjunto conforman una aplicación que permite evaluar cuantitativamente, tanto la usabilidad como la accesibilidad de otras aplicaciones web.

4.3.2.4 JSP

Para crear las vistas en *Spring* se siguen utilizando las JSP que se utilizaban con *Struts 2*, intentando mantener, en la medida de lo posible, el formato. De esta forma se asegura que la curva de aprendizaje del colectivo que ya trabajaba con la aplicación se mantenga.

Se modifican las librerías de etiquetas que proporcionan los elementos de formulario, así como mensajes de error, etc. Para ajustarse a las necesidades de *Spring*.

4.3.3 Paquete model

En este paquete se encuentran las entidades y clases que representan el modelo de dominio de la aplicación. Se trata de simples clases POJO, es decir, clases con atributos que representan su estado, carecen de comportamiento y mantienen relaciones entre sí para ayudar al correcto funcionamiento de la aplicación. Representan elementos del mundo real, que, relacionados y tratados de una determinada manera, contribuyen a crear una solución para un problema en forma de programa.

Tras analizar el modelo sobre el que trabaja Atenea 3.0, se decide realizar una serie de cambios, atendiendo tanto a los atributos como a las relaciones de las clases que lo conforman. Los cambios de los atributos son simples, puesto que únicamente se trata de cambios relacionados con el tipo de estos.

La decisión de realizar este cambio fue tomada al ver que todos los atributos eran tratados como meros campos de texto, en vez de utilizar los tipos que Java proporciona para cada dato, una práctica que está totalmente desaconsejada. Por ejemplo, los campos relativos a fechas u horas se estaban tratando como *String* y se ha decidido convertirlos en campos de tipo *LocalDate* o *LocalDateTime*, la nueva API que Java proporciona para el tratamiento de fechas. De igual manera, se ha llevado cabo con el resto de los atributos.

En cuanto a las relaciones se realizan leves modificaciones ya que, bajo mi punto de vista, algunas de estas no tienen sentido natural o incluso llegan a ser redundantes.

Cabe destacar que las relaciones que comunican las diferentes clases del modelo se realizan a través de un atributo que hace las veces de identificador, y no mediante los propios objetos en sí. Para explicar esta situación lo mejor es ayudarse de una representación en forma de diagrama mostrando el antes y el después.

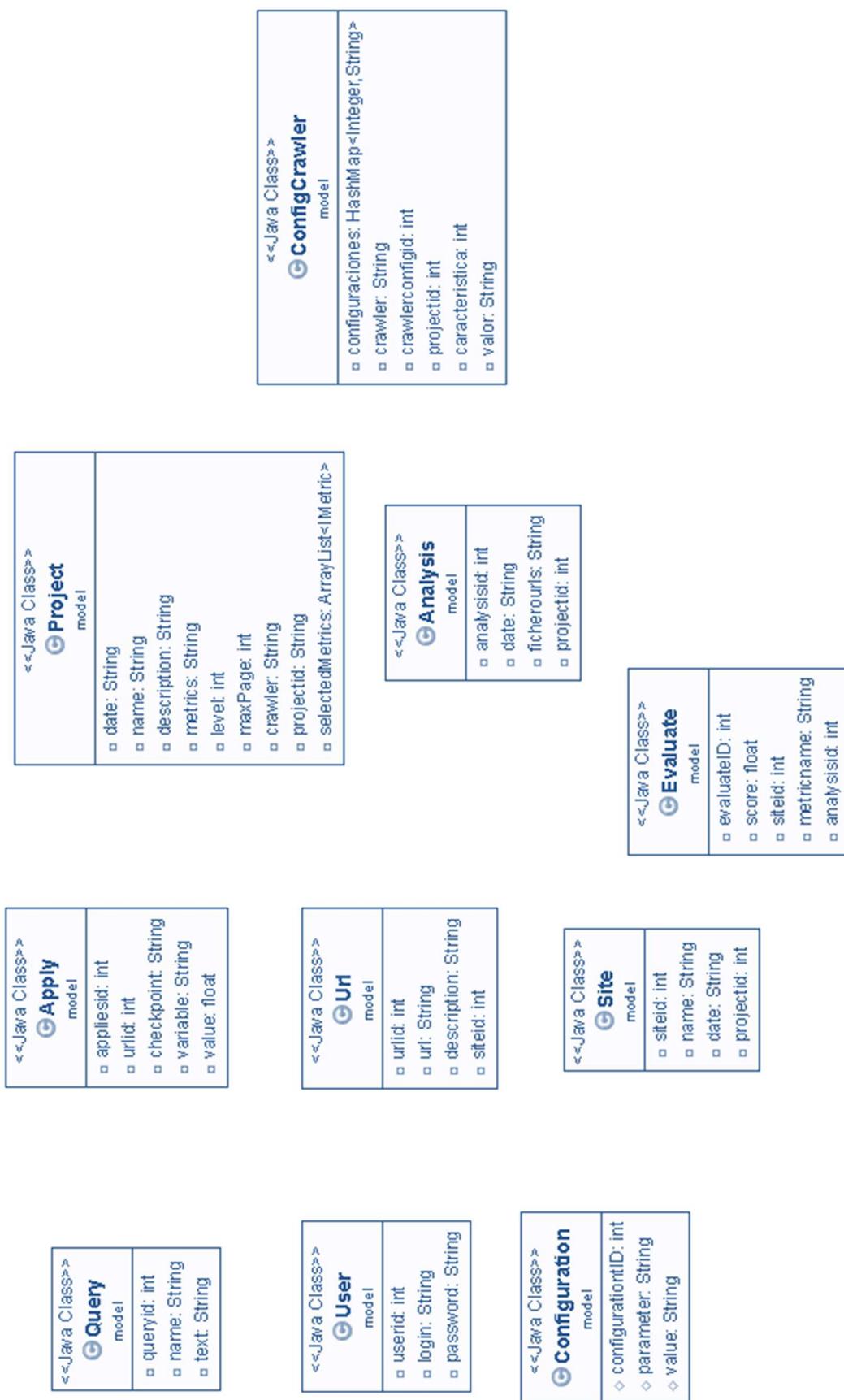


Imagen 16. Modelo de dominio Atenea 3.0

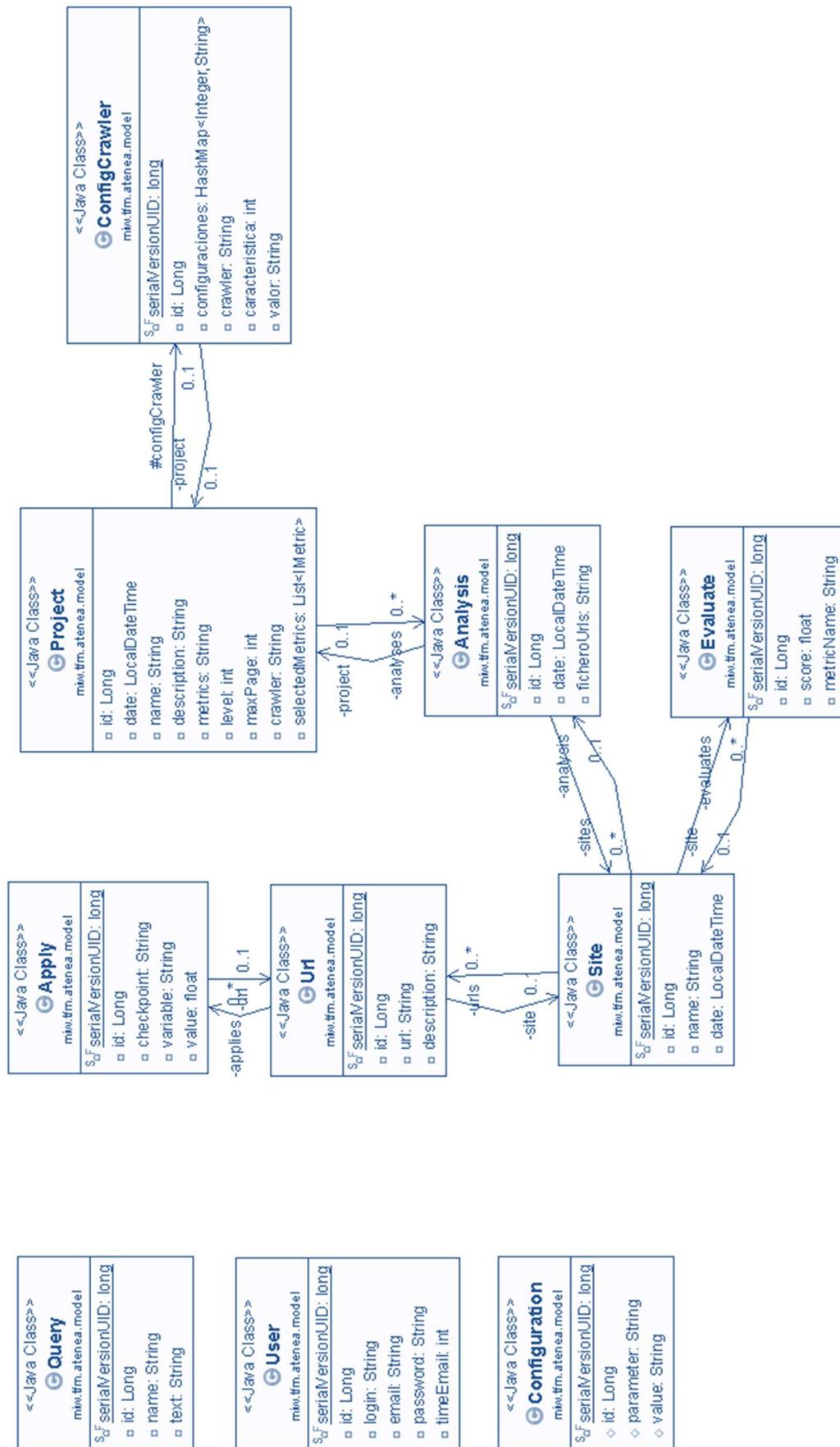


Imagen 17. Modelo de dominio Atenea 4.0

Cabe destacar que se eliminan las siguientes relaciones formadas por *Site-Project* y *Evaluate-Analysis*, por considerarlas redundantes. Además, con las nuevas relaciones añadidas resulta más fácil acceder a la información de ciertos objetos pasando por otros.

Los cambios realizados presentan una serie de mejoras que permiten trabajar con una mayor facilidad y poder implementar un sistema mejorado.

	ATENEA 3.0	ATENEA 4.0
RELACIONES IDENTIFICADAS	x	✓
RELACIONES REDUNDANTES	✓	x
TIPOS REPRESENTATIVOS	x	✓

Tabla 10. Comparativa modelo de dominio

Como se menciona anteriormente, Atenea 3.0 trabaja con *Hibernate* que facilita el mapeo de atributos entre una base de datos relacional y el modelo de dominio de una aplicación. Por lo tanto, cualquier cambio en el modelo de dominio repercute directamente en el modelo entidad-relación de la base de datos.

Para que una clase pueda ser mapeada debe seguir las siguientes restricciones:

- Deben tener un constructor público sin ningún argumento.
- Para cada propiedad que queramos persistir debe haber un conjunto de métodos getter y setter asociados.
- Implementar la interfaz *Serializable*.

Hasta este momento, para identificar qué clases se deben mapear, se utilizan archivos declarativos XML. En Atenea 4.0, estos ficheros XML, se reemplazan por anotaciones.

Como Atenea 3.0 carecía de relaciones a mapear, es normal que, tras la transformación, las clases cuenten con más atributos y las anotaciones que reemplazan a los ficheros XML resulten más tediosas de desarrollar.

Finalmente, cabe destacar que los cambios en el modelo de dominio trascienden a la capa de presentación, persistencia y negocio, por lo que, es necesario ir realizando las pertinentes modificaciones hasta adaptar el sistema a este nuevo modelo de dominio.

4.3.3.1 Diagrama Entidad-Relación

A continuación, se muestra el diagrama Entidad-Relación generado tras las modificaciones pertinentes.

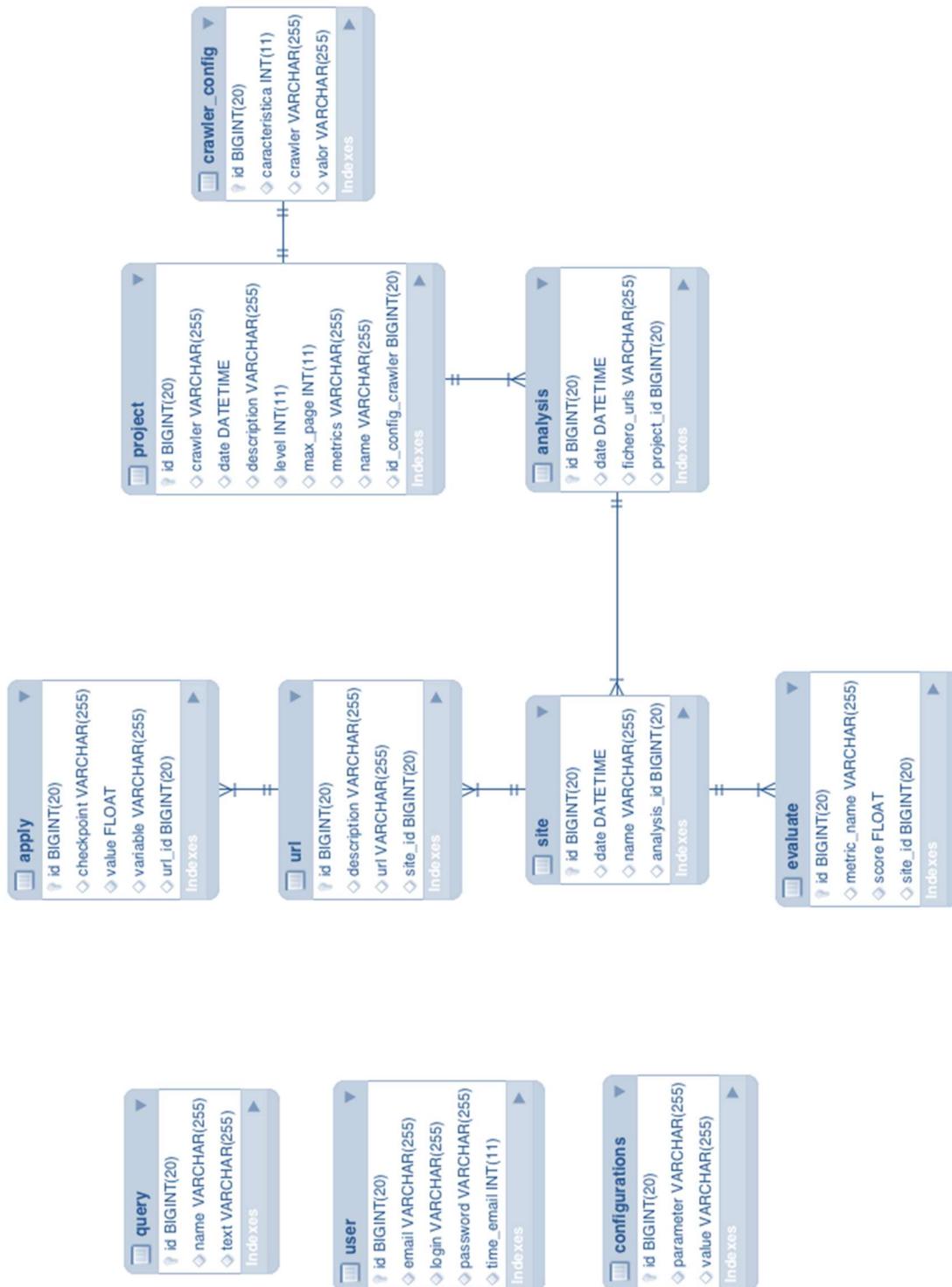


Tabla 11. Modelo E-R Atenea 4.0

Paquete dao

En este paquete se encuentran todas las clases relacionadas con la persistencia del sistema, es decir, aquellas clases que permiten recoger y almacenar datos de una base de datos.

Tanto en Atenea 3.0 como en Atenea 4.0 se mantiene el patrón DAO, permitiendo así, que el acceso a los datos necesarios para el correcto funcionamiento de la aplicación sea totalmente independiente del motor de base de datos utilizado. Por esto, la estructura de este paquete se mantiene en ambas aplicaciones.

El patrón DAO trabaja de la siguiente manera: se crea una interfaz con las acciones que se desean realizar contra la base de datos por cada objeto del modelo de domino. Por cada una de estas interfaces se crea una clase asociada que implementa cada uno de sus métodos. Es en estos métodos donde se realizan las consultas propias a la base de datos.

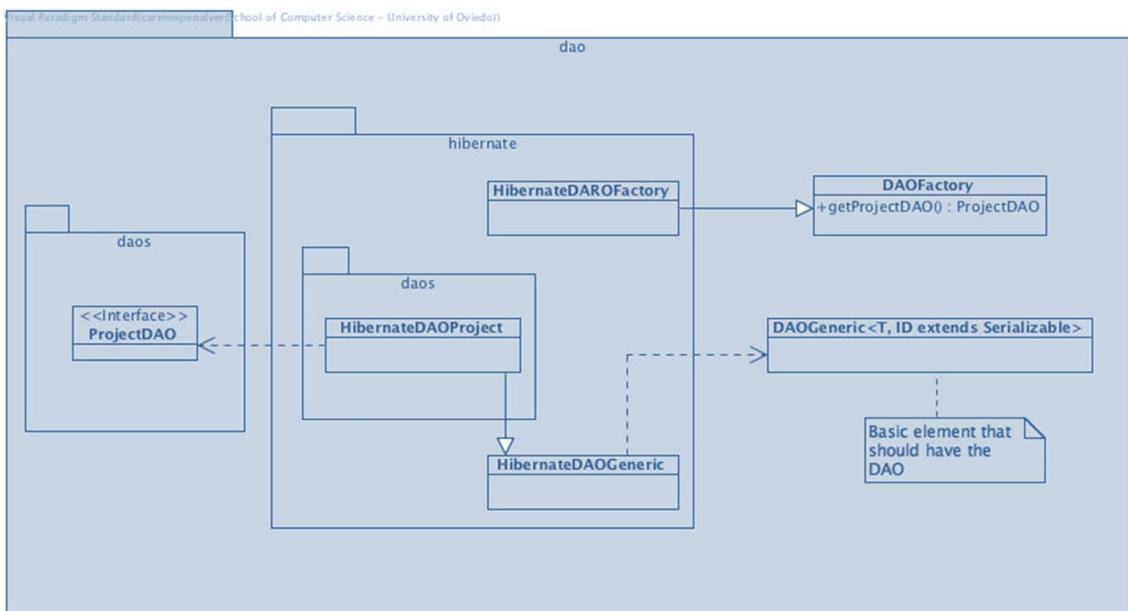


Imagen 18. Diagrama de paquete DAO

Además, las clases encargadas de realizar las consultas a la base de datos heredan de una clase que implementa una serie de métodos genéricos como son: añadir, eliminar, modificar y recuperar información. En el capítulo de análisis se estudiará este patrón de arquitectura en más detalle.

Cabe destacar que, aunque la estructura se mantenga, tanto las consultas como la manera de acceder la base de datos de Atenea 3.0, está totalmente deprecada y desactualizada, por esto, se realizan las pertinentes transformaciones. En Atenea 3.0, para el mapeo objeto-relación se utiliza la versión 3 de la herramienta *Hibernate* y, a lo largo de estos años, ha evolucionado a su versión 5, lo que implica la necesidad de realizar una serie de cambios.

En Atenea 4.0 se modifican y actualizan todas y cada una de las consultas que se realizan a la base de datos. Se crean las nuevas consultas siguiendo las nuevas directrices impuestas por el lenguaje JPQL y se extraen todas a un fichero orm.xml, para mantenerlas juntas y organizadas, evitando así llenar las clases con estas sentencias. Además, el nuevo cambio de modelo

beneficia y propicia la eliminación de muchas consultas que carecen de significado, así como la creación de nuevas consultas.

También se cambia, tanto la manera de enviar parámetros a la base de datos, como la manera de recoger dichos resultados, ya que, en general, se modifica la manera de acceder a la base de datos.

Hasta el momento, es decir, en Atenea 3.0, se trabaja con el objeto *SessionFactory* para comunicarse con la base de datos y realizar las transacciones pertinentes. *Hibernate* obliga a tener siempre disponible una referencia a este objeto, con el fin de que cualquier clase pueda tener siempre acceso al objeto *Session* y, por lo tanto, a todas las funcionalidades de esta herramienta de mapeo.

La principal causa que justifica el cambio de *SessionFactory* para utilizar *EntityManagerFactory*, no es por las ventajas que este pueda proporcionar, sino por su popularidad en estos momentos y el desuso de su opositor. La principal ventaja que aporta utilizar *EntityManager* es que se encarga de abrir y cerrar las transacciones, por lo que el programador solo se preocupa por implementar la lógica de negocio despreocupándose en la totalidad de las transacciones. En este caso, y con el fin de mantener una arquitectura sólida, que pueda mudar a lo largo del tiempo sin mucho esfuerzo por parte del programador, se prescinde de esta característica.

En ambas aplicaciones, la configuración necesaria para comunicarse con la base de datos se sigue obteniendo a través de ficheros declarativos XML, aunque con la presencia de algunos cambios para adaptarse a las necesidades de la nueva tecnología utilizada.

Tanto *SessionFactory* como *EntityManagerFactory*, mediante introspección, son capaces de reconocer estos ficheros declarativos, que deberán recibir un nombre determinado.

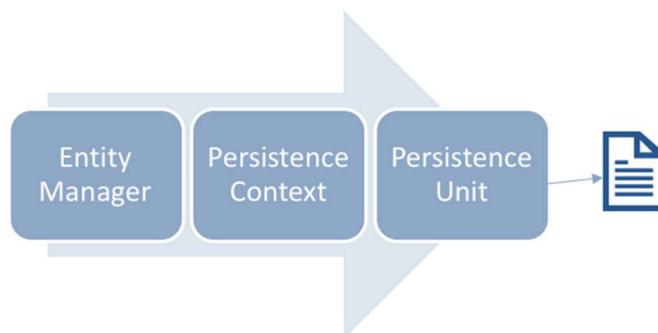


Imagen 19. Persistence manager

4.3.4 Paquete utilities

Este paquete contiene una serie de clases de utilidad necesarias para la aplicación, entre estas clases, se encuentra la función para manejar los resultados o generar un fichero de Log, etc.

Otra función, encapsulada en este paquete, es la generación de un fichero de evaluaciones que permite llevar un seguimiento de los análisis que han tenido lugar en la aplicación, ya sea a través de la zona pública, con un usuario sin registrar o desde la zona privada con un usuario registrado y autenticado. Este fichero recibe el nombre de "evaluations.txt".

En Atenea 3.0, este fichero “evaluations.txt”, tiene que estar situado en el directorio “C:\” de Windows para que la aplicación pueda escribir en él.

Como se ha mencionado anteriormente, Atenea 4.0 cuenta con una carpeta relativa, donde se almacenan este tipo de recursos. Para hacer referencia a la ruta de esta nueva carpeta, es necesario modificar la clase *LogFile*, y permitir así, que el proyecto deje constancia de todos y cada uno de los análisis ejecutados en este nuevo e independiente directorio.

4.3.5 Paquete comands

En este paquete se encapsulan una serie de clases que hacen de intermediario entre la capa de persistencia y la capa de negocio, es decir, se utilizan para realizar las pertinentes invocaciones desde las clases del paquete *services* a las clases del paquete *dao*.

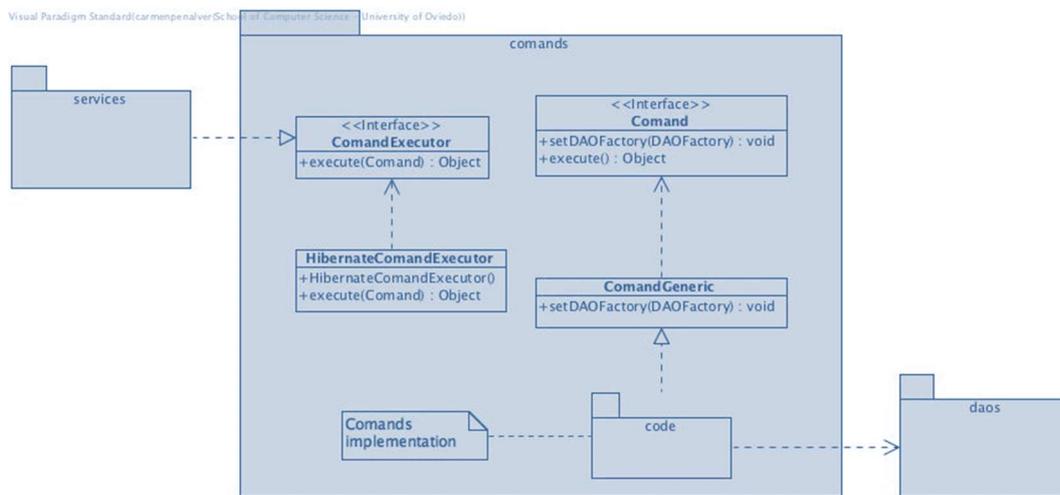


Imagen 20. Diagrama de paquete comands

El cambio del modelo de domino, así como la nueva forma de acceder al motor de base de datos, se traducen en la necesidad de realizar una serie de leves transformaciones en las clases que componen este paquete, con el fin de ajustarte al nuevo contexto del proyecto.

En Atenea 3.0, es la clase *HibernateComandExecutor* la encargada de instanciar el objeto *Session*, a través de *SessionFactoryManager*, para comunicarse con la base de datos y realizar todas las transacciones necesarias. Por lo que es necesario modificar dicha clase para que obtenga una instancia del objeto *EntityManger* mediante *EntityManagerFactory*. *EntityManager* proporciona la nueva forma de acceder a los métodos necesarios para poder interactuar con la base de datos.

El nuevo modelo de domino propicia la eliminación de muchas clases innecesarias existentes en el paquete *code*, así como la creación de otras muchas, con el fin de realizar nuevas acciones. Estas acciones, principalmente, consisten en búsquedas para realizar validaciones, filtrados o mostrar nueva información.

4.3.6 Paquete services

El paquete *services*, junto con los demás no mencionados hasta el momento, componen la lógica de la aplicación, es decir, constituyen la capa de negocio.

El estándar ANS/IEEE Std 1471-2000 (Association, IEEE-Standards 2000) concluye que: “La arquitectura de software se define, por la práctica recomendada, como la organización fundamental de un sistema, definida por sus componentes, las relaciones entre dichos componentes y con el entorno, y los principios que rigen su diseño y evolución.”

Gracias a la correcta definición de la arquitectura que rige este proyecto, y a pesar de todos los cambios producidos y comentados a lo largo del presente capítulo, este paquete no sufre grandes modificaciones.

Esto pone de manifiesto la importancia de crear un proyecto partiendo de un estudio previo y la definición de una arquitectura consolidada y formada por patrones que permiten definir componentes independientes. De esta manera, a la hora de producirse un cambio, como en este caso de la capa de presentación, los cambios no son muy trascendentes.

Sin embargo, cabe destacar que los cambios realizados en el modelo de domino desencadenan leves modificaciones en este paquete. Además, al haber introducido las relaciones lógicas que comunican los objetos de dicho modelo, se puede hacer uso del atributo *cascade* que proporciona *Hibernate*. Este atributo proporciona la capacidad de indicar cómo debe trabajar *Hibernate* con las clases relacionadas cuando realizamos alguna acción con la clase principal.

Hasta ahora en Atenea 3.0 cada vez que surgía la necesidad de añadir, eliminar o actualizar un objeto de la base de datos, se creaba una nueva transacción por cada uno de ellos y por sus objetos relacionados. En este paquete del nuevo proyecto, Atenea 4.0, se crean los objetos y sus relaciones en la capa de negocio y se reproducen en la base de datos solo y exclusivamente cuando se han creado todos los objetos relacionados a uno principal, es decir, se realiza una única transacción a la base de datos, lo que evita que la base de datos quede en un estado inconsistente.

Por ejemplo, cuando se crea un nuevo *Site* es necesario añadirle sus objetos *Url* asociados. En Atenea 3.0 se realiza una inserción tanto para añadir el *Site* como para añadir todos sus objetos *Url*. Ahora, el objeto *Site* cuenta con un listado de objetos *Url* y, en la capa de negocio, se añaden los enlaces de un sitio a esta lista, de esta forma solamente es necesario hacer una única inserción para añadir el *Site* y sus *Urls*. De esta manera o se persiste el objeto *Site* completo o no se persiste nada, pero en ningún momento, la base de datos contará con un registro *Site* sin ninguna *Url* asociada, es decir, no habrá inconsistencias.

Estas acciones se van propagando por todo el paquete *services* hasta que el sistema vuelva a funcionar correctamente con esta nueva adición.

Se prueba el sistema para comprobar que, a pesar de todos los cambios, este sigue funcionando correctamente. A lo largo de estas comprobaciones o pruebas, se encuentran algunos problemas de funcionalidad cuyas soluciones se comentan a continuación.

4.3.6.1 *Clase ErrorServicesImplemented*

Esta clase cuenta con una serie de métodos que implementan la funcionalidad necesaria para mostrar los errores manifestados en la ejecución de los análisis.

Según los requisitos funcionales, el sistema debe mostrar los errores según una métrica dada. En Atenea 3.0, este requisito no se cumple, ya que, a pesar de escoger la métrica, el sistema muestra los errores de todas las métricas. Incluso en algunas ocasiones esta funcionalidad, al completo, presenta errores.

Por todo esto, se llevan a cabo los cambios pertinentes hasta que la lógica asociada a los errores funciona correctamente.

4.3.6.2 *Clase ExportServicesImplemented*

Esta clase se encarga de los servicios asociados a la exportación en diferentes formatos tanto de errores como de resultados. Los formatos en los que se exportan los datos son los siguientes:

- XML
- PDF
- Excel

Probando el sistema me percaté que, en la exportación a Excel exporta únicamente la primera columna, incluso aunque la información a descargar tenga varias columnas.

Al igual que en la clase anterior, se depura la funcionalidad hasta encontrar el error y arreglarlo para que la exportación se desempeñe correctamente.

4.4 Modificaciones externas

Para su correcto funcionamiento, Atenea hace uso de programas externos. Entre estos, se encuentran los archivos JAR que constituyen las métricas y crawlers que contribuyen a la evaluación cuantitativa de sitios web, así como una aplicación web secundaria que permite la configuración, si existe, de estos últimos.

Como es lógico, los cambios realizados en la aplicación se traducen en la necesidad de modificar estas aplicaciones externas, con el fin ajustarlas a dichos cambios y que el programa funcione en su totalidad.

A continuación, se comentarán y desarrollarán todas las actualizaciones que sufren estas aplicaciones externas.

4.4.1 Archivos JAR

Como se mencionó con anterioridad, la integración de Maven en el proyecto da lugar a algunos problemas que se manifiestan tanto en las métricas como en los crawlers que se pueden añadir y utilizar en Atenea. Esto se debe a la estrecha relación que existe entre ellos, ya que presentan una dependencia real y bidireccional.

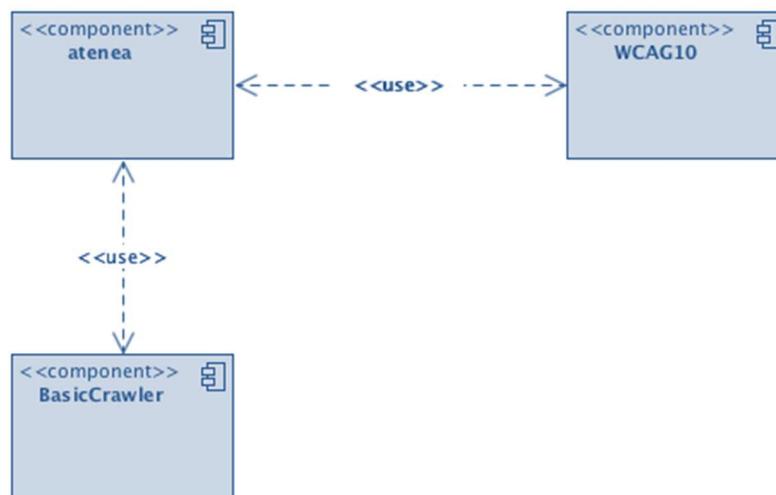


Imagen 21. Dependencia entre Atenea y JARs

Al crear Atenea 4.0 como un nuevo proyecto de tipo Maven se modifica la ruta de los paquetes del sistema, por lo que es necesario volver a generar todas y cada una de las métricas, así como los crawlers existentes, para organizar los pertinentes imports.

Realizando estas acciones, me percaté de que las métricas presentan la tendencia de replicar ciertas clases completas del proyecto, aunque las puedan obtener de la relación que presenta con él.

Una métrica (genérica) está formada por una o más métricas, así como uno o más checkpoints. Toda métrica hereda de la clase `MetricModel` de Atenea y, análogamente, los checkpoints heredan de la clase `CheckPointModel`. Hasta ahora, había una copia de estas clases en cada

métrica, con el fin de declarar la normativa que usan, esta práctica resulta totalmente desaconsejada y errónea, por lo que se decide arreglar dicho error. Por esto, *MetricModel* y *CheckpointModel* se han modificado para que el constructor de ambas clases acepte la normativa como parámetro y evitar así este duplicado totalmente erróneo e innecesario.

4.4.2 Web para configurar los crawlers

Un crawler o araña web es un pequeño programa diseñado para atravesar la red de una manera mecánica y automatizada. Estas arañas web se emplean, principalmente, para recopilar información de sitios web y poder así indexar sus contenidos.

El funcionamiento básico de una araña web o crawler es que a partir de uno o más enlaces iniciales, denominados semillas, se van descargando, de forma local, sus páginas web asociadas. Esto se realiza con la finalidad de buscar, dentro de ellas, nuevos enlaces. Cada nuevo enlace encontrado se almacena en una lista para, posteriormente, volver a visitarla y así recursivamente.

Por lo tanto, en Atenea, se utiliza este tipo de programas para analizar una determinada web y un conjunto de sus enlaces asociados. Además, el sistema permite añadir nuevos tipos de crawlers en tiempo de ejecución con o sin configuración. Con esta función y la posibilidad de que un crawler contenga valores de configuración, aparece un gran inconveniente, ya que no es posible integrar la interfaz de usuario asociada a estos posibles valores en un fichero JAR externo.

Por esto, en Atenea 3.0, la solución adoptada fue utilizar una aplicación web complementaria que aportará la JSP necesaria para cubrir los valores de configuración asociados a un crawler en caso de contenerla. Esto es, la aplicación principal pone a disposición del usuario un enlace a aplicación de configuración, intentando que todo parezca interconectado. Ambas aplicaciones trabajan sobre una la base de datos que hace las veces de punto común entre ellas.

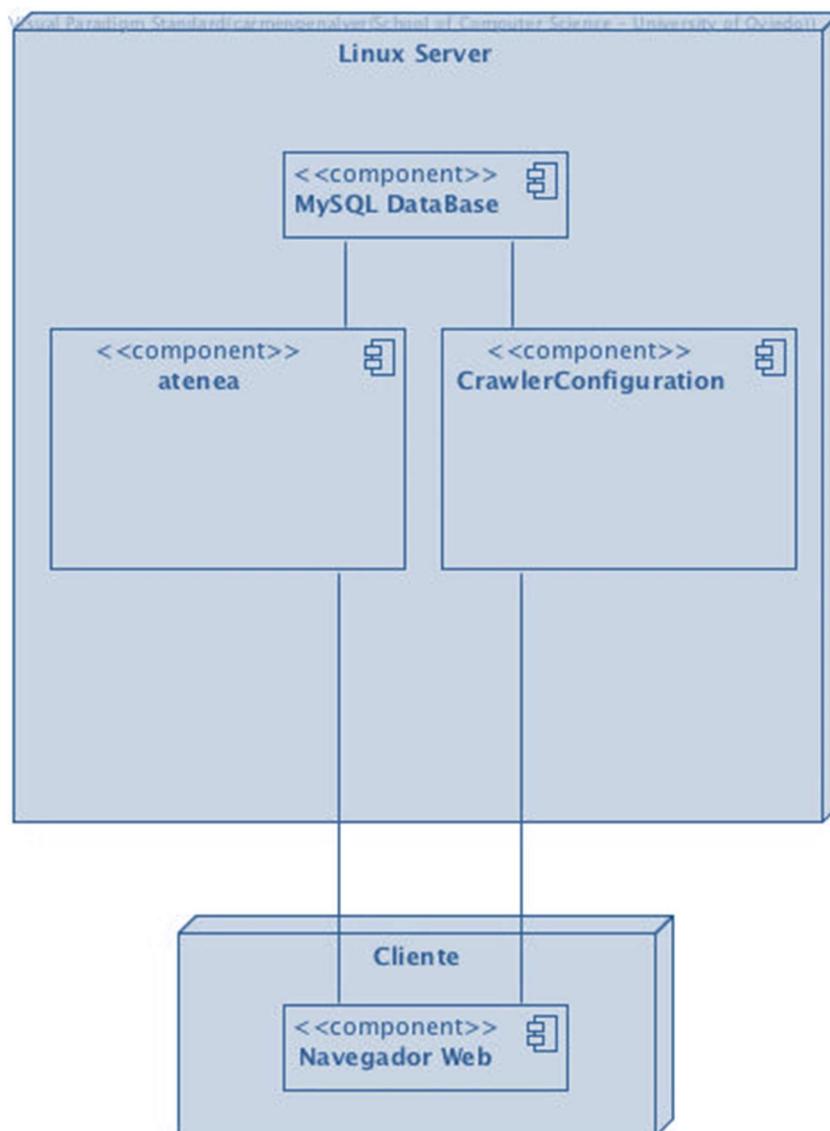


Imagen 22. Diagrama web complementaria configuración de crawler

Aunque esta no es la mejor solución permite añadir diferentes crawlers a la aplicación sin la necesidad de modificar al núcleo lógico de la misma.

En Atenea 4.0 se trabaja y estudia este problema y, finalmente, no se encuentra una mejor solución, por lo que se decide mantener esta infraestructura.

Ahora Atenea 4.0 utiliza como framework web *Spring*, mientras que la aplicación de configuración sigue utilizando el framework *Struts 2*. A priori, ambas aplicaciones podrían utilizarse sin necesidad de migrar esta segunda aplicación, pero aparece el inconveniente del cambio de modelo de dominio. Se aprovecha la necesidad de realizar las transformaciones pertinentes que exige el cambio de modelo de dominio, para migrar esta segunda aplicación complementaria.

La aplicación que permite configurar un crawler mantiene una estructura de paquetes similar a la conforma Atenea. Con las nuevas relaciones del modelo entidad-relación, es necesario cambiar los *services* y los *daos* a crear.

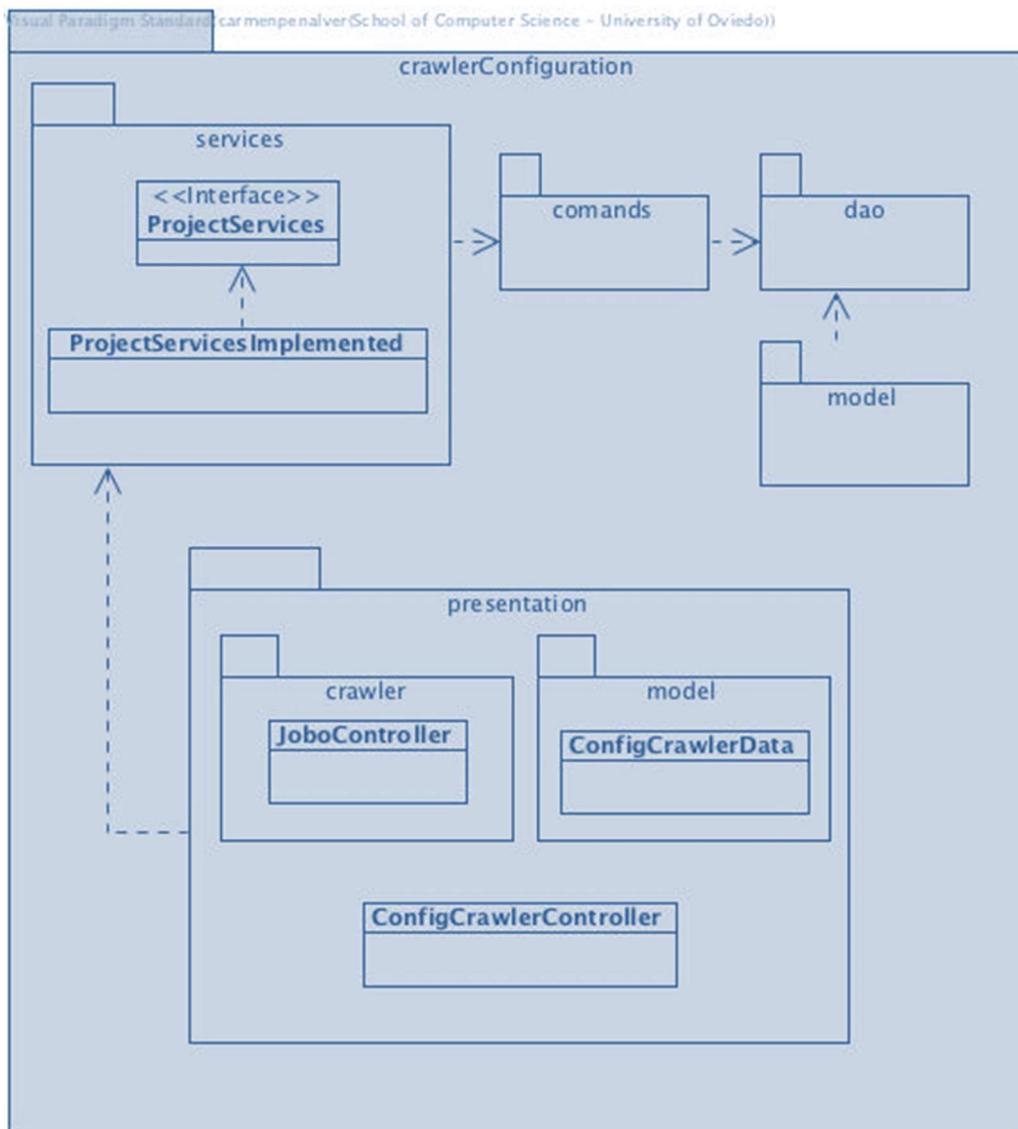


Imagen 23. Diagrama de paquetes CrawlerConfig

4.5 Adiciones al proyecto

Además de todas las modificaciones realizadas en Atenea 3.0 para evolucionar a la actual Atenea 4.0 y comentadas en los apartados previos, se realizan una serie de ampliaciones, con la finalidad de mejorar algunos aspectos y faltas en este proyecto.

4.5.1 Internacionalización

Toda aplicación seria debe tener externalizadas todas las cadenas de texto utilizadas, en este caso, en los ficheros JSP que conformar la interfaz de usuario de la aplicación.

Para mantener organizadas todas las cadenas de texto que se usan en Atenea 4.0, es necesario declarar un *bean* específico que *Spring* proporciona, denominado **ResourceBundles**, en un fichero declarativo XML. En *Spring*, un *bean* es un objeto configurado e instanciado en su contenedor y que puede ser accedido desde cualquier punto de la aplicación.

El siguiente paso es crear en la carpeta local *resources* un fichero *messages.properties* que contenga un conjunto de claves-valor, atendiendo a los textos de todas las JSP que utiliza la aplicación para su correcto funcionamiento. Este fichero debe recibir este nombre concreto para que el *bean* **ResourceBundles** encuentre sus mensajes, mediante introspección.

Finalmente, es necesario sustituir todos los textos de las JSP por mensajes de *Spring* haciendo defenecía a la clave correspondiente al valor que pertenece a ese lugar.

Por ejemplo, en el fichero *messages.properties* se añade la siguiente clave-valor: `evaluate=Evaluate` y en la JSP correspondiente se sustituye la cadena `Evaluate` por `<spring:message code="evaluate"/>`.

Aprovechando esto, se realiza la internacionalización de la aplicación, es decir, se traduce la aplicación para poder visualizarla en dos idiomas: inglés y español. Para ello, se crean dos nuevos ficheros de *properties*, cuyo nombre debe terminar por el código ISO del idioma correspondiente. Se hace uso de un nuevo *bean* específico de *Spring*, para recoger el idioma del navegador del cliente y mostrarle Atenea en su lenguaje por defecto.

En la capa de presentación se utilizan los objetos *MessageSource* y *Locale* para acceder a estos ficheros y poder mostrar los mensajes de verificación o error.

4.5.2 Aspecto de listas y “cargando”

En Atenea 4.0, se introduce una nueva forma de mostrar la información. Esto es, se modifica el aspecto con el que se visualizan las tablas existentes en la aplicación.

Estas tablas permiten, al usuario, realizar las siguientes acciones:

- Realizar búsquedas.
- Pagar los resultados.

- Ordenar los resultados numérica o alfabéticamente.
- Filtrar resultados por un parámetro dado.
- Mostrar más o menos información.

Para introducir esta novedad, se utilizan las DataTables de jQuery. Cabe destacar que también se utiliza esta biblioteca de JavaScript con el fin de mostrar mensajes “Cargando”, mientras se evalúan diferentes páginas webs, ayudándose de hojas de estilo.

4.5.3 Servicio de notificaciones

La aplicación Atenea ofrece la gran ventaja de realizar varios análisis de forma simultánea y en los que el usuario puede olvidarse de ellos, por completo, hasta su finalización. Esto es, La interacción con el usuario se produce al comienzo del análisis y, si este no desea consultar el progreso de dichos análisis, ya no es necesario que se preocupe más hasta el momento de su finalización para consultar el resultado.

Al finalizar Atenea 3.0 se plantea esta deficiencia como una posible ampliación, para que el usuario se despreocupe aún más de los análisis, es decir, se sugiere la idea de contar con un sistema de notificaciones, ya sea a través de mensajes SMS, una aplicación móvil o mensajes electrónicos.

Atenea 4.0 incluye un sistema de avisos por correo electrónico, que notifican al usuario exclusivamente cuando este realiza un análisis desde la zona privada y mediante un fichero que contenga un conjunto de enlaces a analizar.

Es necesario añadir al objeto User de la aplicación un nuevo atributo que indique el correo electrónico de cada uno y poder enviar a esa dirección estas notificaciones. También se añade un nuevo atributo que persiste en la base de datos la frecuencia con la que un usuario quiere recibir avisos sobre el progreso de un análisis. Se añade una nueva sección, en Atene4.0, para que este valor de configuración pueda ser modificado por el usuario a través de la interfaz de la aplicación.

Para llevar a cabo esta función, se desarrolla un nuevo servicio en la capa de negocio que implementa una serie de métodos que realizan las siguientes operaciones:

- Enviar un mensaje al inicio de cada análisis privado de un fichero de enlaces. Este correo electrónico inicial proporciona la información básica del proyecto, el conjunto de métricas escogidas para realizar el análisis, así como el listado de enlaces a analizar.
- Operación que implementa un *Timer* que se ejecuta periódicamente atendiendo al valor con el cual el usuario autenticado desea recibir avisos. En el correo de este cuerpo se muestra el porcentaje analizado, así como los enlaces que han terminado y los que se encuentran en ejecución.
- Enviar un mensaje al finalizar cada análisis privado de un fichero de enlaces. Este correo electrónico avisa el final de un análisis y que ya se pueden ver los resultados obtenidos. Además, este correo muestra la información básica del proyecto, el

conjunto de métricas escogidas para realizar el análisis, así como el listado de enlaces a analizar.

Se utiliza el objeto Locale para que estos mensajes electrónicos traduzcan su contenido, en tiempo de ejecución, según el idioma principal que escoja el usuario.

Finalmente, ha sido necesario crear una cuenta de correo electrónico asociada a Atenea para hacer las veces de remitente. Este correo tiene como dirección miw.atenea@gmail.com y su contraseña es **miwAt3n3A**.

4.5.4 Control de sesiones

Esta funcionalidad se contempla como una ampliación, puesto que la transformación es de un tamaño considerable y, además, se introduce una mayor seguridad, mediante el uso de un componente específico de *Spring*.

El proceso seguido por este componente, denominado *Spring Security*, se corresponde con el siguiente diagrama.

Spring Security es un framework que permitirá gestionar todo lo relativo a la seguridad de nuestra aplicación web, desde el protocolo de seguridad, hasta los roles que necesitan los usuarios para acceder a los diferentes recursos de la aplicación.

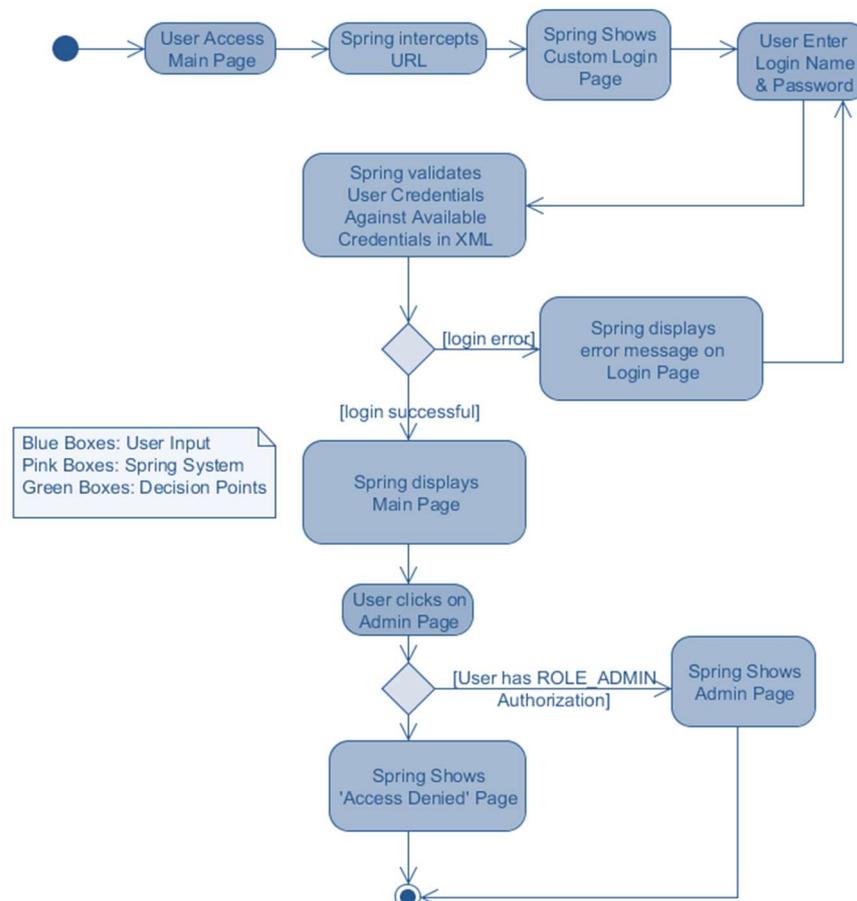


Imagen 24. Ciclo Spring Security

Para el correcto funcionamiento de este proceso es necesario añadir una serie de clases que, en su conjunto, permitirán a los usuarios el acceso a algunos recursos de la aplicación, así como la prohibición a otros. Por ejemplo, cualquier usuario puede acceder a la zona pública de la aplicación, es decir, tiene la capacidad de realizar un análisis público y estudiar sus resultados, pero se le prohibirá el acceso a cualquier otra pantalla de la misma, sin haberse autenticado previamente.

Entre estos recursos, se encuentran algunos ficheros que se mantienen accesible a todos los usuarios y a toda la aplicación continuamente como por ejemplo las hojas de estilo, las imágenes o el código JavaScript.

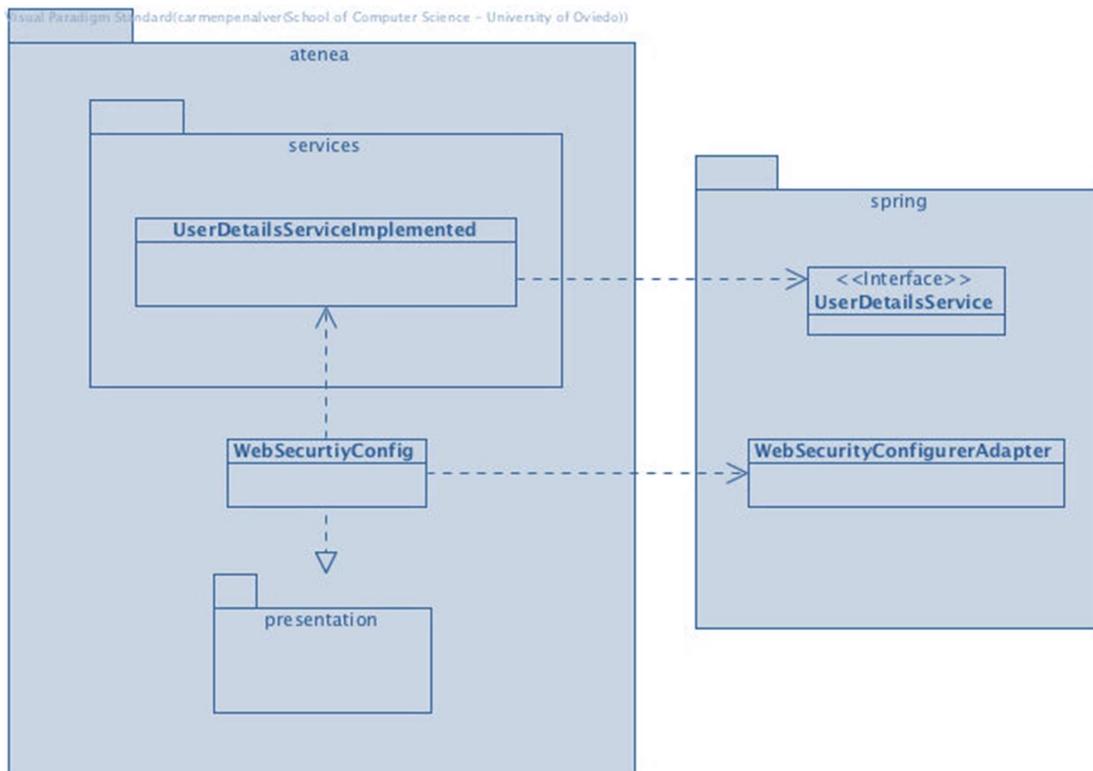


Imagen 25. Diagrama Spring Security

Se crea una clase que servirá para declarar la configuración de seguridad por la cual el sistema se registrará. Esta clase recibe el nombre de `WebSecurityConfig` y cuenta con dos métodos principales. El primer método indica los siguientes aspectos:

- Los recursos públicos.
- Los recursos accesibles por un usuario con un determinado rol, en este caso, ADMIN.
- La dirección URL del inicio de sesión, así como el identificador que deben tener los campos de usuario y contraseña de este mismo formulario.
- La dirección URL a la que se redirige al usuario en caso de que el inicio de sesión sea satisfactorio, así como aquella a la que debe dirigirse en caso de error de autenticación.

El segundo método se encarga de realizar la llamada a una clase, *UserDetailServiceImplemented*, que implementa una interfaz proporcionada por Spring Security, pasándole el usuario y la contraseña introducidos. Será esta clase la encargada de comprobar si dichas credenciales son correctas, consultando la base de datos. Además, se crea un nuevo usuario específico de Spring con el rol necesario para permitir el acceso a los recursos restringidos y poder acceder a él desde cualquier lugar de la aplicación gracias al objeto *Authentication*.

4.6 Listado completo de aportaciones

En este apartado se realiza un resumen, en forma de lista, enumerando todas las aportaciones y modificaciones realizadas al proyecto Atenea.

- Creación de un proyecto de tipo Maven con la tecnología Spring Boot.
- Actualización de la versión de las librerías externas de las que depende el proyecto.
- Creación de una carpeta de recursos relativa al proyecto, con el fin de eliminar cualquier referencia al directorio “C:\” de Windows.
- Modificación del framework web MVC utilizado para el desarrollo de la capa de presentación, es decir, se realiza la migración de Struts 2 a Spring.
- Transformación de las JSP con la finalidad de adaptarse al nuevo framework web.
- Modificación del modelo de domino: se añaden las relaciones naturales entre los objetos, se cambia el tipo de algunos atributos del modelo y, finalmente, se añaden nuevos atributos.
- Se actualiza la versión 3 de Hibernate a la versión 5.
- Actualización de las consultas a la base de datos, así como su organización en un fichero XML.
- Se cambia el objeto que permite realizar las transacciones con la base de datos y, por lo tanto, la forma de pasarle parámetros y recoger los resultados.
- Las transformaciones en el modelo de domino propagan cambios en el paquete *comands*, que une la capa de negocio con la capa de persistencia.
- Cambios y arreglos en la capa de negocio, es decir, en la lógica de la aplicación.
- Eliminación de código duplicado.
- Adaptación de las métricas y crawlers externos.
- Migración a Spring de la aplicación complementaria que permite la configuración de los crawlers.
- Validaciones mejoradas.
- Externalización e internacionalización de los textos de la aplicación.
- Interfaz de usuario enriquecido.
- Agregación de un servicio de notificaciones.
- Accesos a la aplicación con mayor seguridad.

Capítulo 5. Análisis

A lo largo del presente apartado se desarrollará la especificación de los requisitos y la documentación asociada al análisis de la aplicación, que se utilizará como referente para elaborar su posterior diseño.

5.1 Definición del Sistema

5.1.1 Determinación del Alcance del Sistema

Como se ha reflejado, anteriormente, este proyecto consiste en la migración y reingeniería de un sistema existente, es decir, el alcance del proyecto viene condicionado por la necesidad de conservar la funcionalidad original de esta aplicación, con la posibilidad de implementar nuevas funcionalidades.

El sistema existente recibe el nombre de Atenea y consiste en una aplicación web desarrollada con el cometido de analizar aspectos tanto de accesibilidad como de usabilidad en páginas webs. Estos análisis se realizan para mejorar el diseño de las páginas analizadas según estos aspectos y, permitir así, que todos los usuarios puedan acceder en condiciones de igualdad a los contenidos.

Por todo esto, aunque, a lo largo del desarrollo del presente proyecto, Atenea sufra cambios o actualizaciones en su tecnología, estructura, modelo de domino, servidor, etc., deberá mantener su funcionalidad completa. Esto es, se permitirá realizar análisis públicos y privados de una o más páginas webs, ayudándose de las métricas y crawlers existentes en el sistema. Se proporcionará la opción de conocer el estado, traza y porcentaje de progreso de un análisis privado en curso.

Otra funcionalidad importante es la capacidad de gestionar nuevos proyectos a los que asociar diferentes análisis para, posteriormente, poder comprobar y exportar tanto sus resultados como sus errores.

El sistema permitirá añadir nuevas métricas de accesibilidad y usabilidad en tiempo de ejecución, así como nuevos crawlers.

También se deberá mantener la funcionalidad de gestionar consultas, usuarios y modificar la configuración básica de la aplicación.

Finalmente, hay que comentar que este nuevo sistema Atenea contará con la novedosa capacidad de enviar notificaciones a los usuarios con el estado de sus análisis privados.

5.2 Requisitos del Sistema

5.2.1 Obtención de los Requisitos del Sistema

A continuación, se desarrollará la lista de requisitos, tanto funcionales como no funcionales, que se tendrán en cuenta a la hora de desarrollar el sistema.

5.2.1.1 *Requisitos funcionales*

Los requisitos funcionales son, como su nombre indica, aquellos que describen el funcionamiento y estado del sistema, así como la interacción de éste con el entorno. Para describir con claridad estos requisitos, se organizarán en subgrupos atendiendo, tanto a su funcionalidad como a los stakeholders, que son aquellos usuarios finales interesados en el producto.

5.2.1.1.1 Autenticación de usuarios

RF1. El sistema debe permitir a los usuarios, previamente añadidos al mismo, autenticarse para acceder a la zona privada de la aplicación.

RF1.1. Para acceder a la zona privada de la aplicación, el sistema pedirá introducir el nombre de usuario y la contraseña.

RF1.2. Si los datos son correctos, el sistema permitirá al usuario el acceso a la zona privada de la aplicación.

RF1.3. Si los datos son incorrectos, el sistema mostrará un mensaje de error y volverá a pedir que se introduzcan ambos datos.

RF2. El sistema permitirá a los usuarios autenticados en el sistema la opción de cerrar sesión.

5.2.1.1.2 Análisis público

RF3. El sistema permitirá a cualquier usuario realizar un análisis público.

RF3.1. El sistema permitirá introducir una URL a analizar y esta deberá comenzar con el siguiente patrón: `http(s)://`

RF3.2. El sistema pondrá a disposición del usuario un listado de todas las métricas existentes en el sistema para poder escoger una de ellas.

RF3.3. El sistema pondrá a disposición del usuario un listado de todos los crawlers existentes en el sistema para poder escoger uno de ellos.

RF3.4. El sistema no permitirá que ninguno de estos campos este vacío.

RF3.5. Si los datos anteriores son válidos el sistema comenzará a analizar la URL introducida y mostrará un mensaje “cargando”.

RF3.6. Si los datos anteriores no son válidos, el sistema mostrará un mensaje de error y volverá a pedir que se introduzcan los datos de nuevo.

RF4. El sistema mostrará el resultado del análisis en una tabla con la siguiente información: URL del sitio analizado, métrica y crawler escogidos y, finalmente, la puntuación obtenida.

RF4.1. Si el análisis se ha realizado sin ningún inconveniente, la puntuación será mayor que cero (> 0).

RF4.2. Si ha ocurrido algún problema durante el análisis, o no se ha podido analizar la presente URL la puntuación obtenida será -1.

5.2.1.1.3 Gestión de usuarios

RF5. El sistema permitirá a un usuario, previamente registrado y autenticado, añadir nuevos usuarios para que accedan a la zona privada.

RF5.1. Para registrar un nuevo usuario, el sistema pedirá introducir los siguientes datos: nombre del usuario, correo electrónico, contraseña y confirmación de contraseña.

RF5.2. El sistema no permitirá añadir un nombre de usuario que ya esté en uso.

RF5.3. El sistema no permitirá añadir un correo electrónico que ya esté en uso y no siga la siguiente estructura: aaa@bbb.ccc

RF5.4. Tanto la contraseña como la confirmación de la contraseña deberán coincidir.

RF5.5. Si alguno de los campos no se rellena o no sigue la estructura correcta, el sistema mostrará un mensaje de error y no permitirá añadir el nuevo usuario al sistema.

RF5.6. Si todos los datos introducidos son correctos, el sistema verificará la operación mostrando un mensaje y añadiendo la información del nuevo usuario a la base de datos.

RF6. El sistema permitirá a un usuario, previamente registrado y autenticado, eliminar un usuario existente.

RF6.1. Para eliminar un usuario, el sistema mostrará un listado de los usuarios registrados en Atenea. En este listado, se visualizará el nombre de usuario.

RF6.2. El sistema permitirá escoger un usuario del listado para eliminarlo de la base de datos y, por consecuente, del sistema.

RF6.3. Si todo es correcto, el sistema eliminará dicho usuario y mostrará un mensaje comunicando que la acción ha sido realizada con éxito.

RF6.4. Si se da algún error, el sistema no eliminar el usuario y mostrará un mensaje de error.

RF6.5. El sistema permitirá volver a añadir a este usuario en cualquier momento.

5.2.1.1.4 Actualización de la configuración básica

- RF7. El sistema permitirá a un usuario, previamente registrado y autenticado, actualizar los parámetros de configuración básica que luego se utilizarán por defecto a la hora de analizar enlaces.
- RF7.1. El sistema permitirá modificar los siguientes campos: nivel de profundidad, número máximo de páginas a analizar y, finalmente, hilos por defecto utilizados en el análisis.
- RF7.2. El sistema mostrará cada uno de estos campos ya rellenos con los datos actuales registrados en el sistema para cada uno de ellos.
- RF7.3. El sistema no permitirá que ninguno de los campos esté vacío antes de realizar la acción de actualización.
- RF7.4. El sistema solo permitirá que los datos introducidos sean valores numéricos y positivos.
- RF7.5. El nivel de profundidad no podrá ser superior a 3.
- RF7.6. Si todo es correcto, el sistema actualizará los nuevos datos asociados a la configuración básica y mostrará un mensaje verificando la acción.
- RF7.7. Si se produce algún error o los datos no cumplen las directrices anteriores, el sistema no actualizará los datos y mostrará un mensaje de error.

5.2.1.1.5 Añadir una nueva métrica

- RF8. El sistema permitirá a un usuario, previamente registrado y autenticado, añadir una nueva métrica al sistema, para poder analizar tanto aspectos de usabilidad como accesibilidad.
- RF8.1. El sistema mostrará un campo que abrirá el sistema de ficheros del ordenador para poder seleccionar una métrica nueva.
- RF8.2. El sistema solo admitirá archivos con formato JAR.
- RF8.3. Si todo es correcto, el sistema almacenará la nueva métrica para, posteriormente, utilizarla. Se mostrará un mensaje verificando la acción.
- RF8.4. Si ocurre algún error al añadir la métrica, se mostrará un mensaje de error y no se añadirá ninguna métrica nueva al sistema.

5.2.1.1.6 Añadir un nuevo crawler

- RF9. El sistema permitirá a un usuario, previamente registrado y autenticado, añadir un nuevo tipo de crawler, para modificar la forma en la que se obtienen todos los enlaces indexados por una web.
- RF9.1. El sistema mostrará un campo, que abrirá el sistema de ficheros del ordenador para poder seleccionar un nuevo crawler.

RF9.2. El sistema solo admitirá archivos con formato JAR.

RF9.3. Si todo es correcto, el sistema almacenará el nuevo crawler, para posteriormente utilizarlo. Se mostrará un mensaje verificando la acción.

RF9.4. Si ocurre algún error al añadir el crawler, se mostrará un mensaje de error y no se incrementará el número de crawlers en el sistema.

5.2.1.1.7 Gestión de proyectos

RF10. El sistema permitirá a un usuario, previamente registrado y autenticado, añadir un nuevo proyecto al sistema si no hay un proyecto abierto en la sesión.

RF10.1. Para añadir un nuevo proyecto, el sistema pedirá introducir los siguientes datos: nombre del proyecto y una breve descripción del mismo.

RF10.2. El sistema pondrá a disposición del usuario un listado de todas las métricas disponibles en Atenea. El sistema permitirá escoger una o varias métricas asociadas al proyecto y utilizarlas en sus posteriores análisis.

RF10.3. El sistema mostrará un listado de los posibles niveles de profundidad que se han configurado previamente en el sistema. EL usuario deberá escoger para la creación del proyecto el nivel que desee.

RF10.4. El sistema mostrará un listado de números del 0 al número máximo de páginas que se han configurado previamente en el sistema. El usuario deberá escoger, para la creación del proyecto, el número máximo de páginas.

RF10.5. El sistema pondrá a disposición del usuario un listado de todos los crawlers disponibles en Atenea, para poder escoger uno de ellos.

RF10.5.1. Si el crawler cuenta con configuración, el sistema permitirá al usuario escoger si desea usar dicha configuración o no.

RF10.5.1.1. Si el usuario no desea configurar el crawler, el proyecto se creará sin una configuración asociada.

RF10.5.1.2. Si el usuario desea configurar el crawler, el sistema se conectará con otra aplicación web, que se encargará de pedir al usuario los datos necesarios para poder finalizar la configuración del crawler.

RF10.6. El sistema no permitirá finalizar la creación de un proyecto si alguno de los anteriores campos se encuentra vacío.

RF10.7. El sistema almacenará automáticamente la fecha en la que se creará el proyecto.

RF10.8. Si todos los datos son correctos, el sistema verificará la operación mostrando un mensaje y añadiendo la información asociada al nuevo proyecto en la base de datos.

RF10.8.1. Si se ha creado el proyecto con éxito, el sistema mantendrá dicho proyecto como el proyecto abierto en la sesión.

RF10.9. Si algún dato es incorrecto o se da algún problema inesperado, el sistema mostrará un mensaje de error y el proyecto no se persistirá en el sistema.

RF11. El sistema permitirá a un usuario, previamente registrado y autenticado, abrir un proyecto existente en el sistema. Solamente se podrá abrir un proyecto si no se ha abierto otro con anterioridad.

RF11.1. El sistema mostrará un listado de los proyectos existentes en el sistema con los siguientes datos: nombre y breve descripción del proyecto, fecha de creación, nivel de profundidad, número máximo de páginas, el listado de métricas y, finalmente, el crawler.

RF11.2. El sistema permitirá elegir un proyecto, de entre el listado, para ser abierto y proceder a realizar sus posteriores análisis.

RF11.3. El sistema permitirá al usuario ver en todo momento el proyecto que se encuentra abierto.

RF12. El sistema permitirá a un usuario, previamente registrado y autenticado, cerrar un proyecto que permanece abierto en la sesión desde cualquier punto de la aplicación.

RF13. El sistema permitirá a un usuario, previamente registrado y autenticado, eliminar un proyecto existente en el sistema. Solamente se podrá eliminar un proyecto si no hay un proyecto abierto en el sistema.

RF13.1. El sistema mostrará un listado de los proyectos existentes en el sistema con los siguientes datos: nombre y breve descripción del proyecto, fecha de creación, nivel de profundidad, número máximo de páginas, el listado de métricas y, finalmente, el crawler.

RF13.2. El sistema permitirá escoger un proyecto de entre el listado para eliminarlo del sistema y, por consecuente, eliminar toda la información y relaciones asociadas al mismo.

RF13.3. Si el proyecto se elimina con éxito, se mostrará un mensaje de verificación y el proyecto no existirá en el sistema.

RF13.4. Si ocurre algún problema en el momento de la eliminación del proyecto, se mostrará un mensaje de error y el proyecto se mantendrá en el sistema.

RF14. El sistema permitirá a un usuario, previamente registrado y autenticado, exportar la información asociada a un proyecto existente en el sistema. Solamente se podrá exportar la información de un proyecto si no existe un proyecto abierto en el momento de la exportación.

RF14.1. El sistema mostrará un listado de los proyectos existentes en el sistema con los siguientes datos: nombre y breve descripción del proyecto, fecha de creación, nivel de profundidad, número máximo de páginas, el listado de métricas y, finalmente, el crawler.

RF14.2. El sistema permitirá escoger un proyecto de entre el listado para posteriormente exportarlo.

RF14.3. La información del proyecto se exportará en un archivo con formato XML.

RF14.4. El archivo contendrá la siguiente información asociada al proyecto y estructura.

```

▼<Project>
  <Name value="Proyecto"/>
  <Description value="Proyecto"/>
  ▼<Metrics>
    <metric name="TE"/>
    <metric name="UWEM"/>
    <metric name="WAB *"/>
    <metric name="WAB"/>
  </Metrics>
  <Depth_level value="0"/>
  <Max_number_pages value="5"/>
  <Crawler value="Basic"/>
</Project>

```

Imagen 26. Estructura, información y formato al exportar un proyecto

RF14.5. Si el archivo se crea sin ningún inconveniente, el sistema descargará automáticamente el fichero XML en el ordenador del cliente.

RF14.6. Si se produce algún error a la hora de crear y exportar el fichero, el sistema mostrará un mensaje de error.

RF15. El sistema permitirá a un usuario, previamente registrado y autenticado, importar un proyecto en el sistema. El sistema prohibirá importar un proyecto si existe un proyecto abierto en el sistema.

RF15.1. El sistema solamente admitirá importar la información asociada a un proyecto que proceda de un fichero con formato XML.

RF15.2. El archivo contendrá el nombre, descripción, métrica o listado de métricas, nivel de profundidad, número máximo de páginas a analizar y, finalmente, el crawler.

RF15.3. El archivo deberá mantener la siguiente estructura.

```

▼<Project>
  <Name value="Proyecto"/>
  <Description value="Proyecto"/>
  ▼<Metrics>
    <metric name="TE"/>
    <metric name="UWEM"/>
    <metric name="WAB *"/>
    <metric name="WAB"/>
  </Metrics>
  <Depth_level value="0"/>
  <Max_number_pages value="5"/>
  <Crawler value="Basic"/>
</Project>

```

Imagen 27. Estructura, información y formato para importar un proyecto

RF15.4. Si el proyecto se importa con éxito, se mostrará un mensaje de verificación y el proyecto se persistirá en el sistema y se abrirá directamente.

RF15.5. Si ocurre algún problema en el momento de importar el proyecto, se mostrará un mensaje de error y el proyecto no se persistirá en el sistema.

RF16. En todos los listados asociados a la gestión de proyectos, el sistema no filtrará dichos proyectos por usuario. Es decir, todos los usuarios registrados y autenticados podrán ver todos los proyectos creados en el sistema hasta el momento.

5.2.1.1.8 Gestión de consultas

RF17. El sistema permitirá a un usuario, previamente registrado y autenticado, ejecutar una consulta sobre la base de datos asociada al sistema.

RF17.1. El sistema pondrá a disposición del usuario un listado de las tablas y campos accesibles por dicho usuario.

RF17.2. El sistema pedirá introducir una sentencia SQL válida.

RF17.3. Si la sentencia SQL introducida es válida y las tablas y sus campos se corresponden con la base de datos del sistema, se mostrará un listado de los resultados asociados a dicha consulta.

RF17.4. Si se produce algún error a la hora de ejecutar la sentencia introducida, el sistema no mostrará ningún resultado, pero si un mensaje mostrando el error ocurrido.

RF18. El sistema permitirá a un usuario, previamente registrado y autenticado, añadir consultas predeterminadas en el sistema, para poder utilizarlas en posteriores ocasiones.

RF18.1. El sistema pedirá introducir los siguientes datos: nombre de la consulta y la propia consulta SQL.

RF18.2. El sistema prohibirá añadir consultas que ya existan en el sistema, es decir, que con el mismo nombre o sentencia SQL haya alguna ya existente.

RF18.3. Si ambos datos son correctos, el sistema persistirá la nueva consulta en la base de datos y mostrará un mensaje de éxito.

RF18.4. Si algún dato es incorrecto, el sistema mostrará un mensaje de error y no añadirá la nueva consulta.

RF19. El sistema permitirá a un usuario, previamente registrado y autenticado, abrir y ejecutar una consulta existente en el sistema.

RF19.1. El sistema mostrará un listado con todas las consultas existentes en el sistema. Este listado contará con los siguientes datos: nombre de la consulta y sentencia SQL correspondiente.

RF19.2. El sistema permitirá escoger una consulta de entre el listado para poder ser ejecutada.

RF19.3. Si se ha escogido una consulta, el sistema mostrará el resultado obtenido de la misma.

RF19.4. Si no se ha escogido una consulta o se produce algún error, el sistema mostrará un mensaje de error.

RF20. El sistema permitirá a un usuario, previamente registrado y autenticado, eliminar una consulta existente en el sistema.

RF20.1. El sistema mostrará un listado con todas las consultas existentes en el sistema. Este listado contará con los siguientes datos: nombre de la consulta y sentencia SQL correspondiente.

RF20.2. El sistema permitirá escoger una consulta de entre el listado para poder ser eliminada.

RF20.3. Si se ha escogido una consulta, el sistema eliminará la consulta del sistema.

RF20.4. Si no se ha escogido una consulta o se produce algún error, el sistema mostrará un mensaje de error y la consulta seguirá en Atenea.

5.2.1.1.9 Gestión de análisis

RF21. El sistema da, a un usuario registrado y autenticado, la posibilidad de analizar una única dirección URL.

RF21.1. El sistema solamente permitirá analizar una URL si hay un proyecto abierto al que poder asociar dicho análisis.

RF21.2. El sistema pedirá introducir una URL a analizar que deberá comenzar por el siguiente patrón: http(s)://

RF21.3. El sistema pondrá a disposición del usuario un listado del número de hilos que el sistema permitirá utilizar para el análisis. En la primera posición aparecerá el hilo escogido por defecto en la configuración básica.

RF21.4. El sistema obtendrá el crawler, las métricas, el número máximo de páginas y el nivel de profundidad que necesita, para realizar el análisis del proyecto abierto y, por consecuente, su proyecto asociado.

RF21.5. Si todos los datos son correctos, se mostrará un mensaje “Cargando” y se comenzará a realizar el análisis de la URL introducida.

RF21.6. Si se produce algún error a la hora de analizar la URL, se mostrará un mensaje de error y se interrumpirá el proceso.

RF22. El sistema da, a un usuario registrado y autenticado, la posibilidad de analizar un conjunto de direcciones URL.

RF22.1. El sistema solamente permitirá analizar un fichero con varias URLs, si hay un proyecto abierto al que poder asociar dicho análisis.

RF22.2. El fichero deberá estar formado por un listado de URLs separadas por un retorno de carro.

- RF22.3. El sistema permitirá seleccionar el número de hilos concurrentes que serán utilizados a la hora de realizar el análisis. Dicho dato utilizará el número de hilos indicado en la configuración básica como valor por defecto.
- RF22.4. El sistema obtendrá el crawler, las métricas, el número máximo de páginas y el nivel de profundidad que necesita, para realizar el análisis del proyecto abierto y, por consecuente, su proyecto asociado.
- RF22.5. Si todos los datos son correctos, se mostrará un mensaje “Análisis en curso” y se comenzará a realizar el análisis de todas las URLs introducidas.
- RF22.6. Si se produce algún error a la hora de analizar la URLs, se mostrará un mensaje de error y se interrumpirá el proceso.
- RF23. El sistema permitirá a un usuario, previamente registrado y autenticado, consultar y exportar los resultados de un análisis procedente de un fichero de URLs.
- RF23.1. El sistema mostrará un listado de los análisis realizados desde un fichero de URLs, asociados a un proyecto abierto. Este listado contendrá la siguiente información: fecha y nombre del fichero.
- RF23.2. El sistema permitirá escoger un análisis de entre el listado.
- RF23.3. El sistema mostrará los resultados del análisis escogido. En este listado se visualizará el sitio web, la métrica y la puntuación obtenida.
- RF23.4. El sistema permitirá realizar un filtrado de estos resultados por métrica y sitio web.
- RF23.5. El sistema permitirá exportar los resultados asociados a un análisis desde fichero en los siguientes formatos:
- RF23.5.1. El sistema permitirá exportar los resultados a un fichero XML.
- RF23.5.2. El sistema permitirá exportar los resultados a un fichero PDF.
- RF23.5.3. El sistema permitirá exportar los resultados a un fichero XLS.
- RF23.6. Si no se han obtenido resultados para el análisis escogido, el sistema mostrará un mensaje comunicando esta falta.

5.2.1.1.10 Análisis en curso

- RF24. El sistema permitirá a un usuario, previamente registrado y autenticado, conocer el progreso de todos los análisis en curso. Para conocer el estado de los análisis en curso, el sistema debe contar con uno o más análisis en ejecución.
- RF24.1. El sistema mostrará un listado con los análisis en ejecución. Este listado contendrá la siguiente información: nombre del proyecto, fecha del análisis, porcentaje completado y usuario que inicio el análisis.
- RF24.2. El sistema permitirá actualizar el porcentaje de los análisis en ejecución.

- RF24.3. El sistema solamente mostrará los análisis cuyo porcentaje completado sea < 100%.
- RF24.4. Si no existe ningún análisis en curso, el sistema mostrará un mensaje comunicando dicha información.
- RF25. El sistema permitirá a un usuario, previamente registrado y autenticado, conocer la traza de un análisis en curso. Para conocer la traza de un análisis en curso, el sistema debe contar con uno o más análisis en ejecución.
- RF25.1. El sistema mostrará un listado con los análisis en ejecución. Este listado contendrá la siguiente información: nombre del proyecto, fecha del análisis, porcentaje completado y usuario que inicio el análisis.
- RF25.2. El sistema permitirá escoger un análisis de entre el listado para conocer su traza de ejecución. En esta traza se mostrará el listado de URLs a analizar según las métricas y un estado: TERMINADO o EN EJECUCIÓN.
- RF26. El sistema permitirá a un usuario, previamente registrado y autenticado, detener un análisis en curso. Para detener un análisis en curso, el sistema debe contar con uno o más análisis en ejecución.
- RF26.1. El sistema mostrará un listado con los análisis en ejecución. Este listado contendrá la siguiente información: nombre del proyecto, fecha del análisis, porcentaje completado y usuario que inicio el análisis.
- RF26.2. El sistema permitirá escoger un análisis de entre el listado para detenerlo.
- RF26.2.1. Antes de llevar a cabo la detención, el sistema obligará al usuario a confirmar la acción para evitar equivocaciones.
- RF26.3. El sistema almacenará los resultados de los enlaces analizados hasta el momento de la detención.
- RF26.4. Si el análisis se ha detenido correctamente, el sistema mostrará un mensaje verificando la acción.
- RF26.5. Si se da algún error durante la detención de análisis, el sistema mostrará un mensaje de error y el análisis seguirá en ejecución.

5.2.1.1.11 Mostrar resultados

- RF27. El sistema permitirá a un usuario, previamente registrado y autenticado, consultar y exportar los resultados asociados a un proyecto completo abierto. Esto es, se mostrarán los resultados de los análisis tanto de una única URL como de un fichero de ellas.
- RF27.1. El sistema mostrará los resultados de los análisis asociados a un proyecto. En este listado se visualizará el sitio web, la métrica y la puntuación obtenida.

RF27.2. El sistema permitirá realizar un filtrado de estos resultados por métrica y sitio web.

RF27.3. El sistema permitirá exportar los resultados asociados a un análisis desde fichero en los siguientes formatos:

RF27.3.1. El sistema permitirá exportar los resultados a un fichero XML.

RF27.3.2. El sistema permitirá exportar los resultados a un fichero PDF.

RF27.3.3. El sistema permitirá exportar los resultados a un fichero XLS.

RF27.4. Si no se han obtenido resultados para el análisis escogido, el sistema mostrará un mensaje comunicando esta falta.

5.2.1.1.12 Mostrar errores

RF28. El sistema permitirá a un usuario, previamente registrado y autenticado, consultar los errores encontrados según las métricas, a lo largo de los análisis. Para consultar los errores, es necesario tener un proyecto abierto en el sistema y haber ejecutado, al menos, un análisis.

RF28.1. El sistema mostrará un listado de las métricas del proyecto.

RF28.2. El sistema mostrará un listado de los enlaces que se han analizado en este proyecto.

RF28.3. El sistema permitirá escoger una métrica.

RF28.4. El sistema permitirá escoger uno o más enlaces.

RF28.5. El sistema mostrará los errores asociadas a la métrica y enlaces escogidos.

RF28.5.1. Si la métrica cuenta con un servicio de errores integrado, el sistema exportará el listado de errores en forma de archivo PDF.

RF28.5.2. Si la métrica no cuenta con un servicio de errores integrado, el sistema mostrará un listado de los errores. El sistema permitirá filtrar los resultados según los enlaces.

RF28.6. Si no hay errores para un proyecto abierto, se mostrará un listado vacío.

5.2.1.1.13 Sistema de avisos

RF29. El sistema permitirá a un usuario, previamente registrado y autenticado, escoger la periodicidad con la que desea recibir mensajes informativos sobre el análisis en curso.

RF29.1. El sistema proporcionará un listado con una serie de minutos predefinidos. La primera opción de la lista será “No deseo correos”.

RF29.2. El sistema permitirá escoger una periodicidad de entre el listado.

RF29.3. Si la elección es correcta, el sistema actualizará la nueva periodicidad al usuario autenticado en ese momento, y mostrará un mensaje de verificación.

RF29.4. Si se produce algún error, el sistema mostrará un mensaje de error y no añadirá la configuración escogida.

RF29.5. El sistema al crear un usuario, pondrá por defecto la periodicidad a cero, es decir, no se recibirán mensajes informativos.

RF30. El sistema enviará correos electrónicos informando sobre el estado de un análisis de varias URLs.

RF30.1. El sistema enviará un correo al inicio de cada análisis. Este mensaje contendrá la información del usuario en sesión, la información asociada al proyecto, así como los enlaces a analizar.

RF30.2. El sistema enviará un correo al finalizar el análisis. Este mensaje contendrá la información del usuario en sesión, la información asociada al proyecto, así como los enlaces a analizar.

RF30.3. El sistema comprobará si el usuario autenticado desea recibir correos electrónicos.

RF30.3.1. Si el usuario desea recibir correos electrónicos, el sistema consultará la dirección de correo electrónico y la periodicidad con la que el usuario autenticado desea recibir mensajes informativos y enviará el mensaje correspondiente. El mensaje estará formado por el porcentaje de análisis, así como por un listado de los sitios que ya se han analizado y los que se encuentran en ejecución.

RF30.3.2. Si el usuario no desea recibir correos, no recibirá mensajes periódicos.

5.2.1.1.14 Consultar ayuda

RF31. El sistema permitirá al usuario consultar la ayuda de la aplicación.

5.2.1.2 Requisitos no funcionales

Existen diversos procedimientos que desembocan en la extracción, tanto de requisitos funcionales como no funcionales. Uno de estos procedimientos está descrito por Ian Sommerville (Sommerville 2011) (Gamma, y otros 2009), un académico británico dedicado a la ingeniería de sistemas informáticos.

Según Sommerville un requisito no funcional es, como su nombre indica, un requisito que no se relaciona directamente con los servicios específicos que el sistema entrega a sus usuarios. Además, no sólo dio una definición, sino que también los clasifica siguiendo una jerarquía que se puede observar en la siguiente figura.

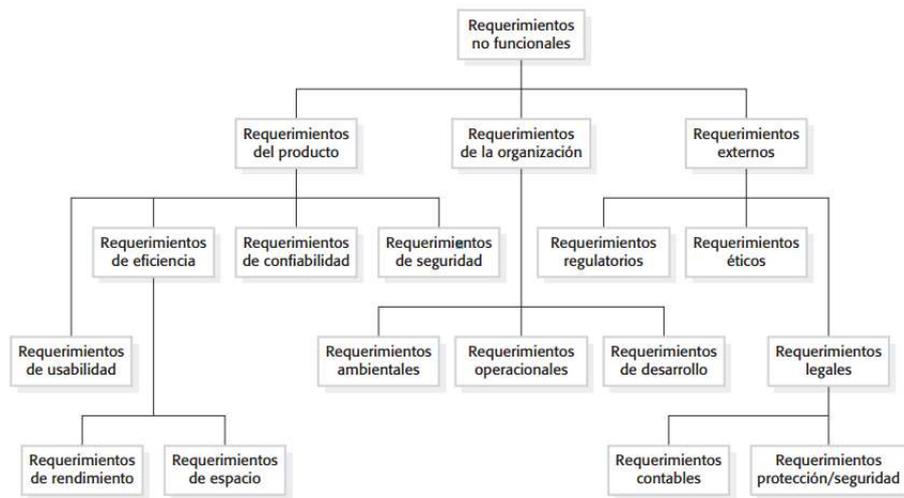


Imagen 28. Clasificación de los requisitos no funcionales según Ian Sommerville

5.2.1.2.1 Requisitos de productos

Estos requisitos derivan de las características requeridas del sistema y especifican restricciones en la ejecución del mismo.

RNF1. El sistema deberá contar con una interfaz de usuario similar a la de sus versiones anteriores, con el fin de mantener la curva de aprendizaje.

RNF2. El sistema debe ser capaz de recuperarse de los errores, así como de responder a los mismos con mensajes claros y concisos.

RNF3. La aplicación debe contar con un sistema de ayuda.

RNF4. El sistema debe seguir un diseño responsivo que permita adecuarlo a las distintas visualizaciones posibles.

RNF5. La aplicación web deberá poder visualizarse en los distintos navegadores disponibles como son: Google Chrome, Internet Explorer, Safari y Mozilla Firefox.

RNF6. La aplicación ha de ser usable.

RNF7. El sistema se deberá presentarse los siguientes idiomas: inglés y español.

5.2.1.2.2 Requisitos de organización

Estos requisitos derivan de las políticas y procedimientos existentes en la organización, tanto del cliente como en la del desarrollador.

RNF8. El sistema se desarrollará en distintos módulos independientes, para evitar que, en futuras modificaciones, sea necesario realizar cambios en todos ellos.

RNF9. La aplicación web alcanzará un nivel AA en cuanto a accesibilidad se refiere.

RNF10. El sistema se implementará utilizando patrones de diseño.

5.2.1.2.3 Requisitos externos

Estos requisitos hacen referencia a aspectos externos al sistema.

- RNF11. La aplicación proporcionará un sistema de encriptación para las contraseñas utilizadas.
- RNF12. Se deben tomar las medidas oportunas para que la aplicación se ejecute en un entorno seguro.

5.2.2 Identificación de Actores del Sistema

A continuación, se detalla la lista de personas interesadas en el proyecto o stakeholders, es decir, aquellas personas que van a ejercer alguna influencia sobre los resultados globales del proyecto.

5.2.2.1 *Usuario anónimo*

Un usuario anónimo puede realizar análisis públicos de diferentes sitios web indicando la métrica y el crawler que desee. El nivel de profundidad de estos análisis siempre será cero, es decir, solamente se podrá analizar la página principal. Además, al finalizar el análisis, el usuario anónimo podrá consultar el resultado del mismo.

También podrá consultar toda la información indexada desde la parte pública de la web como, por ejemplo, aspectos de accesibilidad o la información de contacto.

En caso de contar con una cuenta en el sistema, el usuario podrá iniciar sesión.

5.2.2.2 *Usuario registrado*

Un usuario autenticado y previamente registrado en el sistema será el encargado de realizar las siguientes funciones:

- Actualizar la configuración básica de la herramienta.
- Gestionar proyectos.
- Gestionar usuarios.
- Gestionar consultas a la base de datos.
- Realizar análisis de uno o más sitios webs.
- Consultar los resultados de los análisis ejecutados.
- Consultar los errores encontrados en los análisis ejecutados según una métrica dada.
- Consultar el porcentaje y la traza de los análisis en ejecución.

Finalmente, cabe destacar que un usuario, previamente registrado y autenticado, podrá añadir nuevas métricas y nuevos crawlers al sistema. Además, podrá recibir avisos a través de correo electrónicos al realizar análisis desde un fichero de URLs.

5.2.3 Especificación de Casos de Uso

A continuación se presentan, en forma de diagrama, la interacción de los distintos usuarios para con el sistema. Además se detalla, para cada una de estas acciones, una breve descripción.

5.2.3.1 Casos de uso asociados a un usuario anónimo

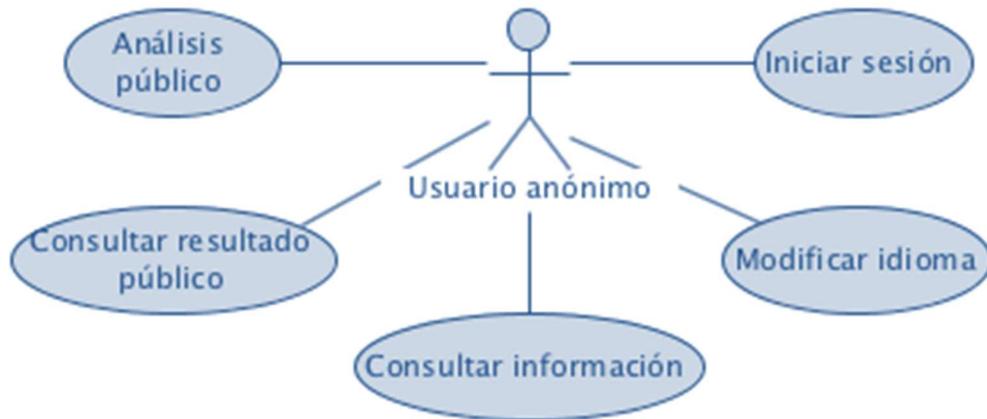


Imagen 29. Casos de uso de un usuario anónimo

Nombre del Caso de Uso
Análisis público
Descripción
El usuario anónimo podrá realizar un análisis público desde la página principal de la Atenea. Esta página deberá ser muy fácil de utilizar. Para realizar esta acción, será necesario seleccionar la URL a analizar, la métrica y el crawler para descargar los enlaces. En esta parte pública, el nivel de profundidad de la ejecución será siempre cero, es decir, solamente se podrá analizar la página que se le indica.

Tabla 12. Caso de uso de análisis público

Nombre del Caso de Uso
Consultar resultado público
Descripción
Una vez que el usuario anónimo inicia un análisis deberá esperar a su finalización, visualizando un mensaje "Cargando". Cuando el análisis finalice, el sistema redirigirá al usuario a la pantalla de resultado público. Esta pantalla mostrará una tabla en la que se especificará la puntuación obtenida para un enlace y una métrica dadas.

Tabla 13. Caso de uso consultar resultado público

Nombre del Caso de Uso
Consultar la información de la zona pública
Descripción
El usuario anónimo podrá visitar las diferentes secciones que componen la zona pública: <ul style="list-style-type: none"> – Accesibilidad – Contacto – Acerca de – Ayuda

Tabla 14. Caso de uso consultar información pública

Nombre del Caso de Uso
Modificar idioma de la aplicación
Descripción
Un usuario anónimo podrá escoger entre dos idiomas para visualizar la aplicación. Estos idiomas son: <ul style="list-style-type: none"> – Español – Inglés

Tabla 15. Caso de uso modificar idioma zona pública

Nombre del Caso de Uso
Inicio de sesión
Descripción
El usuario anónimo podrá iniciar sesión en la aplicación, es decir, acceder a la zona privada de Atenea, si cuenta con las credenciales necesarias. En caso de que las credenciales sean incorrectas, se le volverá a pedir introducir unas credenciales válidas.

Tabla 16. Caso de uso inicio de sesión

5.2.3.2 Casos de uso asociados un usuario registrado

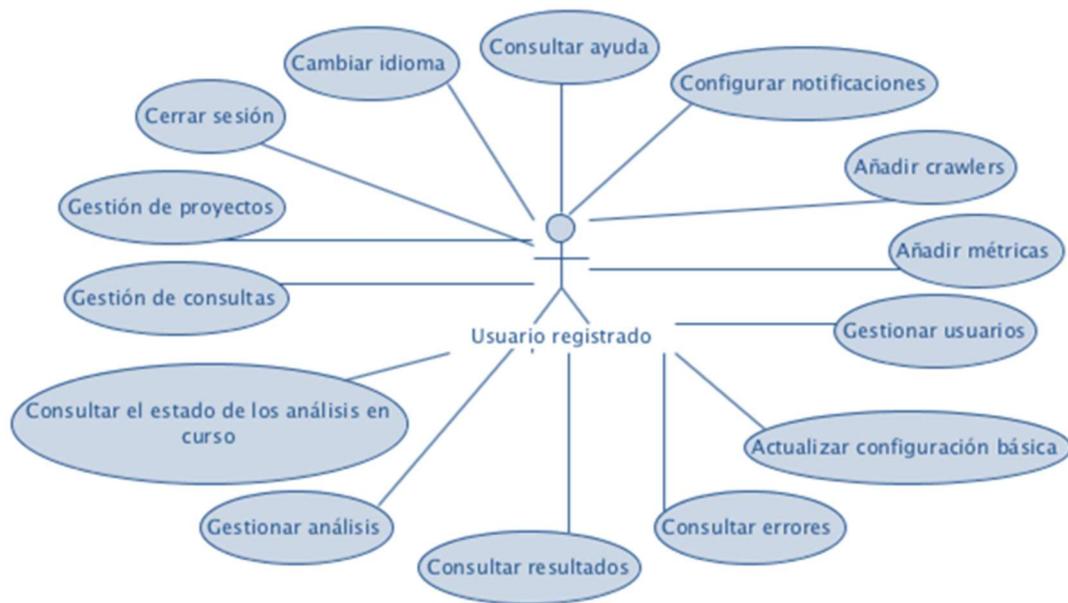


Imagen 30. Casos de uso de un usuario registrado

Nombre del Caso de Uso
Gestión de proyectos
Descripción
<p>Un usuario registrado podrá realizar las siguientes operaciones con los proyectos:</p> <ul style="list-style-type: none"> – Crear un nuevo proyecto. – Abrir un proyecto existente. – Cerrar un proyecto abierto. – Eliminar un proyecto existente. – Importar un proyecto desde un fichero XML. – Exportar un proyecto existente en el sistema a un fichero XML.

Tabla 17. Caso de uso gestión de proyectos

Nombre del Caso de Uso
Gestión de consultas
Descripción
<p>Un usuario registrado podrá realizar las siguientes operaciones con las consultas SQL:</p> <ul style="list-style-type: none"> – Ejecutar una nueva consulta SQL. – Añadir una nueva consulta predeterminada en el sistema. – Abrir una consulta existente en el sistema, para ejecutarla. – Borrar una consulta existente en el sistema.

Tabla 18. Caso de uso gestión de consultas

Nombre del Caso de Uso
Consultar el estado de los análisis en curso
Descripción
<p>Un usuario registrado podrá consultar el estado de todos los análisis en ejecución, ya sean suyos o de otros usuarios.</p> <p>Un usuario registrado podrá realizar una serie de acciones sobre los análisis en curso:</p> <ul style="list-style-type: none"> – Actualizar el estado de los análisis con el fin de refrescar el porcentaje de progreso. – Detener un análisis en curso. – Conocer la traza de un análisis en concreto. Esta traza indicará que enlaces han sido analizados y cuáles no.

Tabla 19. Caso de uso consultar análisis en curso

Nombre del Caso de Uso
Gestionar análisis
Descripción
<p>Un usuario registrado podrá realizar dos tipos de análisis:</p> <ul style="list-style-type: none"> – Análisis de una única URL. – Análisis de un conjunto de URLs, mediante un fichero de texto plano. <p>Además, un usuario registrado puede consultar y exportar el resultado de un análisis concreto realizado desde un fichero con varias URLs. Para esto, se ha creado una pantalla que permite escoger un análisis de entre un listado de ellos asociados a un</p>

Tabla 20. Caso de uso gestión de análisis

Nombre del Caso de Uso
Consultar resultados
Descripción
<p>Un usuario registrado podrá consultar el resultado obtenido para todos los análisis asociados a un proyecto abierto. Además, podrá exportar dichos resultados en varios formatos.</p>

Tabla 21. Caso de uso consultar resultados

Nombre del Caso de Uso
Consultar errores
Descripción
<p>Un usuario registrado podrá consultar los errores obtenido para todos los análisis asociados a un proyecto abierto. Estos errores estarán sujetos a una métrica y una URL determinadas y, por ello, el usuario deberá escoger una métrica para obtener sus errores.</p>

Tabla 22. Caso de uso consultar errores

Nombre del Caso de Uso
Actualizar configuración básica
Descripción
<p>Un usuario registrado podrá modificar la configuración básica que usarán los análisis. Se podrán modificar las siguientes características:</p> <ul style="list-style-type: none"> – Nivel de profundidad. – Número máximo de páginas. – Número de hilos por defecto.

Tabla 23. Caso de uso actualizar configuración básica

Nombre del Caso de Uso
Gestionar usuarios
Descripción
Un usuario registrado podrá añadir nuevos usuarios al sistema, así como eliminar usuarios ya existentes. Un usuario registrado y autenticado nunca podrá eliminar al usuario “admin” ni a sí mismo.

Tabla 24. Caso de uso gestionar usuarios

Nombre del Caso de Uso
Añadir métrica
Descripción
Un usuario registrado puede subir nuevas métricas al sistema con el fin de evaluar nuevos aspectos tanto de accesibilidad como de usabilidad. Para añadir una nueva métrica al sistema, esta, deberá haberse creado según unas pautas específicas y deberá estar comprimida en un archivo JAR.

Tabla 25. Caso de uso añadir métrica

Nombre del Caso de Uso
Añadir crawlers
Descripción
Un usuario registrado puede subir nuevos crawlers al sistema con el fin de modificar el sistema que recoge los enlaces asociados a una web dada. Para añadir un nuevo crawler al sistema, este, deberá haberse creado según unas pautas específicas y deberá estar comprimida en un archivo JAR.

Tabla 26. Caso de uso añadir crawler

Nombre del Caso de Uso
Configurar notificaciones
Descripción
Un usuario registrado puede configurar la frecuencia con la que desea recibir avisos, mientras se están ejecutando sus análisis. Si lo desea, un usuario registrado podrá recibir información sobre el estado de sus propios análisis a través de correos electrónicos.

Tabla 27. Caso de uso configurar notificaciones

Nombre del Caso de Uso
Consultar ayuda
Descripción
Un usuario registrado tendrá, en todo momento, acceso a la ayuda con el fin de conocer el funcionamiento de la zona privada de la aplicación. Esta ayuda estará formada por distintos apartados que se corresponderán con las distintas acciones que se pueden realizar desde la zona privada (Gestión de proyectos, análisis, consultas, usuarios, etc.).

Tabla 28. Caso de uso consultar ayuda

Nombre del Caso de Uso
Cambiar idioma
Descripción
Un usuario registrado también puede escoger entre dos idiomas para visualizar la aplicación. Estos idiomas son: <ul style="list-style-type: none">– Español– Inglés

Tabla 29. Caso de uso modificar idioma II

Nombre del Caso de Uso
Cerrar sesión
Descripción
Un usuario registrado tendrá, en todo momento, acceso al cierre de sesión. Si se utiliza esta función, el sistema volverá a la pantalla de inicio de Atenea y ya no se tendrá acceso a la zona privada.

Tabla 30. Caso de uso cerrar sesión

5.3 Identificación de los Subsistemas en la Fase de Análisis

El objetivo de esta sección es analizar el sistema para poder descomponerlo en sistemas más pequeños o subsistemas que faciliten su posterior análisis.

5.3.1 Descripción de los Subsistemas

Como ya se ha mencionado, una vez terminado el proceso de reingeniería, Atenea deberá mantener su funcionalidad al completo e incluso añadir nuevas funciones. Por todo esto, en general, los principales subsistemas detectados inicialmente se mantendrán entre ambos proyectos y aparecerán nuevos subsistemas. Entre estos se encuentran:

- **Acceso a datos:** Incluye todas las clases que manejan la base de datos. *DAOs*, *Hibernate*, etc.
- **Motor de hilos:** Contiene las clases necesarias para evaluar un conjunto de sitios web en paralelo.
- **Gestión de Métricas:** Contiene las clases necesarias para realizar y guardar la evaluación de una métrica, así como sus checkpoints y los informes de sus errores asociados.
- **Gestión de Crawlers:** Incluye las clases necesarias para la incorporación de nuevos crawlers y sus configuraciones. Además, aparecerán clases que permitan su comunicación con el resto de los subsistemas.
- **Interfaz gráfica:** Este subsistema está compuesto por todas las clases que implementan la interfaz gráfica de la aplicación.
- **Análisis:** Contiene las clases necesarias para la descarga y procesamiento de las diferentes páginas web, obteniendo los enlaces necesarios para realizar un análisis completo. También almacenará todas las clases que permitan realizar la evaluación de uno o varios sitios web.
- **Servicios:** Este subsistema contiene todas las clases relacionadas con la capa de negocio, es decir, son clases que se encargarán de implementar la lógica de la funcionalidad restante asociada a la aplicación.

5.3.2 Descripción de los Interfaces entre Subsistemas

En principio, todos los subsistemas se comunicarán localmente, ya que la aplicación se ejecutará sobre un único servidor. Cada subsistema proveerá una interfaz de operaciones y de acceso a sus datos, por lo que la comunicación entre ellos será a través de estas.

Cabe destacar que, para gestionar la configuración de los crawlers, el sistema se ayudará de una aplicación web complementaria y, por lo tanto, la comunicación entre ambas se realizará a través de un protocolo IP.

5.4 Diagrama de Clases Preliminar del Análisis

En la fase de análisis podemos identificar ya posibles clases del sistema, a partir de los casos de uso y subsistemas ya vistos.

5.4.1 Diagrama de Clases

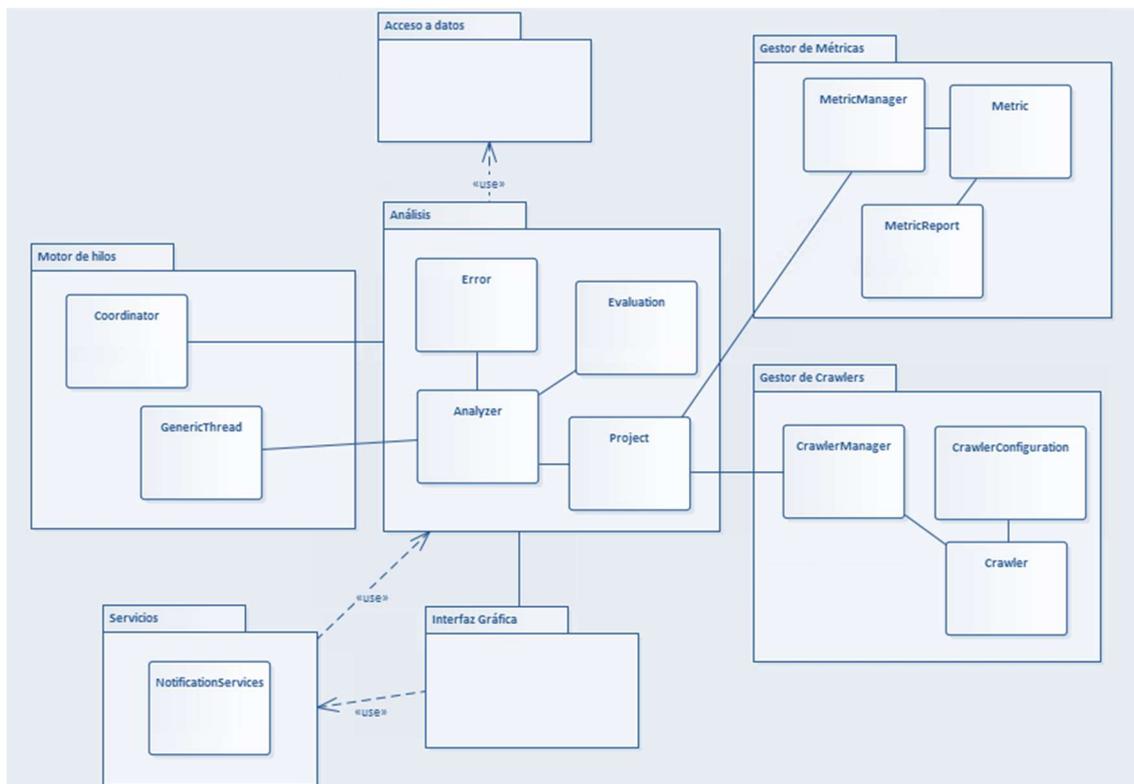


Imagen 31. Diagrama de clases preliminar

5.4.2 Descripción de las Clases

5.4.2.1 Gestor de métricas

Nombre de la Clase
MetricManager
Descripción
Gestiona todas las métricas existentes en el proyecto.
Responsabilidades
Se encarga de cargar las métricas, sus checkpoints y sus informes asociados, de forma correcta.
Atributos Propuestos
metrics: Listado de métricas cargadas. checkpoints: Listado de checkpoints cargados. reports: Listado de informes, si los contienen, de las métricas cargadas.
Métodos Propuestos
getMetrics: Devuelve el listado de métricas existentes en el sistema. getCheckpointsByMetric: Devuelve el listado de checkpoints asociados a una métrica dada. getReportByMetric: Devuelve el informe asociado a una métrica dada.

Tabla 31. Descripción de la clase MetricManager

Nombre de la Clase
Metric
Descripción y Responsabilidad
Representa un objeto métrica.
Atributos Propuestos
name: Nombre de la métrica. checksList: Listado de los checkpoints asociados a esta métrica.
Métodos Propuestos
calculate: Método encargado de calcular el resultado tras evaluar un sitio web con esta métrica.

Tabla 32. Descripción de la clase Metric

Nombre de la Clase
MetricReport
Descripción
Informe de errores asociado a una métrica.
Responsabilidades
Se encarga de generar los informes con los errores producidos durante el análisis para cada métrica. Una métrica puede incluir su propia forma de generar informes y si no es así se el sistema proporciona un informe predeterminado.
Métodos Propuestos
showErrors: Método que se encargada de mostrar los errores asociados a una métrica. La forma de mostrar los errores depende de cada métrica.

Tabla 33. Descripción de la clase MetricReport

5.4.2.2 Gestor de crawlers

Nombre de la Clase
CrawlerManager
Descripción
Gestiona todos los crawlers existentes en el proyecto.
Responsabilidades
Se encarga de cargar los crawlers, así como su configuración.
Atributos Propuestos
crawlers: Listado de crawlers existentes en el sistema.
configCrawler: Listado de las configuraciones asociadas a un crawler determinado.
Métodos Propuestos
getCrawlers: Devuelve el listado de los crawlers existentes en el sistema.
getConfigCrawler: Si existen, este método, se encarga de devolver las características de un crawler.

Tabla 34. Descripción de la clase CrawlerManager

Nombre de la Clase
Crawler
Descripción y Responsabilidad
Representa un objeto crawler genérico.
Atributos Propuestos
name: Nombre del crawler.
configCrawler: Características de configuración asociadas a un crawler. No todos los crawlers tienen configuración.
Métodos Propuestos
getLinks: Método encargado de obtener todos los enlaces existentes en un sitio web dado.

Tabla 35. Descripción de la clase Crawler

Nombre de la Clase
CrawlerConfiguration
Descripción
Representa la configuración asociada a un determinado crawler.
Responsabilidades
Especifica la configuración de un crawler en concreto. Es un elemento opcional. En caso de no existir, el crawler no será configurable, siempre tendrá la misma configuración establecida por defecto.
Atributos Propuestos
crawler: Representa el crawler con esta configuración.
Métodos Propuestos
getConfig: Devuelve el conjunto de características asociadas a un crawler dado.

Tabla 36. Descripción de la clase CrawlerConfiguration

5.4.2.3 Análisis

Nombre de la Clase
Evaluate
Descripción
Clase encargada de almacenar la información referente a un análisis ejecutado.
Atributos Propuestos
score: Puntuación obtenida en el análisis. metric: Hace referencia a la métrica utilizada en el análisis.

Tabla 37. Descripción de la clase Evaluate

Nombre de la Clase
Error
Descripción
Almacena la información sobre los errores obtenidos durante el transcurso de un análisis.
Atributos Propuestos
metric: Métrica utilizada para el análisis. url: URL utilizada en el análisis. checkpoint: Elemento evaluado para este error. score: Puntuación obtenida.

Tabla 38. Descripción de la clase Error

Nombre de la Clase
Analyzer
Descripción
Elemento que evalúa un sitio web y almacena sus resultados.
Responsabilidades
Obtiene toda la configuración de un determinado proyecto y analiza todos los sitios webs que se le pasan. Además, es responsable de almacenar los errores y resultados de la evaluación.
Atributos Propuestos
project: Proyecto abierto para realizar el análisis en curso. Incluiría el crawler, su configuración, las métricas a utilizar, número de páginas, nivel de profundidad, etc. sitesList: Listado de todos los sitios a analizar bajo las condiciones del proyecto.
Métodos Propuestos
execute: Ejecuta el análisis de una determinada URL, bajo los parámetros del proyecto getLinks: Obtiene los enlaces de una determinada URL, atendiendo al crawler seleccionado a la hora de crear el proyecto.

Tabla 39. Descripción de la clase Analyzer

Nombre de la Clase
Project
Descripción y Responsabilidad
Representa un objeto proyecto y almacena toda la información asociada a dicho objeto.
Atributos Propuestos
name: Nombre del proyecto. description: Breve descripción del proyecto. metricList: Listado de las métricas seleccionadas para ejecutar este proyecto. level: Nivel de profundidad del análisis. maxPage: Número máximo de páginas a ejecutar durante un análisis de una URL. crawler: Representa el crawler seleccionado a la hora de crear el proyecto.

Tabla 40. Descripción de la clase Project

5.4.2.4 Motor de hilos

Nombre de la Clase
Coordinator
Descripción
Establecer la coordinación entre todos los hilos creados para los diferentes análisis.
Responsabilidades
Será el encargado de crear todos los hilos que ejecutan una determinada URL. También será responsable de vigilar el proceso de análisis del hilo y su terminación. Cuando un hilo termina, si hay hilos esperando para ejecutar un análisis, deberá ejecutar dichos hilos.
Atributos Propuestos
threadList: Lista de hilos preparados para ser ejecutados.
Métodos Propuestos
getThread: Si es posible, obtiene un hilo listo para ser ejecutado.
run: Método que comienza la ejecución de todos los hilos.

Tabla 41. Descripción de la clase Coordinator

Nombre de la Clase
GenericThread
Descripción
Clase que extiende y modifica los hilos genéricos, para adaptarse a las necesidades del sistema.
Responsabilidades
Es el tipo de hilo utilizado para la ejecución de un análisis.
Atributos Propuestos
analyzer: Elemento que va a analizar este hilo.
Métodos Propuestos
run: Método que se encarga de ejecutar el analizador que contiene.

Tabla 42. Descripción de la clase GenericThread

5.4.2.5 Servicios

Nombre de la Clase
NotificationServices
Descripción
Este servicio de la capa de negocio se encarga de notificar, a un usuario, el estado de sus análisis en curso.
Responsabilidades
Obtiene toda la información asociada a los análisis en curso de un usuario y, gracias a la ayuda de un Timer, la enviará mediante por correos electrónicos.
Atributos Propuestos
project: Proyecto abierto para realizar el análisis en curso. Incluiría el crawler, su configuración, las métricas a utilizar, número de páginas, nivel de profundidad, etc.
sitesList: Listado de todos los sitios a analizar bajo las condiciones del proyecto.
Métodos Propuestos
sendMail: Método que se encarga de enviar, a un usuario, notificaciones por correo electrónico, informando sobre sus análisis en curso. Estos correos se recibirán con una determinada frecuencia, seleccionada con anterioridad.

Tabla 43. Descripción de la clase NotificationServices

5.5 Análisis de Casos de Uso y Escenarios

En esta sección se describirán los casos de uso identificados anteriormente de forma detallada, a través de sus escenarios. Los escenarios describen las interacciones entre los usuarios y el sistema e incluyen información acerca de los objetivos, expectativas, motivaciones, acciones y reacciones que se llevan a cabo.

5.5.1 Caso de uso análisis público

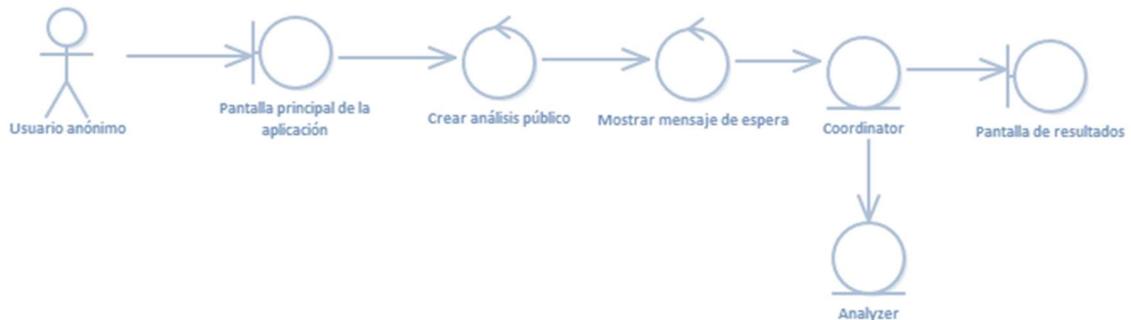


Imagen 32. Diagrama de robustez para análisis público

Realizar un análisis desde la zona pública	
Precondiciones	No
Poscondiciones	El usuario obtiene el resultado del análisis realizado.
Actores	Iniciado y terminado por un usuario anónimo.
Descripción	<p>El usuario:</p> <ol style="list-style-type: none"> 1. Accede a la página principal de la aplicación. 2. Introduce los datos necesarios para hacer un análisis público y son los siguientes: URL, métrica y crawler. 3. Visualiza un mensaje de espera mientras se ejecuta el análisis. 4. Obtiene el resultado del análisis.
Variaciones	<ul style="list-style-type: none"> • Escenario Alternativo 1: Análisis erróneo porque faltan campos obligatorios en el formulario. <ul style="list-style-type: none"> ○ Notificar el hecho al usuario. ○ Volver al paso 1 del escenario principal. • Escenario Alternativo 2: URL inválida. <ul style="list-style-type: none"> ○ Notificar el hecho al usuario, especificando que la URL no sigue un patrón correcto. ○ Volver al paso 1 del escenario principal
Excepciones	<ul style="list-style-type: none"> • La base de datos no está disponible: No se puede ejecutar el análisis, ya que no se puede obtener ni almacenar su información asociada. <ul style="list-style-type: none"> ○ Notificar un error asociado al problema encontrado

Tabla 44. Escenario análisis público

5.5.2 Caso de uso consultar información pública

Consultar información pública	
Precondiciones	No
Poscondiciones	No
Actores	Iniciado y terminado por un usuario anónimo.
Descripción	El usuario: <ol style="list-style-type: none"> 1. Accede a la página principal de la aplicación. 2. Visita una de las siguientes secciones: <ol style="list-style-type: none"> a. Ayuda. b. Contacto. c. Sobre nosotros. d. Accesibilidad.

Tabla 45. Escenario consultar información pública

5.5.3 Caso de uso iniciar sesión en el sistema



Imagen 33. Diagrama de robustez para inicio de sesión

Iniciar sesión	
Precondiciones	El usuario debe estar registrado en el sistema.
Poscondiciones	El usuario está conectado al sistema y se encuentra en la zona privada.
Actores	Iniciado por un usuario anónimo.
Descripción	El usuario: <ol style="list-style-type: none"> 1. Accede a la página de inicio de sesión desde la zona pública. 2. Introduce su usuario y contraseña. 3. Se encuentra en la página de bienvenida de la zona privada.
Variaciones	<ul style="list-style-type: none"> • Escenario Alternativo 1: Análisis erróneo porque faltan campos obligatorios en el formulario. <ul style="list-style-type: none"> ○ Notificar el hecho al usuario. ○ Volver al paso 1 del escenario principal. • Escenario Alternativo 2: Los campos introducidos no corresponden con ningún usuario registrado en el sistema. <ul style="list-style-type: none"> ○ Notificar el hecho al usuario, sin dar detalles por temas de seguridad. ○ Volver al paso 1 del escenario principal
Excepciones	<ul style="list-style-type: none"> • La base de datos no está disponible: No se puede verificar que el usuario está registrado en el sistema. <ul style="list-style-type: none"> ○ Notificar un error asociado al problema encontrado

Tabla 46. Escenario iniciar sesión

5.5.4 Caso de uso modificar idioma

Consultar información pública	
Precondiciones	No
Poscondiciones	No
Actores	Iniciado por un usuario tanto anónimo como registrado.
Descripción	<p>El usuario:</p> <ol style="list-style-type: none"> 1. Modifica el idioma de la aplicación desde cualquier punto de la misma. 2. Puede escoger entre dos idiomas: <ol style="list-style-type: none"> a. Español b. Inglés

Tabla 47. Escenario modificar idioma

5.5.5 Caso de uso actualizar configuración básica

Actualizar configuración básica	
Precondiciones	El usuario debe estar registrado y autenticado en el sistema.
Poscondiciones	La configuración básica del sistema se habrá modificado.
Actores	Iniciado por un usuario registrado.
Descripción	<p>El usuario:</p> <ol style="list-style-type: none"> 1. Accede a la sección de herramientas. 2. Accede a la sección de configuración. 3. Introduce los nuevos datos en los campos que desea modificar. Se puede modificar: <ol style="list-style-type: none"> a. Nivel de profundidad. b. Número máximo de páginas. c. Hilos por defecto. 4. Recibirá una notificación con el éxito de la operación.
Variaciones	<ul style="list-style-type: none"> • Escenario Alternativo 1: Modificación errónea porque algún dato introducido es incorrecto. <ul style="list-style-type: none"> ○ Notificar el hecho al usuario, indicando que los datos deben ser numéricos y, en el caso del nivel de profundidad, estos números deben estar comprendidos entre 0 y 3. ○ Volver al paso 2 del escenario principal.
Excepciones	<ul style="list-style-type: none"> • La base de datos no está disponible: No se puede obtener la configuración actual del sistema. <ul style="list-style-type: none"> ○ Notificar un error asociado al problema encontrado

Tabla 48. Escenario actualizar configuración básica

5.5.6 Caso de uso añadir una métrica

Añadir una métrica	
Precondiciones	El usuario debe estar registrado y autenticado en el sistema.
Poscondiciones	Existirá una nueva métrica en el sistema con la que analizar sitios webs.
Actores	Iniciado por un usuario registrado.
Descripción	El usuario: <ol style="list-style-type: none"> 1. Accede a la sección de herramientas. 2. Accede a la sección de añadir una métrica. 3. A través del sistema que facilita la herramienta, sube el archivo JAR que contiene la implementación de la métrica (y su posible informe de errores asociado). 4. Recibirá una notificación con el éxito de la operación.
Variaciones	<ul style="list-style-type: none"> • Escenario Alternativo 1: El formato del archivo introducido es incorrecto. <ul style="list-style-type: none"> ○ Notificar el hecho al usuario. ○ Volver al paso 2 del escenario principal. • Escenario Alternativo 2: La métrica introducida no está implementada correctamente. <ul style="list-style-type: none"> ○ Notificar el hecho al usuario. ○ Volver al paso 2 del escenario principal.

Tabla 49. Escenario añadir una métrica

5.5.7 Caso de uso añadir un crawler

Añadir un crawler	
Precondiciones	El usuario debe estar registrado y autenticado en el sistema.
Poscondiciones	Existirá un nuevo tipo de crawler en el sistema.
Actores	Iniciado por un usuario registrado.
Descripción	El usuario: <ol style="list-style-type: none"> 1. Accede a la sección de herramientas. 2. Accede a la sección de añadir un crawler. 3. A través del sistema que facilita la herramienta, sube el archivo JAR que contiene la implementación de la métrica (y su posible configuración). 4. Recibirá una notificación con el éxito de la operación.
Variaciones	<ul style="list-style-type: none"> • Escenario Alternativo 1: El formato del archivo introducido es incorrecto. <ul style="list-style-type: none"> ○ Notificar el hecho al usuario. ○ Volver al paso 2 del escenario principal. • Escenario Alternativo 2: El crawler introducido no está implementada correctamente. <ul style="list-style-type: none"> ○ Notificar el hecho al usuario. ○ Volver al paso 2 del escenario principal.

Tabla 50. Escenario añadir un crawler

5.5.8 Caso de uso gestión de usuarios

Dentro del caso de uso de la gestión de usuarios pueden aparecer dos escenarios:

5.5.8.1 Añadir un usuario

Añadir un usuario	
Precondiciones	El usuario debe estar registrado y autenticado en el sistema.
Poscondiciones	Se ha añadido un nuevo usuario al sistema con la frecuencia de avisos a cero.
Actores	Iniciado por un usuario registrado.
Descripción	<p>El usuario:</p> <ol style="list-style-type: none"> 1. Accede a la zona de herramientas. 2. Accede a la zona de añadir usuario. 3. Introduce el nombre de usuario, la contraseña y la dirección de correo electrónico para el nuevo usuario. 4. Obtiene la confirmación de que su proceso ha tenido éxito.
Variaciones	<ul style="list-style-type: none"> • Escenario Alternativo 1: Alta errónea porque faltan campos obligatorios en el formulario. <ul style="list-style-type: none"> ○ Notificar el hecho al usuario. ○ Volver al paso 2 del escenario principal. • Escenario Alternativo 2: Alta errónea porque algún dato es incorrecto y no sigue los patrones establecidos. <ul style="list-style-type: none"> ○ Notificar el hecho al usuario. ○ Volver al paso 2 del escenario principal • Escenario Alternativo 3: Alta errónea porque el nombre de usuario introducido ya existe en el sistema. <ul style="list-style-type: none"> ○ Notificar el hecho al usuario. ○ Volver al paso 2 del escenario principal.
Excepciones	<ul style="list-style-type: none"> • La base de datos no está disponible: No se puede añadir el usuario a la base de datos.

Tabla 51. Escenario añadir usuario

5.5.8.2 Eliminar un usuario

Eliminar usuario	
Precondiciones	El usuario debe estar registrado y autenticado en el sistema. Debe haber algún usuario en el sistema para poder ser eliminado.
Poscondiciones	Se ha eliminado un usuario del sistema.
Actores	Iniciado por un usuario registrado.
Descripción	<p>El usuario:</p> <ol style="list-style-type: none"> 1. Accede a la zona de herramientas. 2. Accede a la zona de eliminar usuario. 3. Visualiza un listado con los demás usuarios del sistema. En este listado no aparecerá el "admin". 4. Selecciona el usuario que desea eliminar. 5. Obtiene la confirmación de que su proceso ha tenido éxito.
Variaciones	<ul style="list-style-type: none"> • Escenario Alternativo 1: Eliminación errónea por no haber escogido un usuario de la lista. <ul style="list-style-type: none"> ○ Notificar el hecho al usuario.

	<ul style="list-style-type: none"> ○ Volver al paso 2 del escenario principal.
Excepciones	<ul style="list-style-type: none"> • La base de datos no está disponible: No se puede eliminar el usuario a la base de datos.

Tabla 52. Escenario eliminar usuario

5.5.9 Caso de uso configurar notificaciones

Configurar notificaciones	
Precondiciones	El usuario debe estar registrado y autenticado en el sistema.
Poscondiciones	Se modificará la frecuencia con la que un usuario desea recibir avisos.
Actores	Iniciado por un usuario registrado.
Descripción	El usuario: <ol style="list-style-type: none"> 1. Accede a la zona de herramientas. 2. Accede a la zona de configurar notificaciones. 3. Visualiza un listado con varios tiempos en minutos. 4. Selecciona un tiempo de la lista. 5. Obtiene la confirmación de que su proceso ha tenido éxito.
Excepciones	<ul style="list-style-type: none"> • La base de datos no está disponible: No se puede modificar la frecuencia con la que un usuario desea recibir avisos.

Tabla 53. Escenario configurar notificaciones

5.5.10 Caso de uso gestión de proyectos

Dentro del caso de uso de la gestión de proyectos pueden aparecer varios escenarios:

5.5.10.1 Crear un proyecto

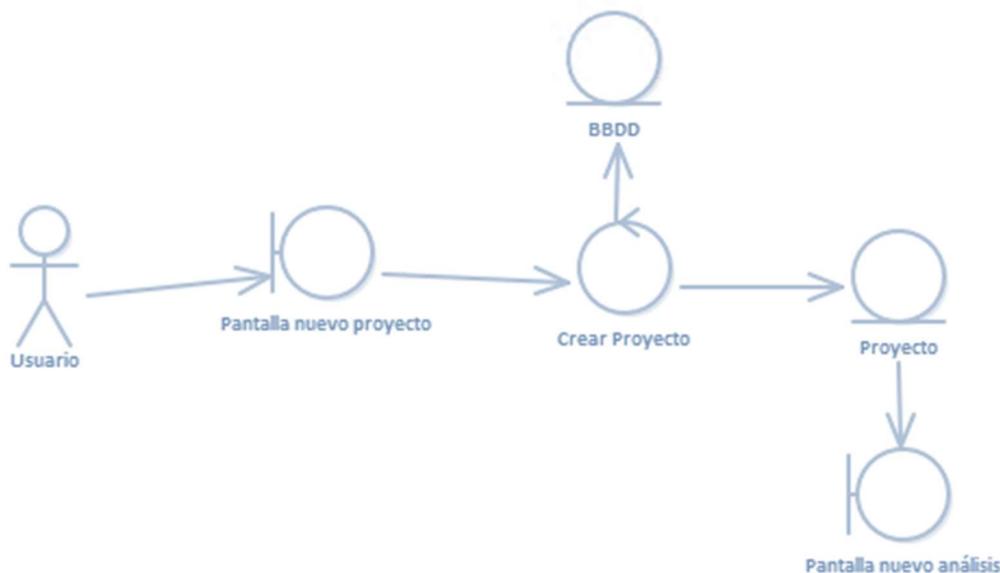


Tabla 54. Diagrama de robustez crear proyecto

Crear un proyecto	
Precondiciones	El usuario debe estar registrado y autenticado en el sistema.
Poscondiciones	Se crea un nuevo proyecto en el sistema.
Actores	Iniciado por un usuario registrado.
Descripción	<p>El usuario:</p> <ol style="list-style-type: none"> 1. Accede a la pantalla de creación de proyectos. 2. Puede visualizar el listado de métricas, crawlers, hilos y número máximo de páginas existentes en el sistema. Selecciona una opción o más (si es posible) de cada listado. 3. Introduce el nombre y descripción del proyecto. 4. Obtiene la información de que su proceso se ha ejecutado con éxito.
Variaciones	<ul style="list-style-type: none"> • Escenario Alternativo 1: Alta errónea porque faltan campos obligatorios en el formulario. <ul style="list-style-type: none"> ○ Notificar el hecho al usuario. ○ Volver al paso 1 del escenario principal.
Excepciones	<ul style="list-style-type: none"> • La base de datos no está disponible: No se puede obtener ni la información existente en el sistema, que el proceso de creación requiere. Además, no se puede guardar el nuevo proyecto en el sistema.

Tabla 55. Escenario crear un proyecto

5.5.10.2 Abrir un proyecto



Imagen 34. Diagrama de robustez abrir un proyecto

Abrir un proyecto	
Precondiciones	El usuario debe estar registrado y autenticado en el sistema.
Poscondiciones	El proyecto estará abierto en el sistema.
Actores	Iniciado por un usuario registrado.
Descripción	<p>El usuario:</p> <ol style="list-style-type: none"> 1. Accede a la pantalla de abrir un proyecto existente. 2. Visualiza un listado con todos los proyectos existentes en el sistema. 3. Selecciona un proyecto de entre el listado para abrirlo. 4. Obtiene un proyecto abierto.
Variaciones	<ul style="list-style-type: none"> • Escenario Alternativo 1: Se produce un error porque no se ha seleccionado ningún proyecto. <ul style="list-style-type: none"> ○ Notificar el hecho al usuario. ○ Volver al paso 1 del escenario principal.
Excepciones	<ul style="list-style-type: none"> • La base de datos no está disponible: No se pueden obtener los proyectos existentes en el sistema.

Tabla 56. Escenario abrir proyecto

5.5.10.3 Eliminar un proyecto

Eliminar un proyecto	
Precondiciones	El usuario debe estar registrado y autenticado en el sistema.
Poscondiciones	El sistema contará con un proyecto menos.
Actores	Iniciado por un usuario registrado.
Descripción	El usuario: <ol style="list-style-type: none"> 1. Accede a la pantalla de eliminar un proyecto existente. 2. Visualiza un listado con todos los proyectos existentes en el sistema. 3. Selecciona un proyecto de entre el listado para eliminarlo. 4. Obtiene la confirmación de que su proceso ha tenido éxito
Variaciones	<ul style="list-style-type: none"> • Escenario Alternativo 1: Se produce un error porque no se ha seleccionado ningún proyecto. <ul style="list-style-type: none"> ○ Notificar el hecho al usuario. ○ Volver al paso 1 del escenario principal.
Excepciones	<ul style="list-style-type: none"> • La base de datos no está disponible: No se pueden eliminar un proyecto de la base de datos.

Tabla 57. Escenario eliminar un proyecto

5.5.10.4 Importar un proyecto

Importar un proyecto	
Precondiciones	El usuario deberá contener un proyecto en formato XML disponible para poder subirlo al sistema.
Poscondiciones	El sistema contará con un nuevo proyecto y estará abierto.
Actores	Iniciado por un usuario registrado.
Descripción	El usuario: <ol style="list-style-type: none"> 1. Selecciona la opción de importar un proyecto. 2. A través del sistema que facilita la herramienta, sube el archivo XML que contiene la información del proyecto a crear. 3. Recibirá una notificación con el éxito de la operación.
Variaciones	<ul style="list-style-type: none"> • Escenario Alternativo 1: El formato del archivo introducido es incorrecto. <ul style="list-style-type: none"> ○ Notificar el hecho al usuario. ○ Volver al paso 1 del escenario principal.
Excepciones	<ul style="list-style-type: none"> • La base de datos no está disponible: No se puede guardar el nuevo proyecto.

Tabla 58. Escenario importar un proyecto

5.5.10.5 Exportar un proyecto

Exportar un proyecto	
Precondiciones	
Poscondiciones	El usuario contará en su ordenador con la información de un proyecto en formato XML.
Actores	Iniciado por un usuario registrado.
Descripción	El usuario: <ol style="list-style-type: none"> 1. Selecciona la opción de exportar un proyecto.

	<ol style="list-style-type: none"> 2. Visualiza un listado con todos los proyectos existentes en el sistema. 3. Selecciona un proyecto de entre el listado para exportarlo. 4. Obtiene un fichero XML en su ordenador con la información del proyecto seleccionado.
Variaciones	<ul style="list-style-type: none"> • Escenario Alternativo 1: Se produce un error porque no se ha seleccionado ningún proyecto. <ul style="list-style-type: none"> ○ Notificar el hecho al usuario. ○ Volver al paso 1 del escenario principal.
Excepciones	<ul style="list-style-type: none"> • La base de datos no está disponible: No se puede recoger la información del proyecto a exportar.

Tabla 59. Escenario exportar un proyecto

5.5.10.6 Cerrar un proyecto

Cerrar un proyecto	
Precondiciones	Un proyecto debe estar abierto en el sistema.
Poscondiciones	Ningún proyecto abierto en el sistema.
Actores	Iniciado por un usuario registrado.
Descripción	El usuario: <ol style="list-style-type: none"> 1. Podrá cerrar un proyecto desde cualquier punto de la aplicación.

Tabla 60. Escenario cerrar proyecto

5.5.11 Caso de uso gestión de consultas

Dentro del caso de uso de la gestión de consultas pueden aparecer varios escenarios:

5.5.11.1 Ejecutar una consulta

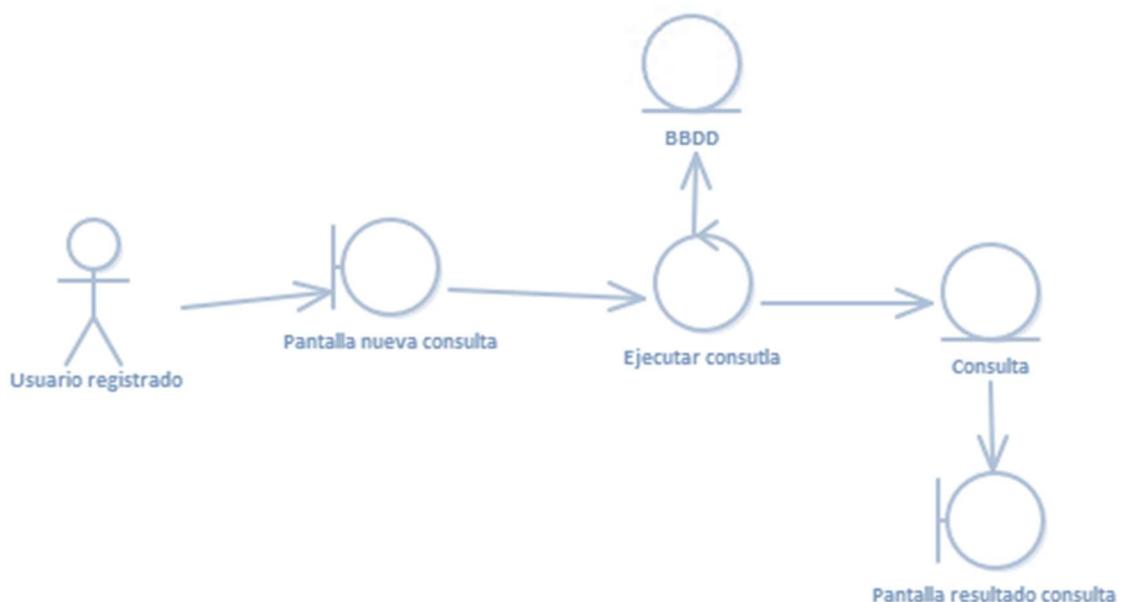


Imagen 35. Diagrama de robustez ejecutar consulta

Ejecutar una consulta	
Precondiciones	El usuario debe estar registrado y autenticado en el sistema.
Poscondiciones	Se obtiene el resultado de la consulta ejecutada.
Actores	Iniciado por un usuario registrado.
Descripción	El usuario: <ol style="list-style-type: none"> 1. Accede a la pantalla de nueva una consulta. 2. Introduce una sentencia SQL en el espacio proporcionado para tal función. 3. Obtiene el resultado de la consulta.
Variaciones	<ul style="list-style-type: none"> • Escenario Alternativo 1: Ejecución errónea porque faltan campos obligatorios en el formulario. <ul style="list-style-type: none"> ○ Notificar el hecho al usuario. ○ Volver al paso 1 del escenario principal. • Escenario Alternativo 2: Ejecución errónea porque el dato introducido no es una sentencia SQL correcta o no corresponde con las tablas de la base de datos. <ul style="list-style-type: none"> ○ Notificar el hecho al usuario. ○ Volver al paso 2 del escenario principal
Excepciones	<ul style="list-style-type: none"> • La base de datos no está disponible: No se puede obtener el resultado producido por la consulta, ya que no se puede ejecutar.

Tabla 61. Escenario ejecutar consulta

5.5.11.2 Añadir una consulta

Añadir una consulta	
Precondiciones	El usuario debe estar registrado y autenticado en el sistema.
Poscondiciones	El sistema cuenta con una nueva consulta predeterminada.
Actores	Iniciado por un usuario registrado.
Descripción	El usuario: <ol style="list-style-type: none"> 1. Accede a la pantalla de añadir una consulta. 2. Introduce el nombre de la consulta y una sentencia SQL asociada. 3. Recibirá una notificación con el éxito de la operación.
Variaciones	<ul style="list-style-type: none"> • Escenario Alternativo 1: Alta errónea porque faltan campos obligatorios en el formulario. <ul style="list-style-type: none"> ○ Notificar el hecho al usuario. ○ Volver al paso 1 del escenario principal. • Escenario Alternativo 2: Alta errónea porque ya existe una consulta con ese nombre, con esa sentencia o ambas <ul style="list-style-type: none"> ○ Notificar el hecho al usuario. ○ Volver al paso 1 del escenario principal • Escenario Alternativo 3: Ejecución errónea porque el dato introducido no es una sentencia SQL correcta o no corresponde con las tablas de la base de datos. <ul style="list-style-type: none"> ○ Notificar el hecho al usuario. ○ Volver al paso 1 del escenario principal
Excepciones	<ul style="list-style-type: none"> • La base de datos no está disponible: No se puede obtener el resultado producido por la consulta, ya que no se puede ejecutar.

Tabla 62. Escenario añadir consulta

5.5.11.3 Abrir una consulta

Abrir una consulta	
Precondiciones	El usuario debe estar registrado y autenticado en el sistema.
Poscondiciones	Se obtiene el resultado de la consulta ejecutada.
Actores	Iniciado por un usuario registrado.
Descripción	<p>El usuario:</p> <ol style="list-style-type: none"> 4. Accede a la pantalla de abrir una consulta existente. 5. Visualiza un listado con todas las consultas existentes en el sistema. 6. Selecciona una consulta de entre el listado para abrirla y ejecutarla. 7. Obtiene el resultado de la consulta.
Variaciones	<ul style="list-style-type: none"> • Escenario Alternativo 1: Se produce un error porque no se ha seleccionado ninguna consulta. <ul style="list-style-type: none"> ○ Notificar el hecho al usuario. ○ Volver al paso 1 del escenario principal.
Excepciones	<ul style="list-style-type: none"> • La base de datos no está disponible: No se pueden obtener las consultas existentes en el sistema.

Tabla 63. Escenario abrir consulta

5.5.11.4 Eliminar una consulta

Eliminar una consulta	
Precondiciones	El usuario debe estar registrado y autenticado en el sistema.
Poscondiciones	El sistema contará con una consulta predeterminada menos.
Actores	Iniciado por un usuario registrado.
Descripción	<p>El usuario:</p> <ol style="list-style-type: none"> 1. Accede a la pantalla de eliminar una consulta existente. 2. Visualiza un listado con todas las consultas existentes en el sistema. 3. Selecciona una consulta de entre el listado para eliminarla. 4. Obtiene la confirmación de que su proceso ha tenido éxito.
Variaciones	<ul style="list-style-type: none"> • Escenario Alternativo 1: Se produce un error porque no se ha seleccionado ninguna consulta. <ul style="list-style-type: none"> ○ Notificar el hecho al usuario. ○ Volver al paso 1 del escenario principal.
Excepciones	<ul style="list-style-type: none"> • La base de datos no está disponible: No se pueden eliminar una consulta de la base de datos.

Tabla 64. Escenario eliminar una consulta

5.5.12 Caso de uso consultar análisis en curso

Dentro de este caso de uso aparecen dos escenarios:

5.5.12.1 Ver estado de los análisis en curso

Ver estado de los análisis en curso	
Precondiciones	El usuario se ha conectado al sistema previamente para entrar en la zona privada.
Poscondiciones	El usuario visualiza los análisis en ejecución.
Actores	Iniciado por el usuario registrado.
Descripción	El usuario: <ol style="list-style-type: none"> 1. Accede a la pantalla de análisis en curso. 2. Obtiene una tabla con todos los análisis que se están ejecutando en ese momento en el sistema. La tabla cuenta con el porcentaje de progreso. 3. Puede actualizar la tabla para ver el progreso del porcentaje. 4. Puede ver la traza de cada análisis.
Notas	Puede ocurrir que no exista ningún análisis en curso en un momento determinado. Se mostrará una pantalla que indica tal hecho.

Tabla 65. Escenario ver estado de los análisis en curso

5.5.12.2 Detener un análisis en curso

Detener análisis en curso	
Precondiciones	El usuario debe estar registrado y autenticado en el sistema.
Poscondiciones	Habrà un análisis en curso menos.
Actores	Iniciado por un usuario registrado.
Descripción	El usuario: <ol style="list-style-type: none"> 1. Accede a la pantalla de análisis en curso. 2. Visualiza un listado con todos los análisis en curso. 3. Selecciona un análisis en curso para detenerlo. 4. Obtiene la confirmación de que su proceso ha tenido éxito, tanto por pantalla como por correo electrónico.
Variaciones	<ul style="list-style-type: none"> • Escenario Alternativo 1: Se produce un error porque no se ha seleccionado ningún análisis. <ul style="list-style-type: none"> ○ Notificar el hecho al usuario. ○ Volver al paso 1 del escenario principal.
Notas	Puede ocurrir que no exista ningún análisis en curso en un momento determinado. Se mostrará una pantalla que indica tal hecho.

Tabla 66. Escenario detener análisis en curso

5.5.13 Caso de uso gestionar análisis

Dentro del caso de uso de gestionar análisis se encuentran tres escenarios:

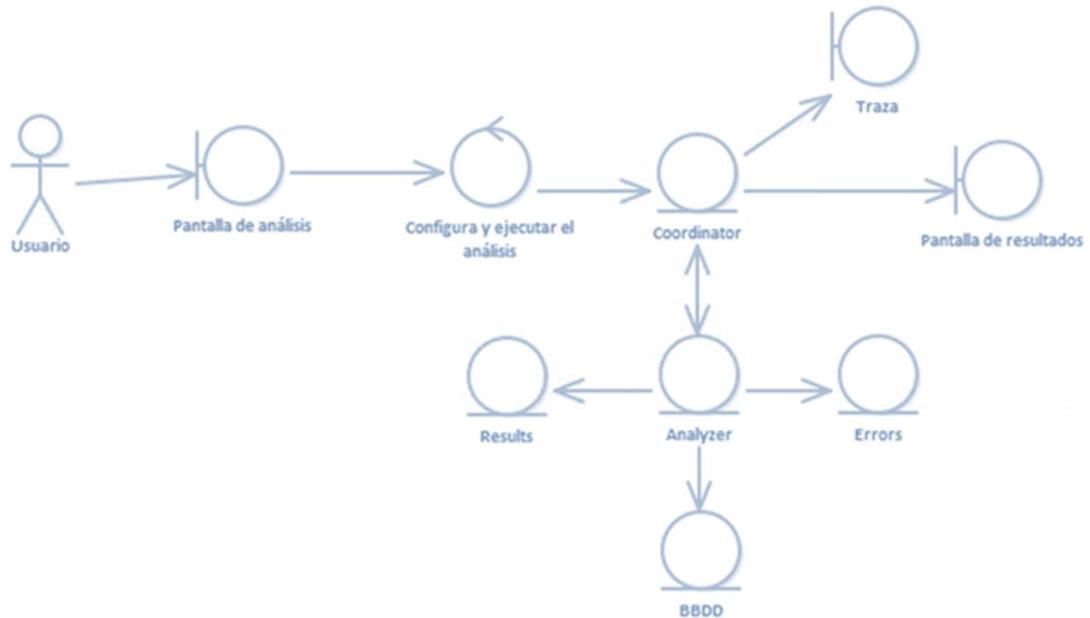


Imagen 36. Diagrama de robustez análisis URL

Analizar una URL	
Precondiciones	El usuario debe estar registrado autenticado en el sistema. El sistema debe contar con un proyecto abierto.
Poscondiciones	Nuevos resultados y errores asociados a este análisis.
Actores	Iniciado por un usuario registrado
Descripción	El usuario: <ol style="list-style-type: none"> 1. Accede a la pantalla de analizar una única URL. 2. Introduce la URL a analizar y el número de hilos que utilizará en el análisis. 3. Visualiza un mensaje de espera mientras se ejecuta el análisis. 4. Una vez finalizado el análisis, el usuario puede visualizar los resultados y errores.
Variaciones	<ul style="list-style-type: none"> • Escenario Alternativo 1: Análisis erróneo porque faltan campos obligatorios en el formulario. <ul style="list-style-type: none"> ○ Notificar el hecho al usuario. ○ Volver al paso 1 del escenario principal. • Escenario Alternativo 2: URL inválida. <ul style="list-style-type: none"> ○ Notificar el hecho al usuario, especificando que la URL no sigue un patrón correcto. ○ Volver al paso 1 del escenario principal
Excepciones	<ul style="list-style-type: none"> • La base de datos no está disponible: No se puede obtener la información necesaria para realizar el análisis (hilos por defecto, proyecto). Además, no se puede almacenar el análisis y sus objetos relacionados.

Tabla 67. Escenario analizar una URL

Analizar una lista de URLs	
Precondiciones	El usuario debe estar registrado autenticado en el sistema. El sistema debe contar con un proyecto abierto.
Poscondiciones	Nuevos resultados y errores asociados a este análisis.
Actores	Iniciado por un usuario registrado
Descripción	El usuario: <ol style="list-style-type: none"> 1. Accede a la pantalla de analizar un conjunto de URLs. 2. Introduce un fichero TXT con un listado de URLs separadas por un retorno de carro. 3. Visualiza un mensaje comunicando que el análisis se encuentra en ejecución.
Variaciones	<ul style="list-style-type: none"> • Escenario Alternativo 1: Análisis erróneo porque faltan campos obligatorios en el formulario. <ul style="list-style-type: none"> ○ Notificar el hecho al usuario. ○ Volver al paso 1 del escenario principal. • Escenario Alternativo 2: El formato del archivo introducido es incorrecto. <ul style="list-style-type: none"> ○ Notificar el hecho al usuario. ○ Volver al paso 1 del escenario principal.
Excepciones	<ul style="list-style-type: none"> • La base de datos no está disponible: No se puede obtener la información necesaria para realizar el análisis (hilos por defecto, proyecto). Además, no se puede almacenar el análisis y sus objetos relacionados.

Tabla 68. Escenario analizar una lista de URLs

Exportar análisis	
Precondiciones	El usuario debe estar registrado autenticado en el sistema. El sistema debe contar con un proyecto abierto.
Poscondiciones	El usuario obtiene todos los análisis asociados al proyecto.
Actores	Iniciado por el usuario registrado.
Descripción	El usuario: <ol style="list-style-type: none"> 1. Accede a la pantalla de exportar análisis 2. Visualiza un listado con los análisis procedentes de ficheros realizados para el proyecto abierto. 3. Selecciona un análisis del listado para obtener sus resultados. 4. Visualiza los resultados del análisis escogido. 5. Puede filtrar los resultados por sitio web y métrica. 6. Puede exportar los resultados en diferentes formatos.
Variaciones	<ul style="list-style-type: none"> • Escenario Alternativo 1: Se produce un error porque no se ha seleccionado ningún análisis. <ul style="list-style-type: none"> ○ Notificar el hecho al usuario. ○ Volver al paso 1 del escenario principal.
Excepciones	<ul style="list-style-type: none"> • La base de datos no está disponible: No se pueden obtener los análisis del presente proyecto ni los resultados del análisis seleccionado.

Tabla 69. Escenario exportar análisis

5.5.14 Caso de uso consultar resultados

Consultar resultados	
Precondiciones	El usuario debe estar registrado y autenticado en el sistema. Un proyecto debe estar abierto.
Poscondiciones	Se obtiene una lista con todos los resultados de los análisis asociados al proyecto abierto.
Actores	Iniciado por un usuario registrado
Descripción	<p>El usuario:</p> <ol style="list-style-type: none"> 1. Accede a la pantalla de resultados. 2. Obtiene una lista de todos los resultados de los análisis asociados al proyecto abierto. 3. Puede filtrar los resultados por métrica y sitio web. 4. Puede exportar los resultados en varios formatos.
Excepciones	<ul style="list-style-type: none"> • La base de datos no está disponible: No se pueden obtener los resultados de los análisis del proyecto abierto.

Tabla 70. Escenario consultar resultados

5.5.15 Caso de uso consultar errores

Consultar errores	
Precondiciones	El usuario debe estar registrado autenticado en el sistema. El sistema debe contar con un proyecto abierto.
Poscondiciones	El usuario obtiene todos los errores asociados a los análisis de una métrica y de un proyecto abierto.
Actores	Iniciado por un usuario registrado
Descripción	<p>El usuario:</p> <ol style="list-style-type: none"> 1. Accede a la pantalla de consultar errores. 2. Visualiza un listado con las métricas asociadas al proyecto. 3. Visualiza un listado con los sitios webs asociados al proyecto. 4. Selecciona una métrica y uno o más sitios webs. 5. Se obtienen los errores y el formato dependerá de la métrica: <ol style="list-style-type: none"> a. Obtiene un archivo PDF con los errores. b. Obtiene un listado con los errores y se puede filtrar por el sitio web.
Variaciones	<ul style="list-style-type: none"> • Escenario Alternativo 1: Se produce un error por no rellenar los campos obligatorios. <ul style="list-style-type: none"> ○ Notificar el hecho al usuario. ○ Volver al paso 1 del escenario principal.
Excepciones	<ul style="list-style-type: none"> • La base de datos no está disponible: No se pueden obtener las métricas y los enlaces asociados a un proyecto. No se puede obtener los errores.

Tabla 71. Escenario consultar errores

5.5.16 Caso de uso consultar ayuda

Ver la ayuda	
Precondiciones	El usuario ha de estar registrado y autenticado.
Poscondiciones	El usuario obtiene la ayuda.
Actores	Iniciado por un usuario registrado
Descripción	<p>El usuario:</p> <ul style="list-style-type: none"> • Accede a la sección de ayuda. • Selecciona la sección sobre la que quiere obtener ayuda. • Obtiene la ayuda requerida.

Tabla 72. Escenario consultar ayuda

5.5.17 Caso de uso cerrar sesión

Salir de la zona privada	
Precondiciones	El usuario se ha conectado al sistema previamente para entrar en la zona privada.
Poscondiciones	El usuario se encuentra en la zona pública del sistema.
Actores	Iniciado por el usuario registrado.
Descripción	<p>El usuario:</p> <ul style="list-style-type: none"> • Selecciona la opción de cerrar sesión. • Aparece en la página principal de la aplicación.
Notas	El usuario debe volver a acceder a la zona privada en cualquier otro momento.

Tabla 73. Escenario cerrar sesión

5.6 Análisis de Interfaces de Usuario

A la hora de diseñar un interfaz de usuario, se deben cumplir una serie de normas atendiendo a la comunicación persona-máquina y procurando que el interfaz sea usable, permita manejar el programa de manera eficiente y que no sea propenso a provocar errores en los usuarios.

En esta sección se detallarán, mediante una descripción y un documento gráfico, los nuevos prototipos de pantalla asociados a las nuevas funciones que se ha añadido en la nueva versión de Atenea. Además, se crearán nuevos prototipos para funcionalidades ya existentes, puesto que algunas se van a modificar tanto que es necesario rehacer su interfaz.

Para el resto de las funciones no se diseñan ni se especifican nuevos interfaces de usuario, debido a la necesidad de mantener la curva de aprendizaje.

Cabe destacar, que para mantener una navegación constante todas las páginas de esta web, así como de sus complementarias, cuentan con una cabecera, un cuerpo y un pie.

5.6.1 Descripción de la Interfaz

5.6.1.1 Configurar notificaciones

En este interfaz, se puede comprobar cómo será la página destinada a configurar la frecuencia con la que el usuario desea recibir notificaciones. Estas notificaciones serán en forma de correo electrónico y permitirán tener una idea del progreso de los análisis.

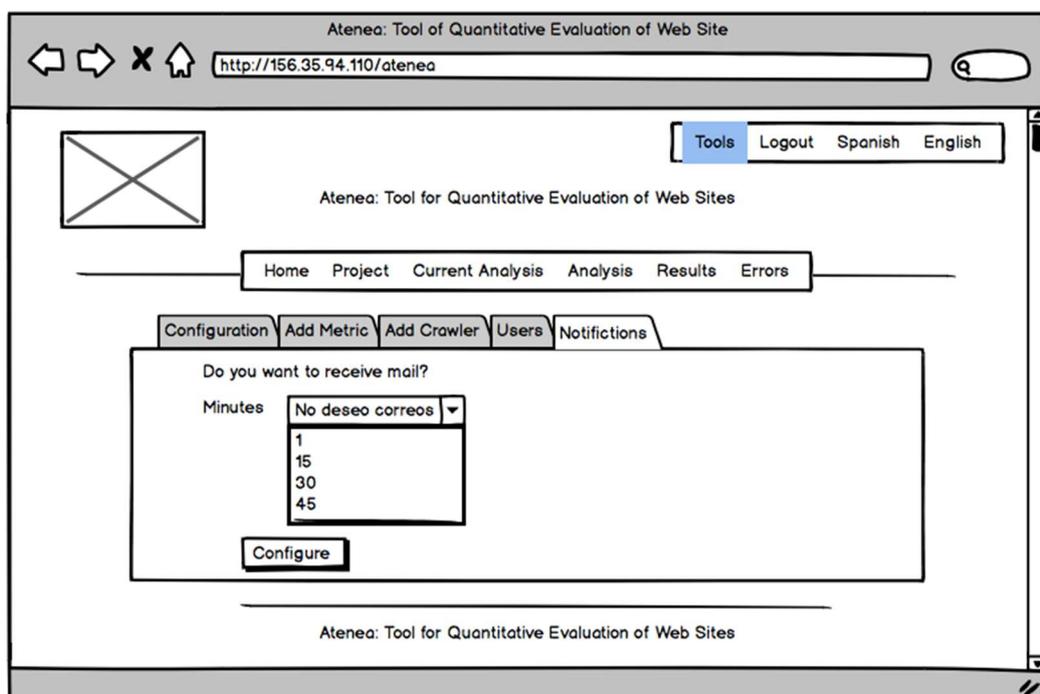


Imagen 37. Prototipo de interfaz para configurar notificaciones

5.6.1.2 Consultar errores

Esta interfaz permite obtener los errores asociados a todos los análisis que se han realizado dentro de un proyecto. Cabe destacar que cada métrica puede tener su propio informe de errores. Por lo tanto, los errores se organizan por métricas.

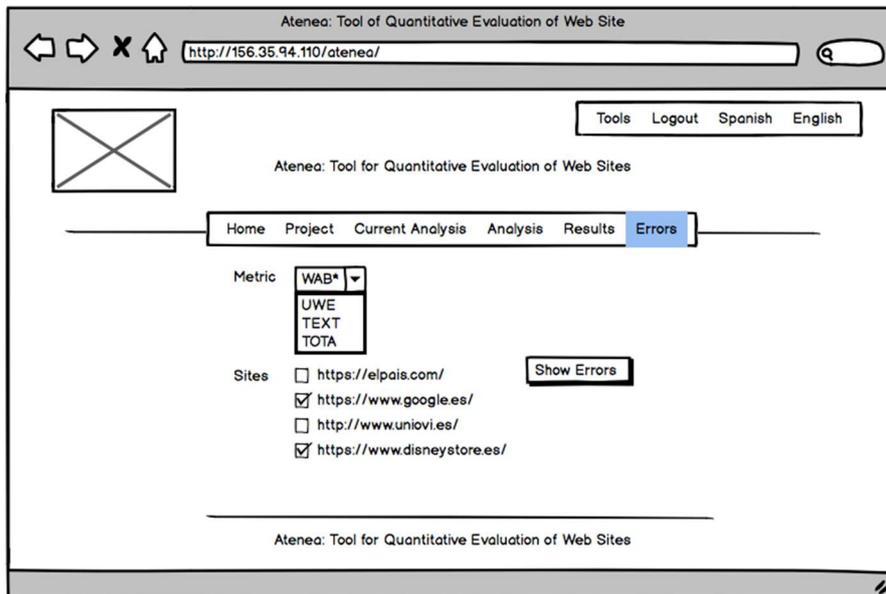


Imagen 38. Prototipo de interfaz para consultar errores

5.6.1.3 Listados

En la versión anterior de Atenea, casi todos los listados se representan mediante selectores sin dar mucha información, lo que dificulta la elección del usuario. Por esto, se decide diseñar un nuevo sistema para mostrar la información.

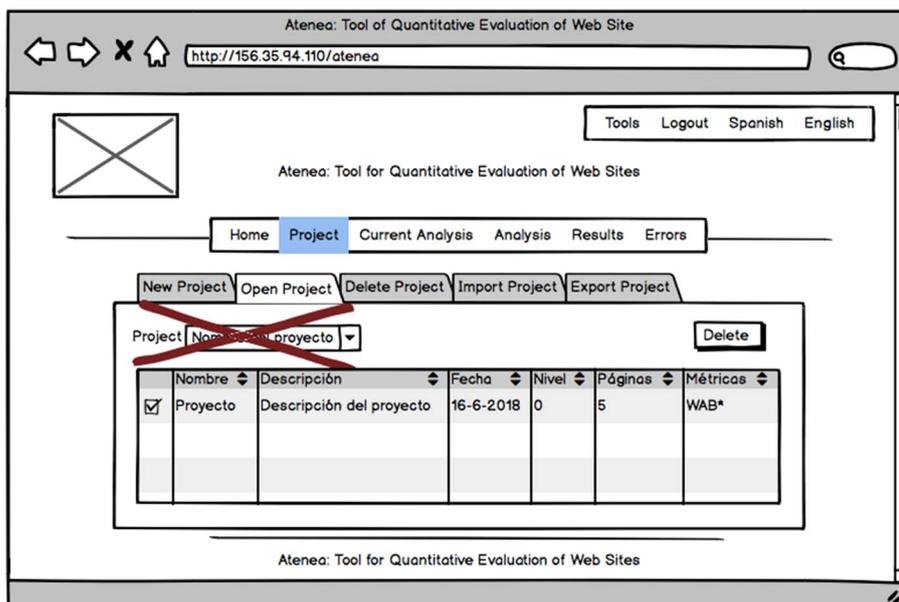


Imagen 39. Prototipo de interfaz para los listados

5.6.2 Descripción del Comportamiento de la Interfaz

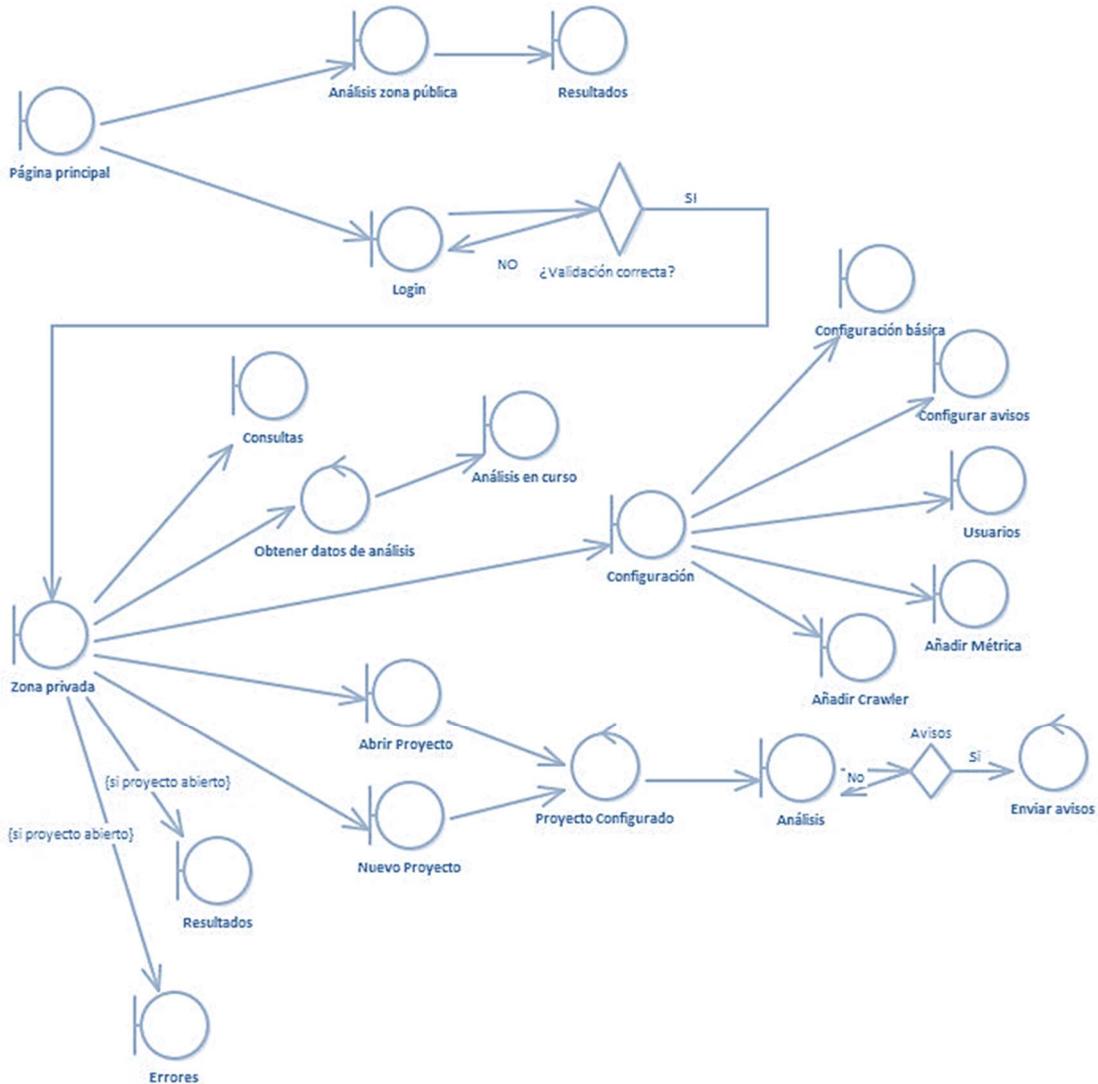
Uno de los requisitos de usabilidad especificados, es la robustez de la interfaz. Por tanto, los mensajes de error se mostrarán en diálogos explicativos y en ningún caso el hecho de producirse un error ha de alterar el funcionamiento normal de la interfaz o provocar que la herramienta finalice su ejecución de forma inesperada.

Se proporcionará al usuario una ayuda general, lo más completa y explicativa posible. También se proporcionará ayuda contextual para las pantallas o situaciones de mayor complejidad.

Esta herramienta web cuenta dos tipos de ayuda. Por un lado, se encuentra la ayuda de la zona pública. Esta ayuda trata los aspectos generales del funcionamiento de la aplicación en su zona pública. Por otro lado, existe una ayuda para la parte privada. Esta ayuda es más exhaustiva e incluye toda la información necesaria para realizar cualquier operación en la parte privada.

5.6.3 Diagrama de Navegabilidad

En esta sección se muestran un diagrama con la finalidad de presentar la navegación que habrá entre las pantallas del programa y su relación con las computaciones que tienen lugar en las mismas. Se simplificará el diagrama para ganar en claridad.



5.7 Especificación del Plan de Pruebas

En esta sección se creará y diseñará el plan de pruebas de la aplicación y sus funciones, así como todos los mecanismos que se utilizarán para detectar errores y corregirlos ya en la fase de implementación.

Escenario 1: Realizar un análisis desde la zona pública	
Prueba	Resultado Esperado
Realizar un análisis. Se introduce la URL, se selecciona la métrica y el crawler.	El sistema lleva al usuario a la pantalla de espera. Una vez finalizado el análisis, el usuario ve el resultado.
Prueba	Resultado Esperado
El usuario abandona la web antes de ver el resultado del análisis	El sistema permanece sin cambios. El análisis continúa, pero el usuario nunca verá el resultado.
Prueba	Resultado Esperado
El usuario deja en blanco el campo de la URL.	El sistema muestra un error.
Prueba	Resultado Esperado
El usuario introduce una URL con formato erróneo.	El Sistema muestra un error.
Prueba	Resultado Esperado
El usuario no selecciona ninguna métrica o crawler.	El Sistema muestra un error.

Tabla 74. Plan de pruebas para análisis público

Escenario 2: Iniciar sesión	
Prueba	Resultado Esperado
El usuario se valida con su usuario y contraseña	El sistema dirige al usuario hacia la zona privada.
Prueba	Resultado Esperado
El usuario no introduce usuario y/o contraseña.	El sistema muestra un error.
Prueba	Resultado Esperado
El usuario introducido no existe en el sistema.	El sistema muestra un error.
Prueba	Resultado Esperado
La contraseña del usuario es incorrecta.	El sistema muestra un error.

Tabla 75. Plan de pruebas para inicio de sesión

Escenario 3: Modificar configuración básica	
Prueba	Resultado Esperado
Se introducen todos los campos a modificar de forma correcta.	El sistema modifica la configuración básica de la herramienta.
Prueba	Resultado Esperado
Se deja algún un campo vacío.	El sistema devuelve un error.
Prueba	Resultado Esperado
Cancelar la operación.	La configuración básica del sistema permanece sin cambios.
Prueba	Resultado Esperado

Se introduce un carácter en un campo numérico.	El sistema no permite la acción.
Prueba	Resultado Esperado
Se introducen valores numéricos negativos.	El sistema muestra un error.
Prueba	Resultado Esperado
Para el nivel de profundidad, el usuario introduce un valor superior a 3.	El sistema muestra un error.

Tabla 76. Plan de pruebas para configuración básica

Escenario 4: Añadir métrica	
Prueba	Resultado Esperado
Se sube un archivo JAR que contiene una métrica correctamente implementada.	El sistema muestra la confirmación de la operación realizada.
Prueba	Resultado Esperado
Se sube un archivo JAR que no contiene una métrica correctamente implementada.	El sistema muestra un error.
Prueba	Resultado Esperado
Se sube un archivo con un formato diferente al especificado.	El sistema no permite la acción.
Prueba	Resultado Esperado
Cancelar la operación	El sistema permanece sin cambios.
Prueba	Resultado Esperado
Se intenta subir una métrica que ya ha sido subida	El sistema sobrescribe la información.
Prueba	Resultado Esperado
No se introduce ningún archivo, es decir, el campo se deja vacío.	El sistema muestra un error.

Tabla 77. Plan de pruebas para añadir una métrica

Escenario 5: Añadir crawler	
Prueba	Resultado Esperado
Se sube un archivo JAR que contiene un crawler correctamente implementado.	El sistema muestra la confirmación de la operación realizada.
Prueba	Resultado Esperado
Se sube un archivo JAR que no contiene un crawler correctamente implementado.	El sistema muestra un error.
Prueba	Resultado Esperado
Se sube un archivo con un formato diferente al especificado.	El sistema no permite la acción.
Prueba	Resultado Esperado
Cancelar la operación	El sistema permanece sin cambios.
Prueba	Resultado Esperado
Se intenta subir una crawler que ya ha sido subido.	El sistema sobrescribe la información.
Prueba	Resultado Esperado
No se introduce ningún archivo, es decir, el campo se deja vacío.	El sistema muestra un error.

Tabla 78. Plan de pruebas para añadir un crawler

Escenario 6: Añadir un usuario	
Prueba	Resultado Esperado
Se introducen todos los campos de forma correcta, para añadir un nuevo usuario.	El sistema muestra la confirmación de la operación.
Prueba	Resultado Esperado
Añadir un nombre de usuario existente en el sistema.	El sistema muestra un error.
Prueba	Resultado esperado
Añadir un correo electrónico existente en el sistema.	El sistema muestra un error.
Prueba	Resultado esperado
Se introduce un correo electrónico que no sigue el siguiente patrón aa@bb.cc	El sistema muestra un error.
Prueba	Resultado Esperado
Dejar algún un campo sin rellenar.	El sistema muestra un error.
Prueba	Resultado Esperado
La contraseña y la repetición de la contraseña no coinciden.	El sistema muestra un error.

Tabla 79. Plan de pruebas para añadir usuarios

Escenario 7: Eliminar un usuario	
Prueba	Resultado Esperado
Eliminar un usuario de entre la lista de usuarios del sistema.	El sistema muestra la confirmación a la operación.
Prueba	Resultado Esperado
Se accede a la sección de eliminar un usuario y no puede aparecer ni el usuario en sesión, ni el administrador.	El sistema no muestra estos dos usuarios.
Prueba	Resultado Esperado
No se escoge ningún usuario a eliminar.	El sistema muestra un error.
Prueba	Resultado Esperado
Cancelar la operación.	El sistema no cambia.

Tabla 80. Plan de pruebas para eliminar usuario

Escenario 8: Configuración de notificaciones	
Prueba	Resultado Esperado
El usuario escoge la frecuencia con la que desea obtener avisos. Puede escoger la opción de no recibir avisos.	El sistema actualiza esta información y muestra un mensaje de éxito.
Prueba	Resultado Esperado
Cancelar la operación.	El sistema permanece sin cambios.

Tabla 81. Plan de pruebas para configurar notificaciones

Escenario 9: Crear un proyecto	
Prueba	Resultado Esperado
Introducir y seleccionar los datos correctamente.	El sistema crea un nuevo proyecto, lo abre y muestra un mensaje de verificación.
Prueba	Resultado Esperado
No seleccionar o introducir algún los campos obligatorios.	El sistema no permite la creación del Proyecto.
Prueba	Resultado Esperado
Cancelar la operación	El sistema permanece sin cambios.

Tabla 82. Plan de pruebas para crear un proyecto

Escenario 10: Abrir un proyecto	
Prueba	Resultado Esperado
Seleccionar un proyecto de entre la lista de proyectos del sistema.	El sistema muestra la confirmación a la operación y se abre un proyecto.
Prueba	Resultado Esperado
No se escoge ningún proyecto para ser abierto.	El sistema muestra un error.
Prueba	Resultado Esperado
Cancelar la operación.	El sistema no cambia.

Tabla 83. Plan de pruebas para abrir un proyecto

Escenario 11: Eliminar un proyecto	
Prueba	Resultado Esperado
Seleccionar un proyecto de entre la lista de proyectos del sistema.	El sistema muestra la confirmación a la operación y se elimina el proyecto.
Prueba	Resultado Esperado
No se escoge ningún proyecto para ser eliminado.	El sistema muestra un error.
Prueba	Resultado Esperado
Cancelar la operación.	El sistema no cambia.

Tabla 84. Plan de pruebas para eliminar un proyecto

Escenario 12: Importar un proyecto	
Prueba	Resultado Esperado
Subir un archivo XML con un proyecto de Atenea.	El sistema crea un nuevo proyecto y lo abre.
Prueba	Resultado Esperado
Subir un fichero con formato diferente a XML.	El sistema no permite esta acción.
Prueba	Resultado Esperado
Subir un fichero en formato XML, pero con un contenido inadecuado.	El sistema no permite la creación del proyecto y muestra un mensaje de error.
Prueba	Resultado Esperado
Cancelar la operación	El sistema permanece sin cambios.

Tabla 85. Plan de pruebas para importar un proyecto

Escenario 13: Exportar un proyecto	
Prueba	Resultado Esperado
Seleccionar un proyecto de entre la lista de proyectos del sistema y pulsar el botón de exportar.	El sistema exporta al ordenador del cliente el fichero.
Prueba	Resultado Esperado
No se escoge ningún proyecto a exportar.	El sistema muestra un error y no se exporta ningún fichero.
Prueba	Resultado Esperado
Cancelar la operación.	El sistema no exporta nada al ordenador del cliente.

Tabla 86. Plan de pruebas para exportar un proyecto

Escenario 14: Cerrar un proyecto	
Prueba	Resultado Esperado
Seleccionar cerrar un proyecto.	El sistema muestra la confirmación de la operación y se cierra el proyecto.

Tabla 87. Plan de pruebas para cerrar un proyecto

Escenario 15: Ejecutar una consulta SQL	
Prueba	Resultado Esperado
Ejecutar una consulta SQL correcta.	El sistema devuelve la información demandada.
Prueba	Resultado Esperado
Introducir una consulta SQL errónea.	El sistema devuelve un error.
Prueba	Resultado Esperado
No introducir ninguna consulta SQL, es decir, no rellenar un campo obligatorio.	El sistema devuelve un error.
Prueba	Resultado Esperado
Cancelar la operación.	El sistema permanece sin cambios.
Prueba	Resultado Esperado
Intentar hacer un INSERT en una tabla de la base de datos.	El sistema devuelve un error.

Tabla 88. Plan de pruebas para ejecutar una consulta

Escenario 16: Añadir una consulta predeterminada	
Prueba	Resultado Esperado
Se introducen un nombre y una sentencia SQL correctos.	El sistema muestra la confirmación de la operación.
Prueba	Resultado Esperado
Añadir un nombre de consulta existente en el sistema.	El sistema muestra un error.
Prueba	Resultado esperado
Añadir una sentencia SQL existente en el sistema.	El sistema muestra un error.
Prueba	Resultado esperado
Se introduce una sentencia SQL con una sintaxis incorrecta.	El sistema muestra un error.
Prueba	Resultado Esperado
Dejar algún campo sin rellenar.	El sistema muestra un error.
Prueba	Resultado Esperado
Cancelar la operación.	El sistema permanece sin cambios.

Tabla 89. Plan de pruebas para añadir una consulta

Escenario 17: Abrir una consulta	
Prueba	Resultado Esperado
Seleccionar una consulta de entre la lista de consultas existentes en el sistema.	El sistema ejecuta la consulta y muestra los resultados obtenidos.
Prueba	Resultado Esperado
No se escoge ninguna consulta de la lista.	El sistema muestra un error.
Prueba	Resultado Esperado
Cancelar la operación.	El sistema no cambia.

Tabla 90. Plan de pruebas para abrir una consulta

Escenario 18: Eliminar una consulta	
Prueba	Resultado Esperado
Seleccionar una consulta de entre la lista de consultas para eliminar.	El sistema muestra un mensaje de verificación y elimina la consulta.
Prueba	Resultado Esperado
No se escoge ninguna consulta de la lista.	El sistema muestra un error.
Prueba	Resultado Esperado
Cancelar la operación.	El sistema no cambia.

Tabla 91. Plan de pruebas para eliminar consulta

Escenario 19: Ver el estado de los análisis en curso	
Prueba	Resultado Esperado
Seleccionar la pestaña de análisis en curso.	El sistema muestra todos los análisis en curso.
Prueba	Resultado Esperado
Seleccionar la pestaña de análisis en curso cuando no existe ningún análisis en ejecución	El sistema muestra un error.
Prueba	Resultado Esperado
Seleccionar la traza de un análisis en curso.	El sistema muestra la traza de dicho análisis.
Prueba	Resultado Esperado
Actualizar los análisis para ver el nuevo porcentaje completado.	El Sistema actualiza la tabla con el nuevo porcentaje.

Tabla 92. Plan de pruebas para análisis en curso

Escenario 20: Detener un análisis en curso	
Prueba	Resultado Esperado
Seleccionar un análisis de la lista de los análisis en curso para detenerlo.	El sistema pide confirmación para detenerlo. Si la respuesta es afirmativa, lo detiene.
Prueba	Resultado Esperado
No se escoge ningún análisis de la lista.	El sistema muestra un error.
Prueba	Resultado Esperado
Cancelar la operación.	El sistema no cambia.

Tabla 93. Plan de pruebas para detener análisis

Escenario 21: Analizar una URL	
Prueba	Resultado Esperado
Introducir una URL correcta para analizar.	El sistema analiza la URL y devuelve el resultado.
Prueba	Resultado Esperado
Introducir una URL con un formato erróneo.	El sistema muestra un error.
Prueba	Resultado Esperado
Dejar el campo de la URL vacío.	El sistema no permite esta acción.
Prueba	Resultado Esperado
Introducir números en la zona reservada para la URL	El sistema muestra un error.
Prueba	Resultado Esperado
Cerrar la aplicación sin haber terminado el análisis.	El análisis sigue ejecutándose en segundo plano. Cuando el análisis termina, el usuario puede ver el resultado.

Prueba	Resultado Esperado
Cerrar la ventana una vez terminado el análisis	El sistema permanece sin cambios. El resultado del análisis ya ha sido almacenado.

Tabla 94. Plan de pruebas para analizar una URL

Escenario 22: Analizar una lista de URLs	
Prueba	Resultado Esperado
Introducir un fichero TXT con un conjunto de URLs separadas por un retorno de carro.	El sistema comienza el análisis.
Prueba	Resultado Esperado
Introducir una lista de URLs con un formato incorrecto.	El sistema muestra un error.
Prueba	Resultado Esperado
Introducir un fichero vacío.	El sistema muestra un error.
Prueba	Resultado Esperado
Introducir un fichero con un formato diferente a TXT.	El sistema no permite esta acción.
Prueba	Resultado Esperado
Si el usuario desea notificaciones y el análisis comienza.	El sistema enviará varios correos electrónicos con la información asociada a dicho análisis.
Prueba	Resultado Esperado
Cerrar la aplicación sin haber terminado el análisis.	El sistema continúa haciendo el análisis sin necesidad de atención por parte del usuario.
Prueba	Resultado Esperado
Cerrar la aplicación una vez terminado el análisis.	El sistema permanece sin cambios. Los resultados del análisis ya han sido almacenados.

Tabla 95. Plan de pruebas para analizar un conjunto de URLs

Escenario 23: Exportar un análisis asociado un conjunto de URLs	
Prueba	Resultado Esperado
Seleccionar un análisis de entre la lista de los análisis ejecutados del sistema.	El sistema mostrará los resultados asociados a este análisis y permitirá su exportación.
Prueba	Resultado Esperado
No se escoge ningún análisis para ver y exportar sus resultados.	El sistema muestra un error.
Prueba	Resultado Esperado
Cancelar la operación.	El sistema no exporta nada al ordenador del cliente.

Tabla 96. Plan de pruebas para exportar análisis

Escenario 24: Consultar y exportar resultados	
Prueba	Resultado Esperado
Seleccionar la pestaña de resultados para ver los resultados asociados a un proyecto.	El sistema muestra los resultados.
Prueba	Resultado Esperado
Se intenta acceder a los resultados sin tener un proyecto abierto.	El sistema no permite esta acción, ya que no tiene la pestaña activa.
Prueba	Resultado Esperado
El sistema permite exportar los resultados	El sistema exporta el fichero con los resultados en

y se selecciona el formato.	el formato seleccionado al ordenador del cliente.
-----------------------------	---

Tabla 97. Plan de pruebas para consultar resultados

Escenario 25: Consultar y exportar errores	
Prueba	Resultado Esperado
El usuario selecciona una métrica y uno o más sitios webs de forma correcta.	El sistema dependiendo de la métrica muestra o exporta los errores correctamente.
Prueba	Resultado Esperado
No se selecciona la métrica o no se seleccionan los sitios o ambos a la vez.	El sistema no permite esta acción.
Prueba	Resultado Esperado
No existen errores para una métrica o un determinado sitio.	El sistema muestra o exporta un listado vacío.
Prueba	Resultado Esperado
Cancelar operación	El sistema permanece sin cambios.

Tabla 98. Plan de pruebas consultar errores

Escenario 26: Cerrar sesión	
Prueba	Resultado Esperado
Seleccionar cerrar sesión.	El sistema cierra la sesión y dirige al usuario a la página inicial de la aplicación.

Tabla 99. Plan de pruebas cerrar sesión

Capítulo 6. Diseño del Sistema

6.1 Arquitectura del Sistema

6.1.1 Diagramas de Paquetes

Los diagramas de paquetes se utilizan, especialmente, cuando el diagrama de clases de todo el sistema es demasiado grande para visualizarse correctamente. A lo largo de este apartado se va a mostrar el nuevo diagrama de paquetes sobre el que está implementado Atenea.

EL objetivo es mostrar una abstracción de la funcionalidad de las clases y sus relaciones.

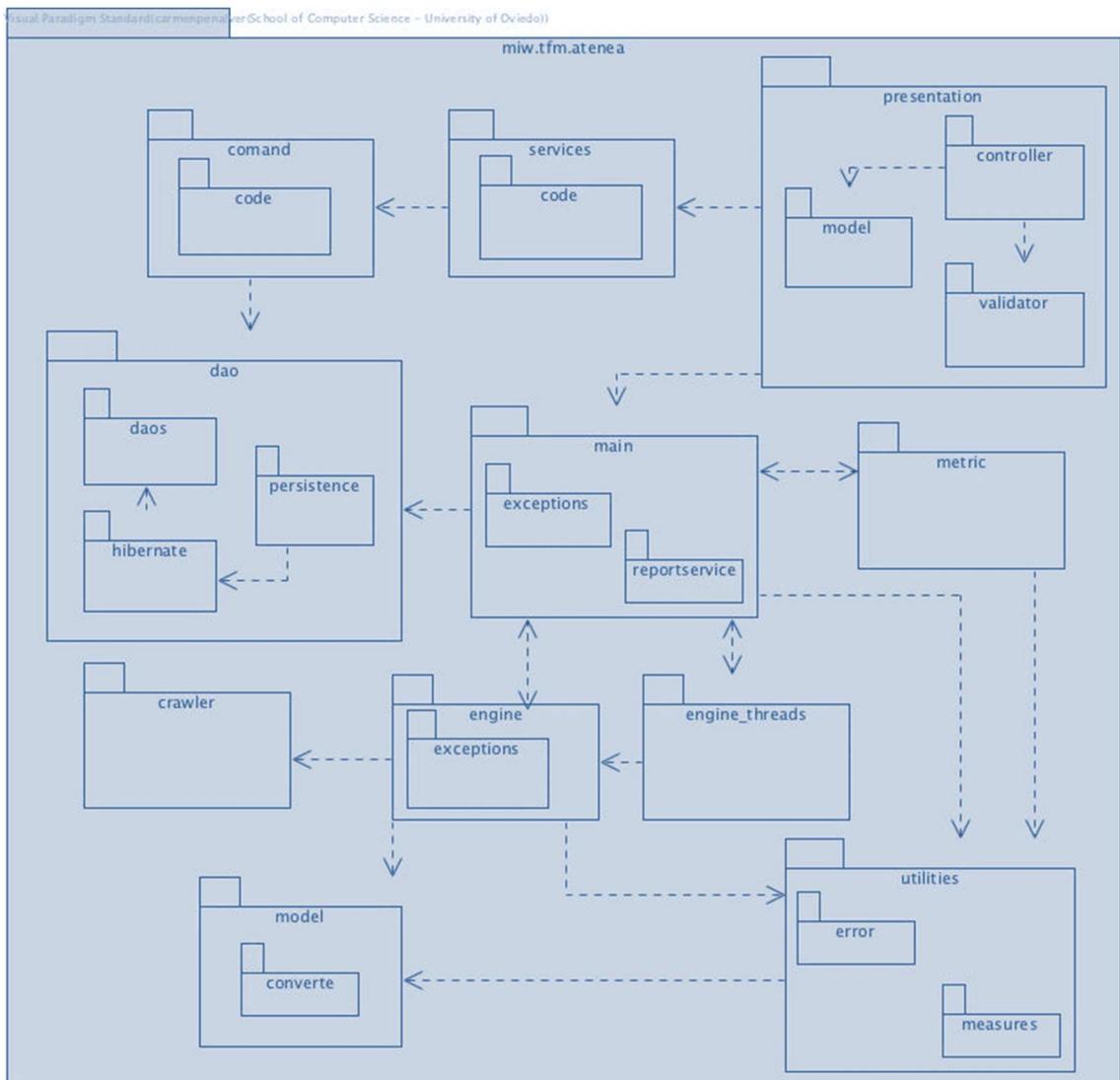


Imagen 40. Diagrama de paquetes

6.1.1.1 *Paquete model*

Este paquete comprende las clases que representan el modelo de dominio de la aplicación, es decir, contiene clases que modelan objetos reales y son manejados en todo momento tanto por el sistema como por el usuario.

Cuenta con un paquete converter, donde se encuentran unas clases de conversión, que permiten manejar los atributos de tipo fecha y poder así persistirlos en la base de datos.

Contiene algunas clases que no pertenecen al modelo de dominio, pero al tratarse de clases sin comportamiento, se decide encapsularlas en este mismo paquete.

6.1.1.2 *Paquete utilities*

Este paquete contiene una serie de clases de utilidad necesarias para la aplicación. Contiene clases para manejar los resultados, generar un fichero de Log, etc.

Cabe destacar la aparición de las clases *ObserverTrace* y *ObservableTrace* utilizadas para el manejo del patrón *Observer* durante la ejecución de la traza.

6.1.1.3 *Paquete main*

Representa la funcionalidad del sistema y la gestión de proyectos. Además, proporciona un interfaz de comunicación entre el paquete de presentación y los demás componentes de la herramienta.

6.1.1.4 *Paquete engine*

En este paquete se encuentran las clases que, en conjunto, se consideran el núcleo del sistema, ya que son las encargadas de realizar la evaluación de los diferentes sitios web.

6.1.1.5 *Paquete engine_threads*

Las clases de este paquete implementan el motor que posibilita la ejecución de varios hilos en paralelo. Incorpora las clases que permiten que el hilo principal de ejecución no esté ocupado mientras se realiza un análisis (la aplicación no quede bloqueada durante las evaluaciones). También incluye las clases que hacen posible la *traza* durante una ejecución.

6.1.1.6 *Paquete crawler*

En este paquete se almacenan las clases encargadas de la descarga y obtención de enlaces para su posterior evaluación.

6.1.1.7 *Paquete metric*

Este paquete contiene todas las clases relacionadas con la gestión de métricas, es decir, contiene las interfaces para la creación de las métricas, así como sus informes y checkpoints asociados.

Además, este paquete contiene todas las clases relacionadas con la gestión de crawlers, es decir, contiene las interfaces para la creación de los crawlers existentes en el sistema, así como su configuración.

La clase *MetricCrawlerManager* se encarga de la carga ordenada de todos los elementos relacionados con las métricas y los crawlers.

6.1.1.8 *Paquete services*

Este paquete contiene las clases que, junto a las clases de todos los paquetes anteriormente comentados, constituyen la capa de negocio, es decir, la lógica de la aplicación.

Las clases de este paquete se consideran intermediarios entre la capa de presentación y la capa de persistencia, pasando por el paquete *comand*.

6.1.1.9 *Paquete presentation*

Este paquete encapsula las clases asociadas a la capa de presentación, es decir, contiene todos los controladores que comunican al usuario con las posibles acciones que el sistema puede realizar.

Además, cuenta con un paquete que contiene las clases necesarias para validar todas las entradas que proceden del exterior. También se crea un modelo asociado a la capa de presentación, que servirá para recoger los datos introducidos por el usuario a través de las vistas, así como para presentar los datos en estas.

Para facilitar la creación de este paquete y sus clases se ha utilizado el framework web Spring.

6.1.1.10 *Paquete dao*

Este paquete contiene todas las clases necesarias para interactuar con la base de datos asociada al sistema, es decir, se encargan de la persistencia de la aplicación.

Internamente este paquete está estructurado siguiendo el patrón DAO y dividido en varios subpaquetes, con el fin de mantener una estructura clara y organizada.

6.1.1.11 *Paquete comand*

En este paquete se encapsulan una serie de clases que hacen de intermediario entre la capa de persistencia y la capa de negocio, es decir, se utilizan para realizar las pertinentes invocaciones desde las clases del paquete *services* a las clases del paquete *dao*.

6.1.2 Diagramas de Componentes

Mediante el siguiente diagrama se mostrarán los diferentes componentes del sistema y las dependencias entre ellos.

Lo que se representa son los componentes más importantes que ayuden a comprender el funcionamiento global del sistema.

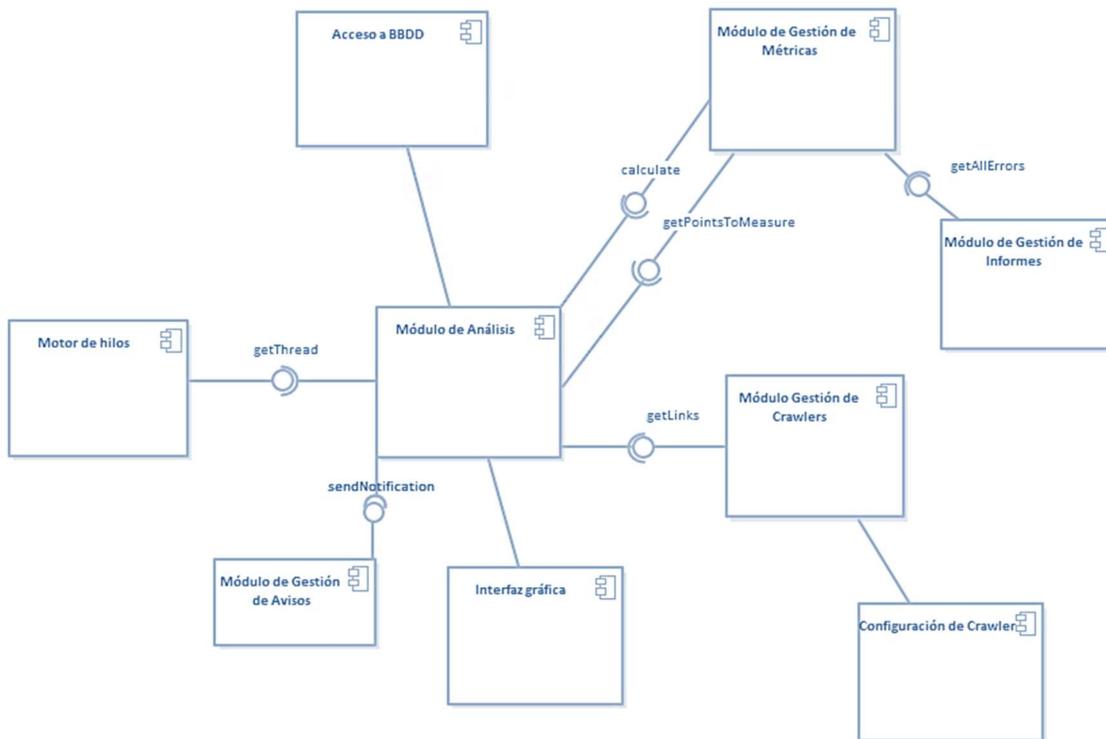


Imagen 41. Diagrama de componentes

6.1.3 Diagramas de Despliegue

El diagrama de despliegue ofrece una vista de los módulos en ejecución y sobre los equipos previstos.

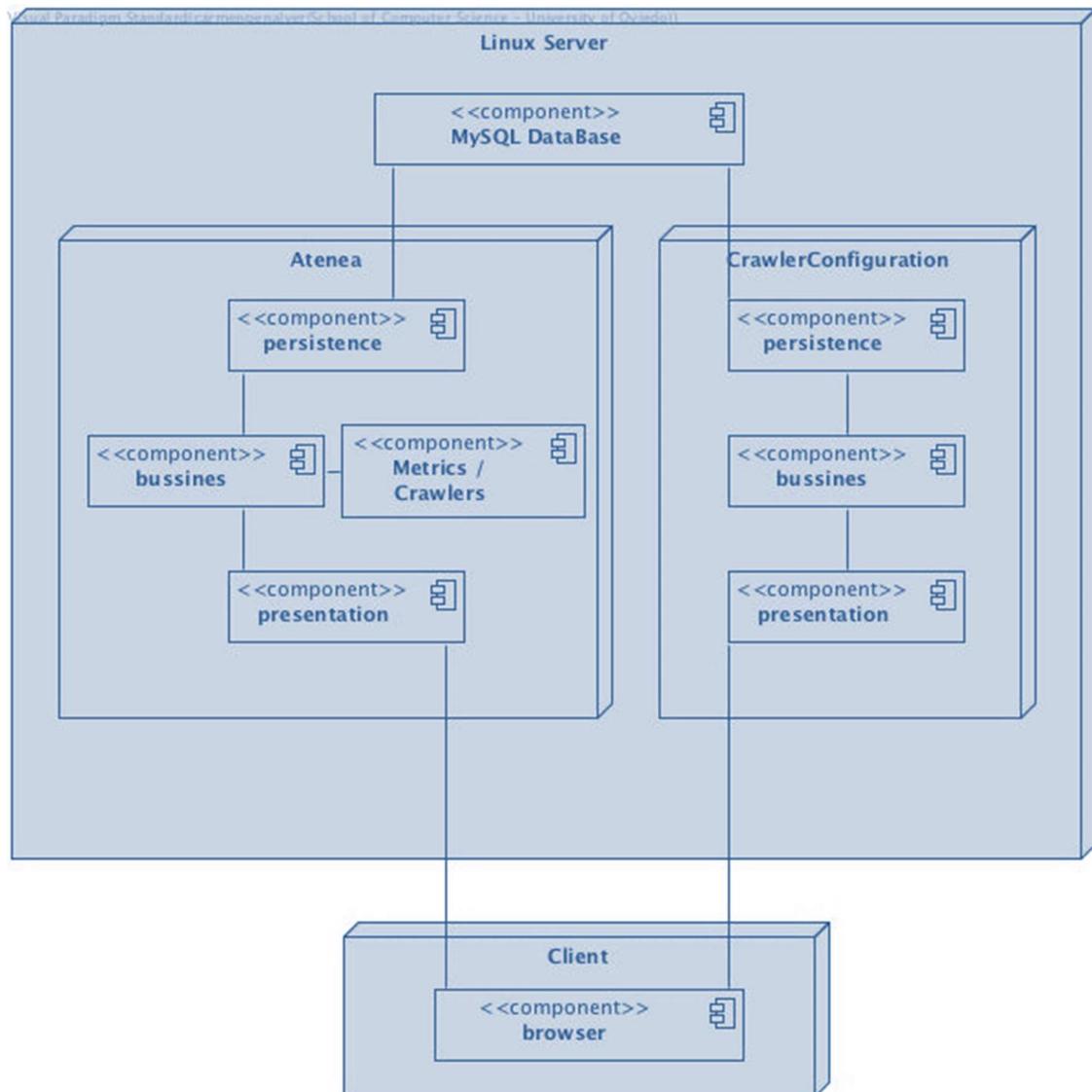


Imagen 42. Diagrama de despliegue

6.1.3.1 Elemento Client

Soporta el navegador donde se presenta la página web de Atenea y permite que el usuario realice diferentes acciones a través de peticiones HTTP(S).

6.1.3.2 Elemento Server

Este elemento representa el servidor donde se alojan la aplicación tanto principal como su complementaria. Además, sobre el servidor está corriendo la base de datos MySQL que permite almacenar y recoger toda la información necesaria para el correcto funcionamiento del sistema completo.

6.1.3.3 *Elemento Atenea*

Este elemento comprende todos los componentes que en conjunto conforman el funcionamiento de la aplicación. Se comunica con la base de datos a través del módulo de persistencia.

Se comunica con su aplicación secundaria a través de la base de datos y, de esta manera, se permite dar configuración a los crawlers existentes en el sistema.

6.1.3.4 *Elemento CrawlerConfiguration*

Este elemento comprende todos los componentes que, en conjunto, permiten configurar las características de los crawlers que acusen su necesidad.

Se comunica con la aplicación Atenea a través de la base de datos.

6.2 Diseño de Clases

En el siguiente apartado, se describen los atributos y métodos de cada una de las clases que componen la herramienta. Cada diagrama se corresponde con un paquete de los ya descritos anteriormente.

6.2.1 Diagrama de clases del paquete model

Estas clases han sido creadas siguiendo los principios de mapeo de *Hibernate*, lo que nos permite trabajar de una forma mucho más sencilla con dichos objetos.

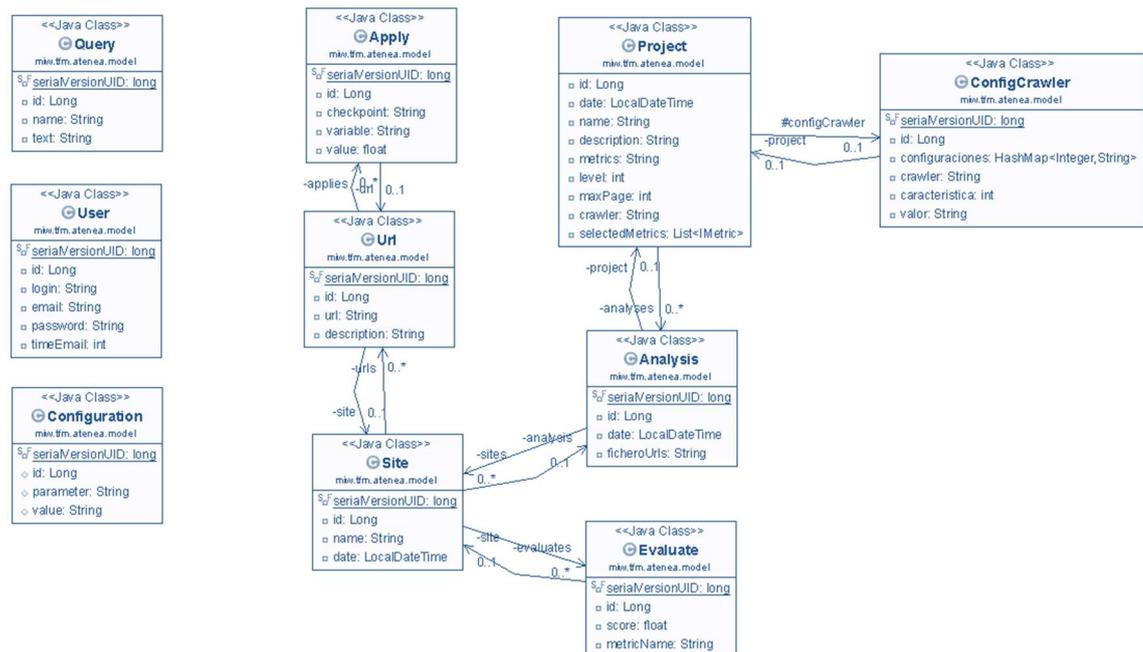


Imagen 43. Diagrama de clases del paquete model

6.2.2 Diagrama de clases del paquete utilities

Este paquete contiene una serie de clases de utilidad necesarias para la aplicación. Contiene clases para manejar los resultados, generar un fichero de Log, etc. En el siguiente diagrama se muestran las más significativas.

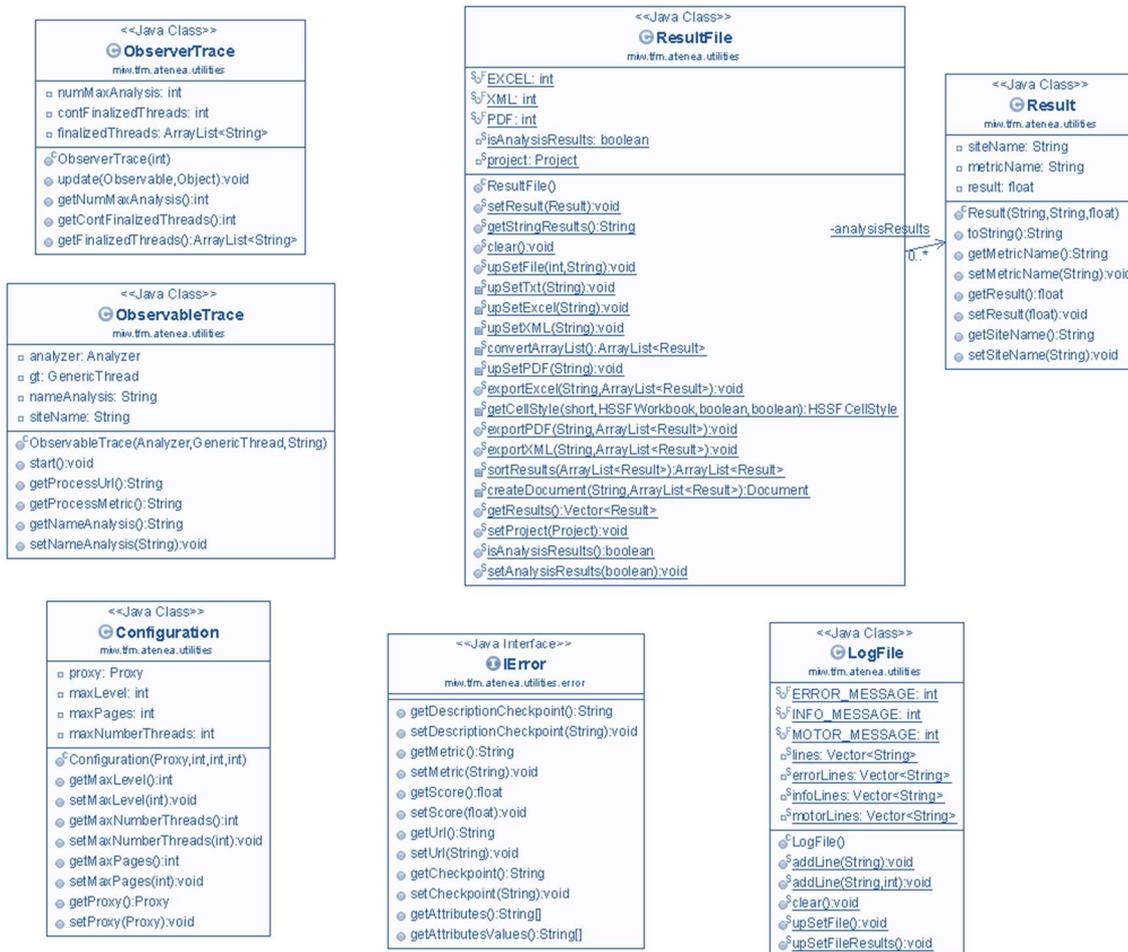


Imagen 44. Diagrama de clases del paquete utilities

6.2.3 Diagrama de clases del paquete main

A continuación, se muestra el diagrama de clases del paquete *main*, en el que se realizan las operaciones principales de la aplicación.



Imagen 45. Diagrama de clases del paquete main

6.2.4 Diagrama de clases del paquete engine

Estas clases conforman el núcleo de la aplicación, ya que se encargan de la funcionalidad de evaluar los diferentes sitios webs que se proporcionan.

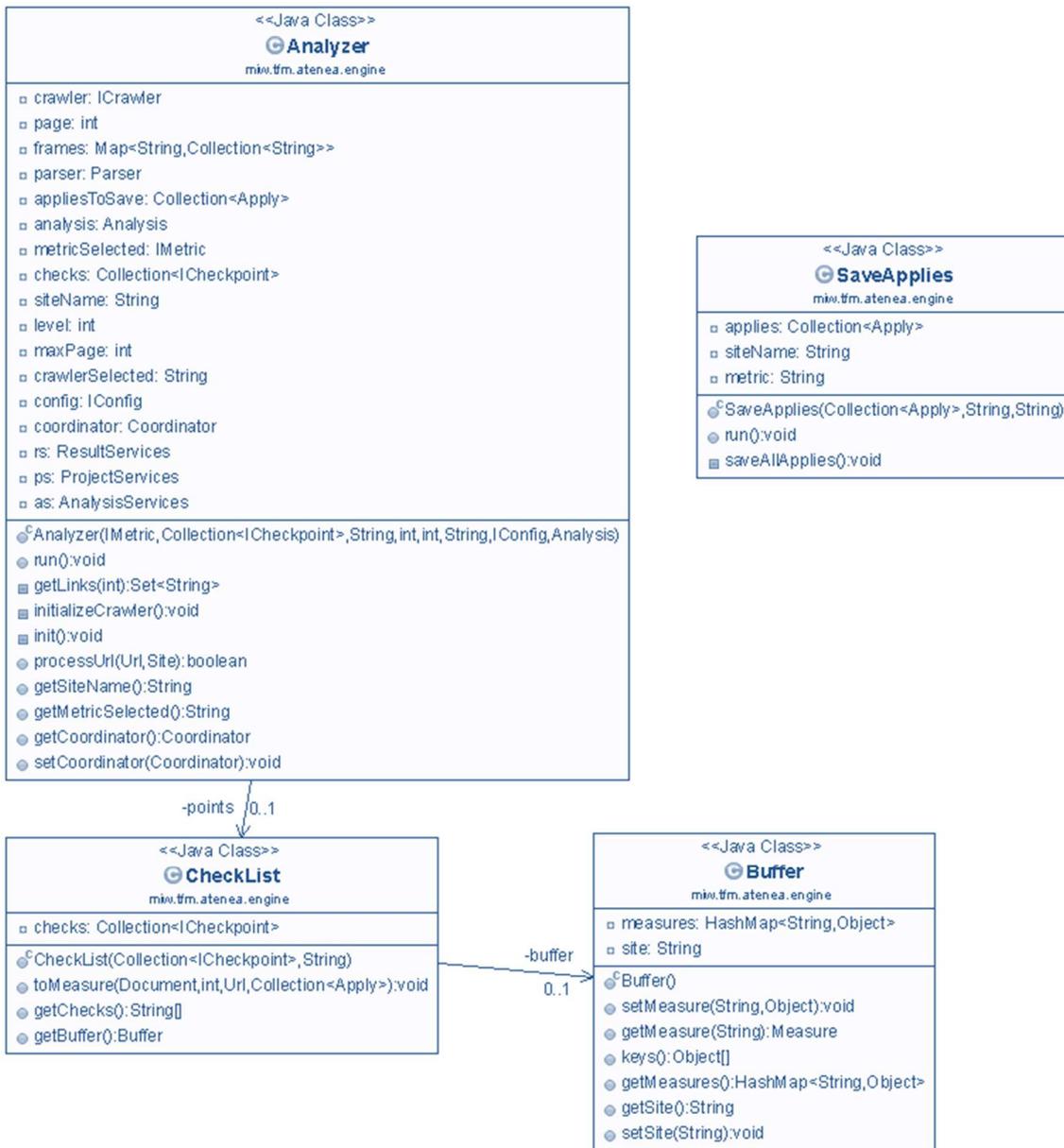


Imagen 46. Diagrama de clases del paquete engine

6.2.5 Diagrama de clases del paquete engine_threads

Las clases de este paquete implementan el motor que posibilita la ejecución de varios hilos en paralelo. Incorpora la clase *MainThread* que permite que el hilo principal de ejecución no esté ocupado mientras se realiza un análisis (la aplicación no quede bloqueada durante las evaluaciones).

También incluye las clases que hacen posible la *traza* durante una ejecución.

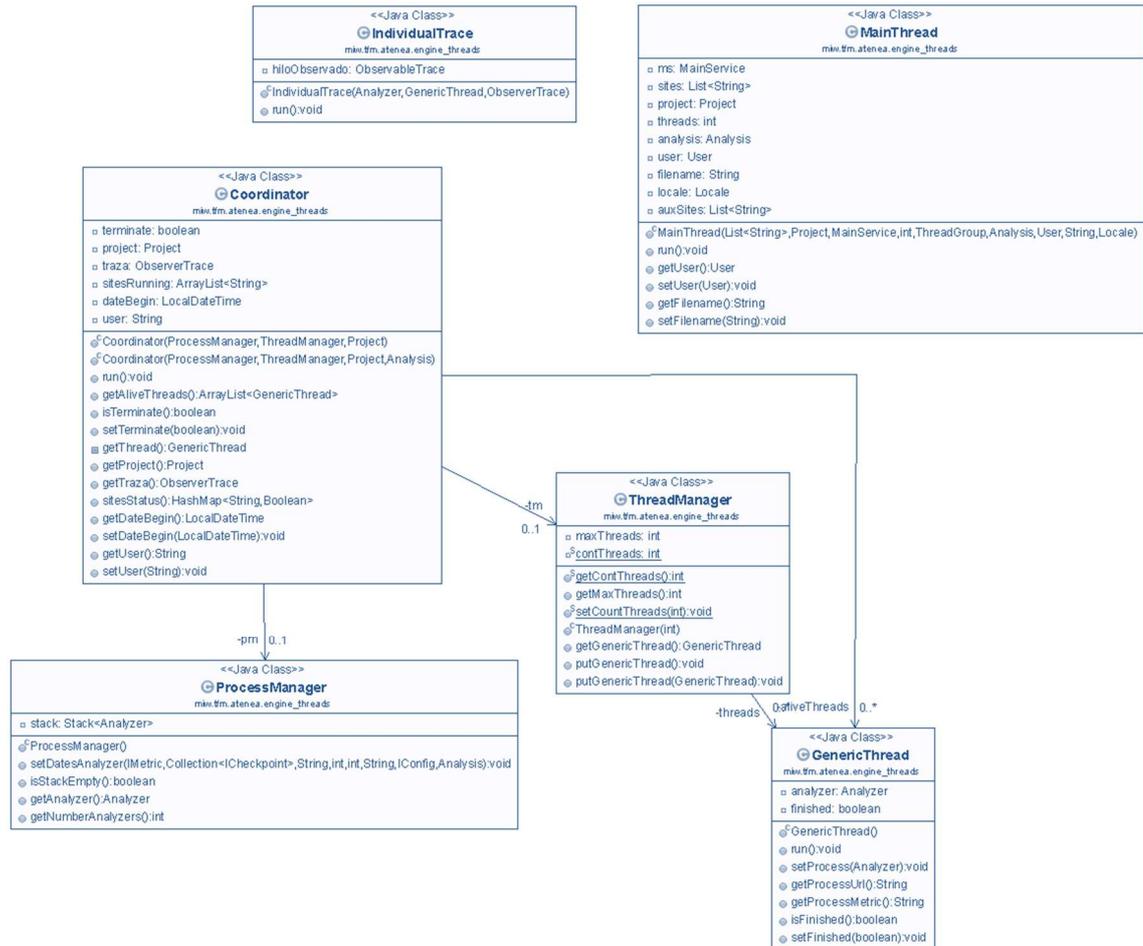


Imagen 47. Diagrama de clases del paquete engine_threads

6.2.6 Diagrama de clases de los paquetes metric y crawler

El paquete metric contiene todas las clases relacionadas con la gestión de métricas, es decir, contiene las interfaces para la creación de las métricas, así como sus informes y checkpoints asociados.

En el paquete crawler se integran varias clases para poder permitir la configuración de un crawler. Además, se incorpora un nuevo adaptador que permite comunicar al crawler con el resto de la aplicación.

La clase *MetricCrawlerManager* se encarga de la carga ordenada de todos los elementos relacionados con las métricas y los crawlers.

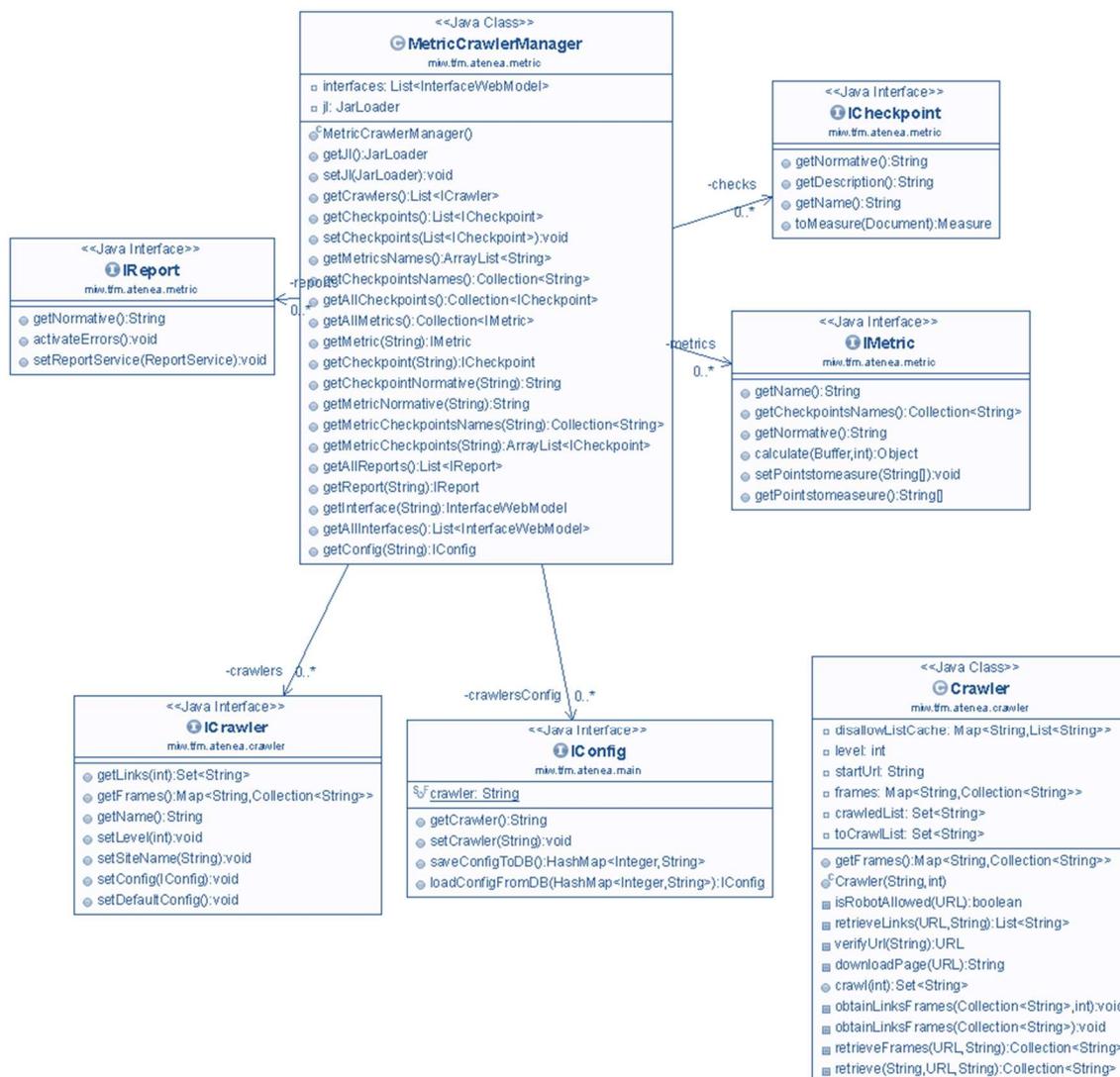


Imagen 48. Diagrama de clases de los paquetes metric y crawler

6.2.7 Diagrama de clases del paquete services

Este paquete contiene las clases que, junto a las clases de todos los paquetes anteriormente comentados, constituyen la capa de negocio, es decir, la lógica de la aplicación. Las clases de este paquete se consideran intermediarios entre la capa de presentación y la capa de persistencia, pasando por el paquete *command*.

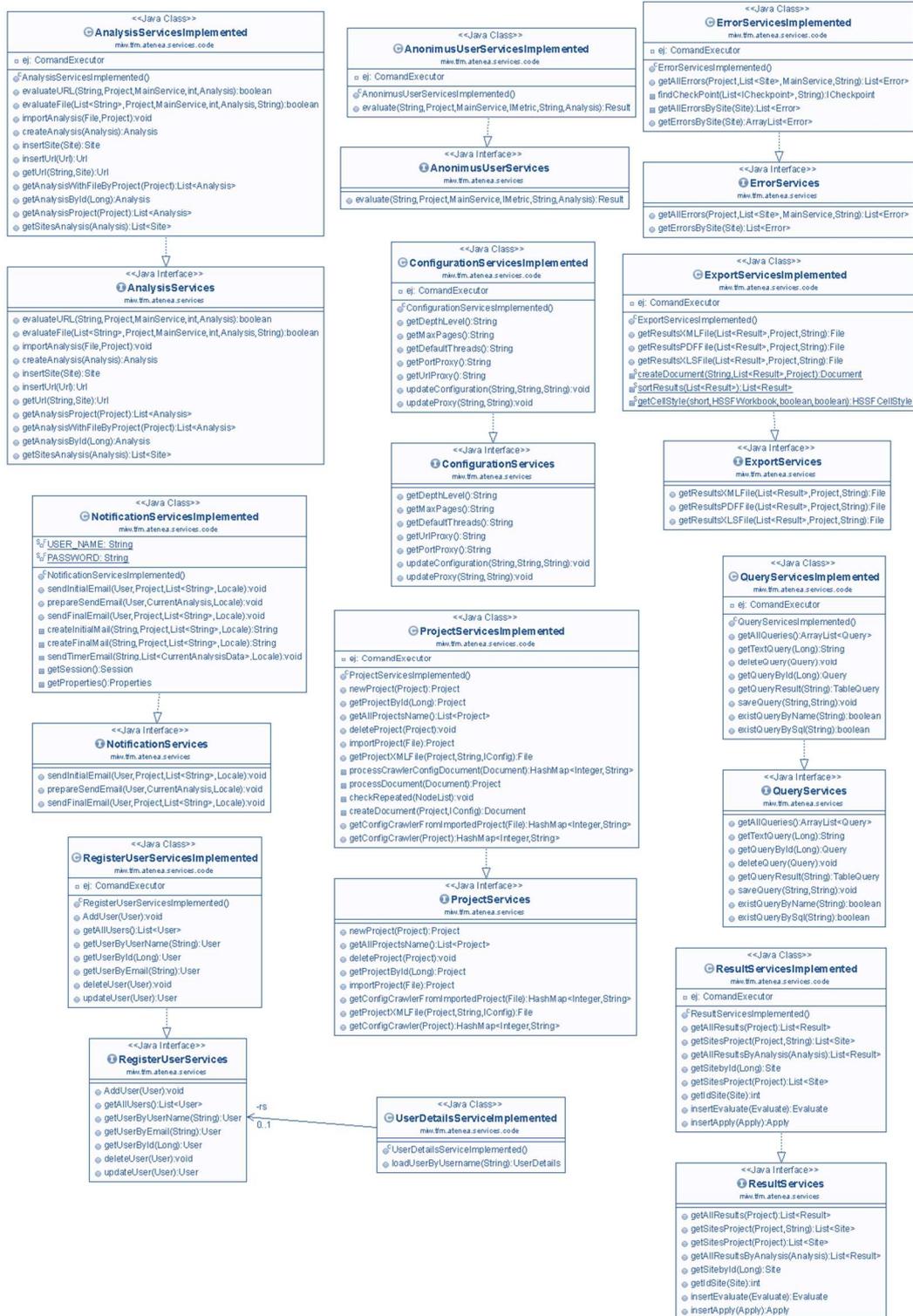


Imagen 49. Diagrama de clases del paquete services

6.2.8 Diagrama de clases del paquete dao

Como puede verse en el siguiente gráfico, para la comunicación de esta aplicación web con su base de datos se ha utilizado el patrón DAO. Se ha incluido además la herramienta *Hibernate*, la cual realiza el mapeo de atributos entre la base de datos y el modelo de objetos de la aplicación.

No tiene sentido hablar de este paquete sin su complementario, el paquete *comand*.

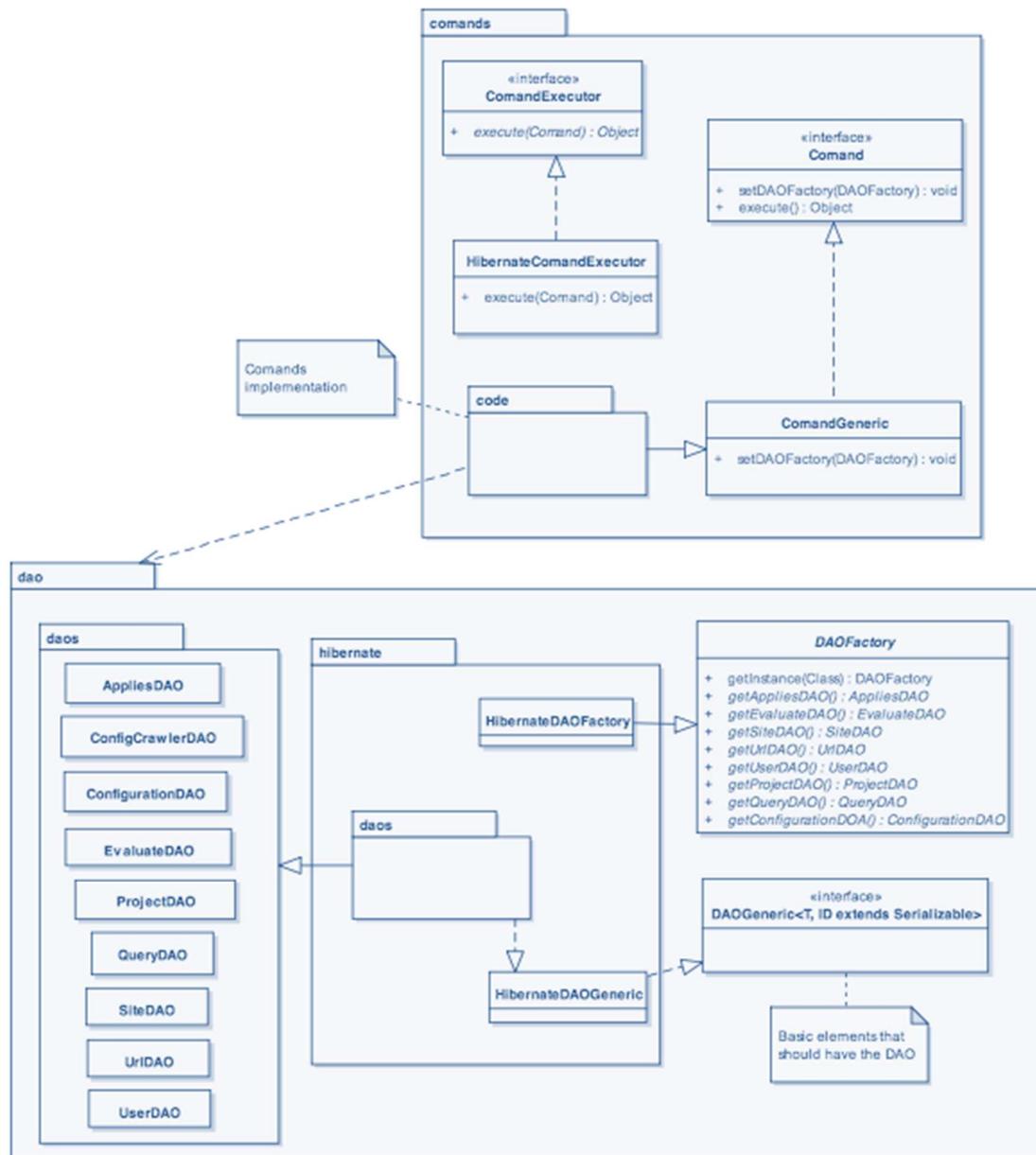


Imagen 50. Diagrama de clases de los paquetes dao y comand

Dentro del paquete daos se encuentran las clases donde se realizan las consultas a la base de datos. Estas consultas están agrupadas según el objeto sobre el que trabajan.

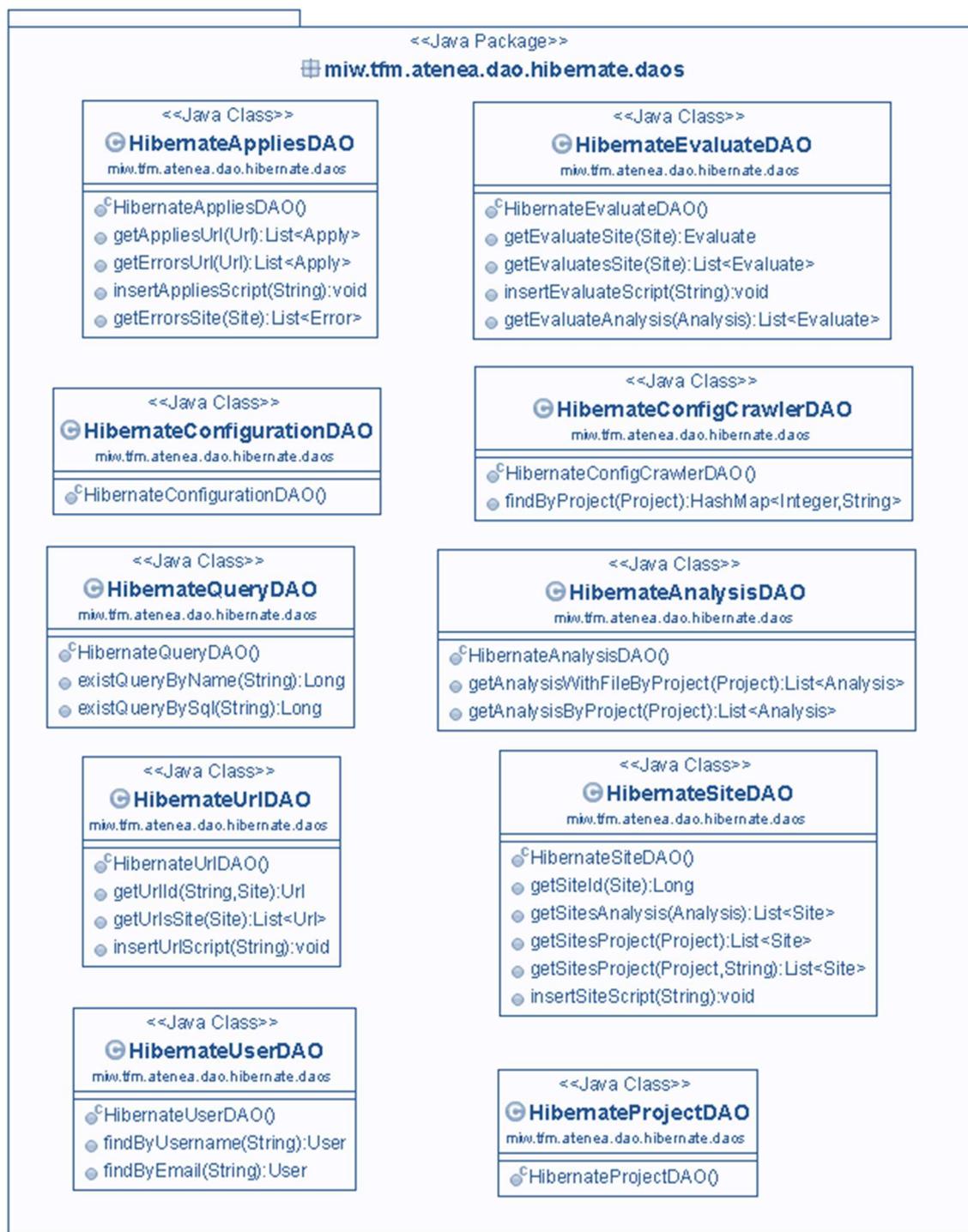


Imagen 51. Diagrama de clases del paquete daos

Además, estas clases encargadas de realizar las consultas a la base de datos heredan de una clase que implementa una serie de métodos genéricos como son: añadir, eliminar, modificar y recuperar información.

6.2.9 Diagrama de clases del paquete presentation

Para implementar la capa de presentación se ha utilizado el framework Spring, que trabaja sobre un patrón Modelo-Vista-Controlador. Para presentar las vistas al usuario, se ha decidido utilizar JSP.

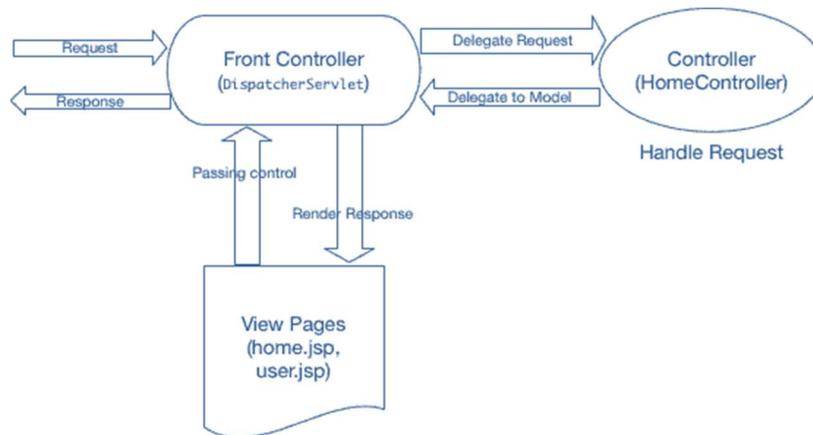


Imagen 52. Arquitectura de Spring MVC

Como se comentó en el capítulo Reingeniería del sistema, este paquete está subdividido en paquetes, atendiendo a su funcionalidad: model, validator y controller.

El paquete *controller* a su vez se encuentra dividido en diferentes paquetes que hacen referencia a las acciones que el usuario puede realizar a través de la aplicación.

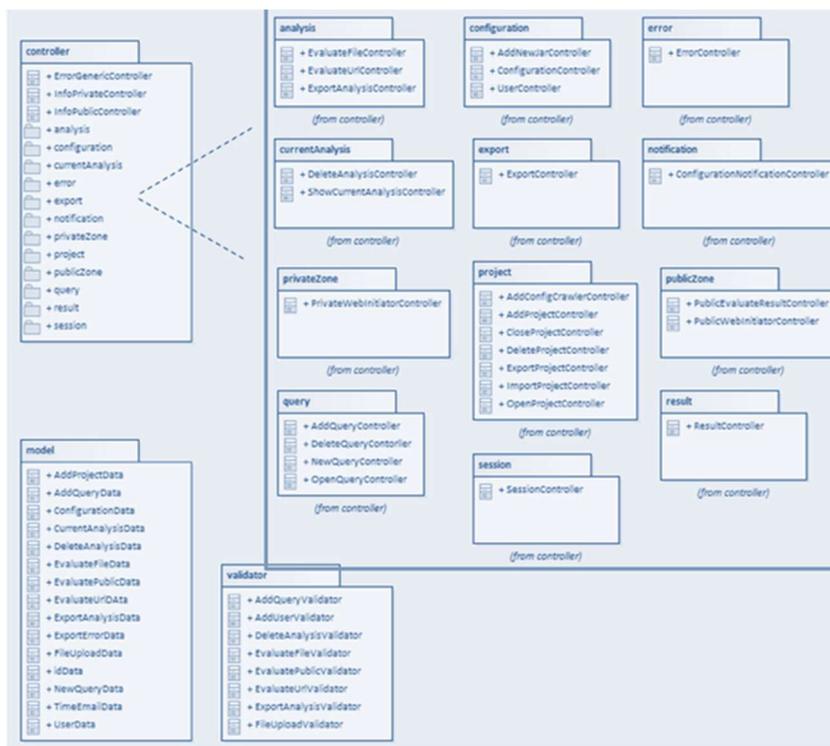


Imagen 53. Diagrama de clases del paquete presentation

Para mayor claridad del lector, a continuación, se van a mostrar las clases asociadas a un único ciclo de la capa de presentación, es decir, se mostrarán las clases necesarias para poder añadir una nueva consulta al sistema. De forma similar, en cuanto a presentación se refiere, se comportan las demás funcionalidades.

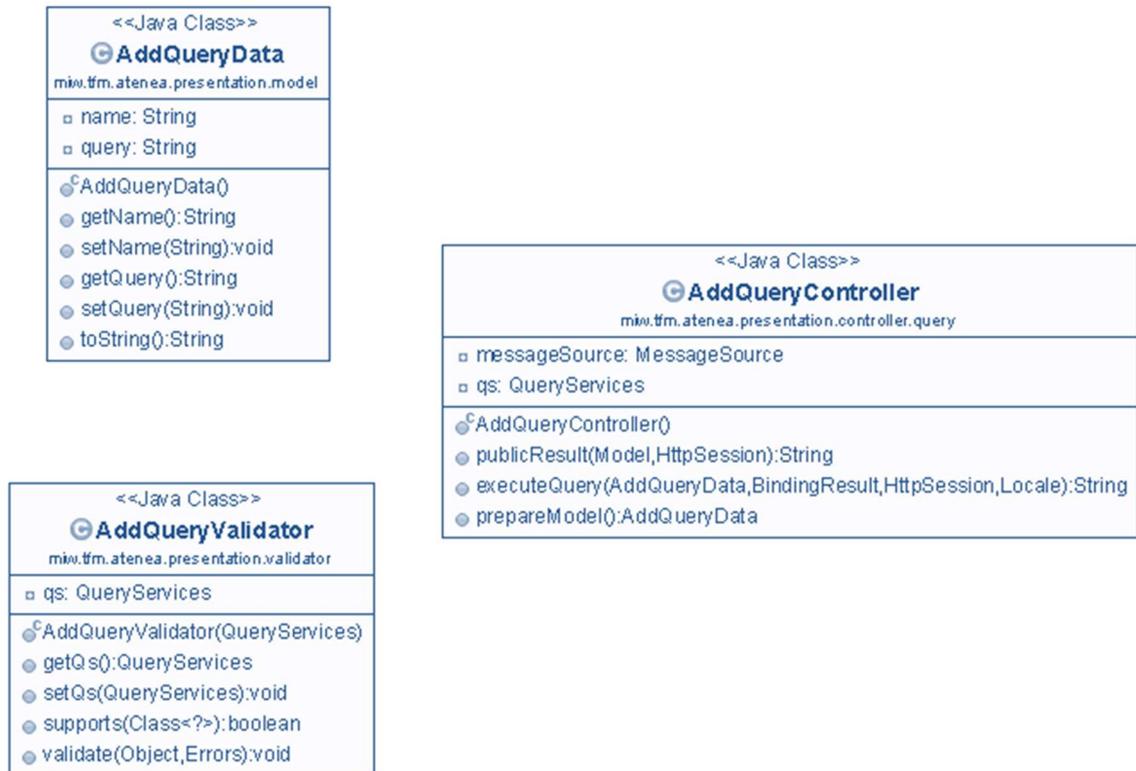


Imagen 54. Diagrama de clases de la acción añadir consulta

No todas las acciones que realiza el usuario necesitan, estrictamente, las clases para recoger o validar los datos de entrada, pero siempre es necesario un controlador que mapee la ruta introducida desde el navegador cliente para devolver la JSP correspondiente.

6.3 Diagramas de Interacción y Estados

6.3.1 Zona pública: Analizar un sitio web

El siguiente diagrama de interacción muestra el proceso que el sistema debe llevar a cabo para analizar un sitio web desde la zona pública. Este proceso comienza cuando el usuario introduce un enlace a analizar, una métrica y un crawler. Tras pasar por las fases correspondientes, el proceso finaliza mostrando la pantalla de resultados.

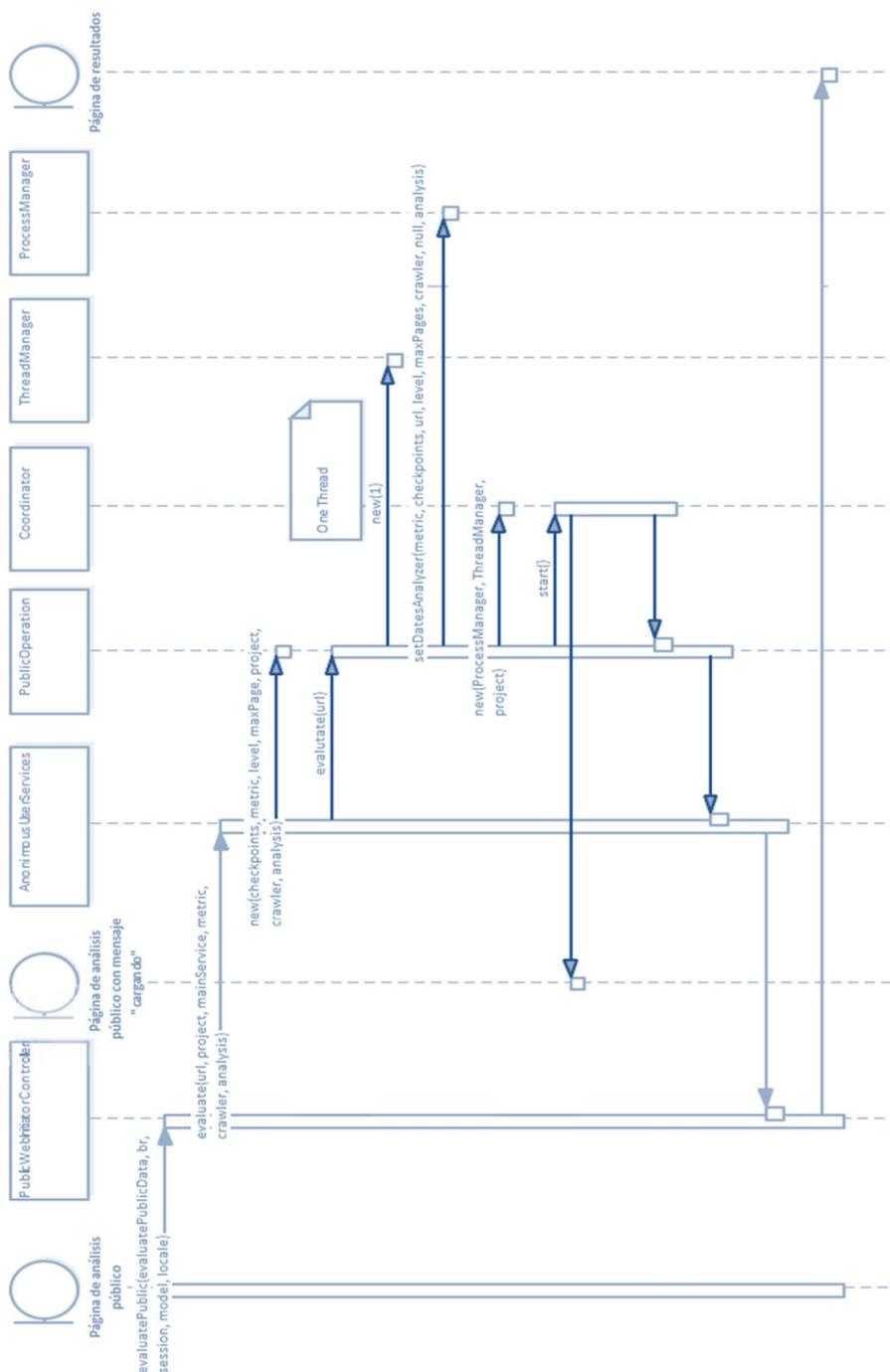


Imagen 55. Diagrama de interacción para un análisis público

1. El usuario introduce la URL a analizar desde la página principal de la aplicación web. También selecciona una métrica y un crawler.
2. Un objeto del modelo asociado a la capa de presentación recoge esta información y se la traspasa al *controller* correspondiente. El controlador *PublicWebInitiatorController* verifica que los datos son correctos e invoca al servicio necesario para iniciar la evaluación, en este caso, *AnonymousUserServices*.
3. Dentro de *AnonymousUserServices* existe un método *evaluate* que se invoca para iniciar la evaluación. Este método maneja la clase *PublicOperation* que se encarga de toda la gestión de una evaluación desde la zona pública, con los datos que recibe por parámetro.
4. *PublicOperation* crea un *ThreadManager* y este crea tantos *GenericThread* como número máximo de hilos ha recibido en el constructor. En el caso de la evaluación pública solo se crea un único hilo.
5. Para la métrica seleccionada, el *ProcessManager*, creado en la clase *PublicOperation*, crea y apila un único *Analyzer* al que le pasa los siguientes datos: crawler y configuración, métrica, profundidad, número máximo de páginas, configuración del crawler y también le pasa los checkpoints correspondientes a la métrica actual. En este caso el nivel de profundidad será siempre cero, es decir, solamente se analizará la página principal de la web dada. En este caso el número máximo a analizar será diez.
6. El *PublicOperation* inicia el hilo coordinador de procesos. El coordinador inicializa la traza, que está implementada sobre un patrón observer para conocer cuándo termina un *Analyzer*.
7. El coordinador coge el único *Analyzer* del *ProcessManager* y pide un *GenericThread* al *ThreadManager*. Si no hay ningún *GenericThread* libre se duerme durante un determinado tiempo y después lo vuelve a pedir. Si encuentra uno libre establece el *Analyzer* como proceso a ejecutar por el *GenericThread* y lo arranca, ejecutando éste último el método principal del *Analyzer*.
8. Nada más iniciarse la ejecución del *Coordinator* y, mientras dura la evaluación, el usuario verá un mensaje “cargando”.
9. Cuando termina la evaluación se muestran los resultados al usuario.

6.3.2 Zona privada: Analizar un único sitio web

El siguiente diagrama de interacción muestra el proceso que el sistema debe llevar a cabo para analizar un sitio web desde la zona privada. Para realizar este análisis, es necesario que el sistema cuente con un proyecto, previamente creado y abierto, ya que de él obtendrá todos los datos necesarios. Este proceso comienza cuando el usuario introduce un enlace a analizar desde la opción habilitada en la zona privada de Atenea.

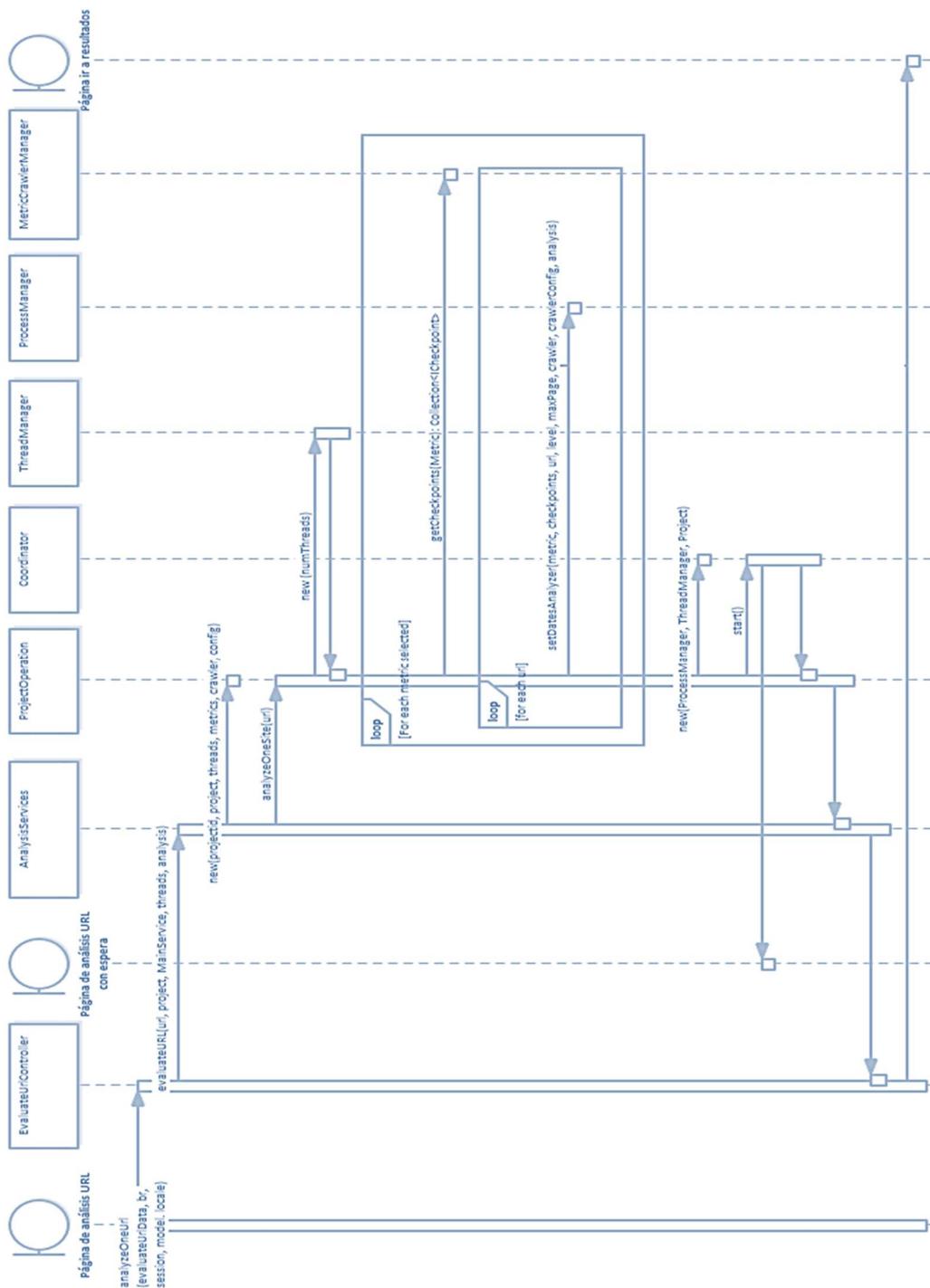


Imagen 56. Diagrama de interacción para analizar una única URL

1. Desde la página para analizar un único sitio web de la zona privada, el usuario introduce la URL a analizar, así como el número de hilos que desea. Como se ha mencionado, el resto de las características vendrán impuestas por el proyecto abierto en el sistema.
2. Un objeto del modelo asociado a la capa de presentación recoge esta información y se la traspasa al *controller* correspondiente. El controlador *EvaluateUrlController* verifica que los datos son correctos e invoca al servicio necesario para iniciar la evaluación, en este caso, *AnalysisServices*.
3. Dentro de *AnalysisServices* existe un método *evaluateUrl* que se invoca para iniciar la evaluación. Este método maneja la clase *ProjectOperation* que se encarga de toda la gestión de una evaluación desde la zona pública, con los datos que recibe por parámetro.
4. *ProjectOperation* crea un *ThreadManager* y este crea tantos *GenericThread* como número máximo de hilos ha recibido mediante la petición del usuario.
5. Por cada métrica seleccionada y por cada sitio, el *ProcessManager*, creado en la clase *ProjectOperation*, crea y apila un *Analyzer* al que le pasa los siguientes datos: crawler y configuración, métrica, profundidad, número máximo de páginas, configuración del crawler y también le pasa los checkpoints correspondientes a la métrica actual. La clase *ProjectOperation* es utilizada para analizar, tanto una única URL como un conjunto de ellas, por lo que, en este caso, solo se analizará un sitio por cada métrica.
6. El *ProjectOperation* inicia el hilo coordinador de procesos. El coordinador inicializa la traza, que está implementada sobre un patrón *observer* para conocer cuándo termina un *Analyzer*.
7. El coordinador coge un *Analyzer* del *ProcessManager* y pide un *GenericThread* al *ThreadManager*. Si no hay ningún *GenericThread* libre se duerme durante un determinado tiempo y después lo vuelve a pedir. Si encuentra uno libre establece el *Analyzer* como proceso a ejecutar por el *GenericThread* y lo arranca, ejecutando este último el método principal del *Analyzer*.
8. Nada más iniciarse la ejecución del *Coordinator* y, mientras dura la evaluación, el usuario verá un mensaje “cargando”.
9. Cuando termina la evaluación se muestra la página que indica que la operación ha tenido éxito.
10. A diferencia del análisis en la zona pública, el resultado y los errores de este análisis si quedan almacenados para poder consultarlos posteriormente.

6.3.3 Zona privada: Analizar un conjunto de URLs

El sistema también permite, desde la zona privada, analizar un conjunto de URLs. Para ello, se pide al usuario introducir un fichero en formato TXT con los enlaces a analizar separados por un retorno de carro. Como es habitual, para realizar este tipo de análisis, es necesario contar con un proyecto abierto y creado.

Este análisis cuenta con un nuevo sistema de avisos que permitirá al usuario despreocuparse.

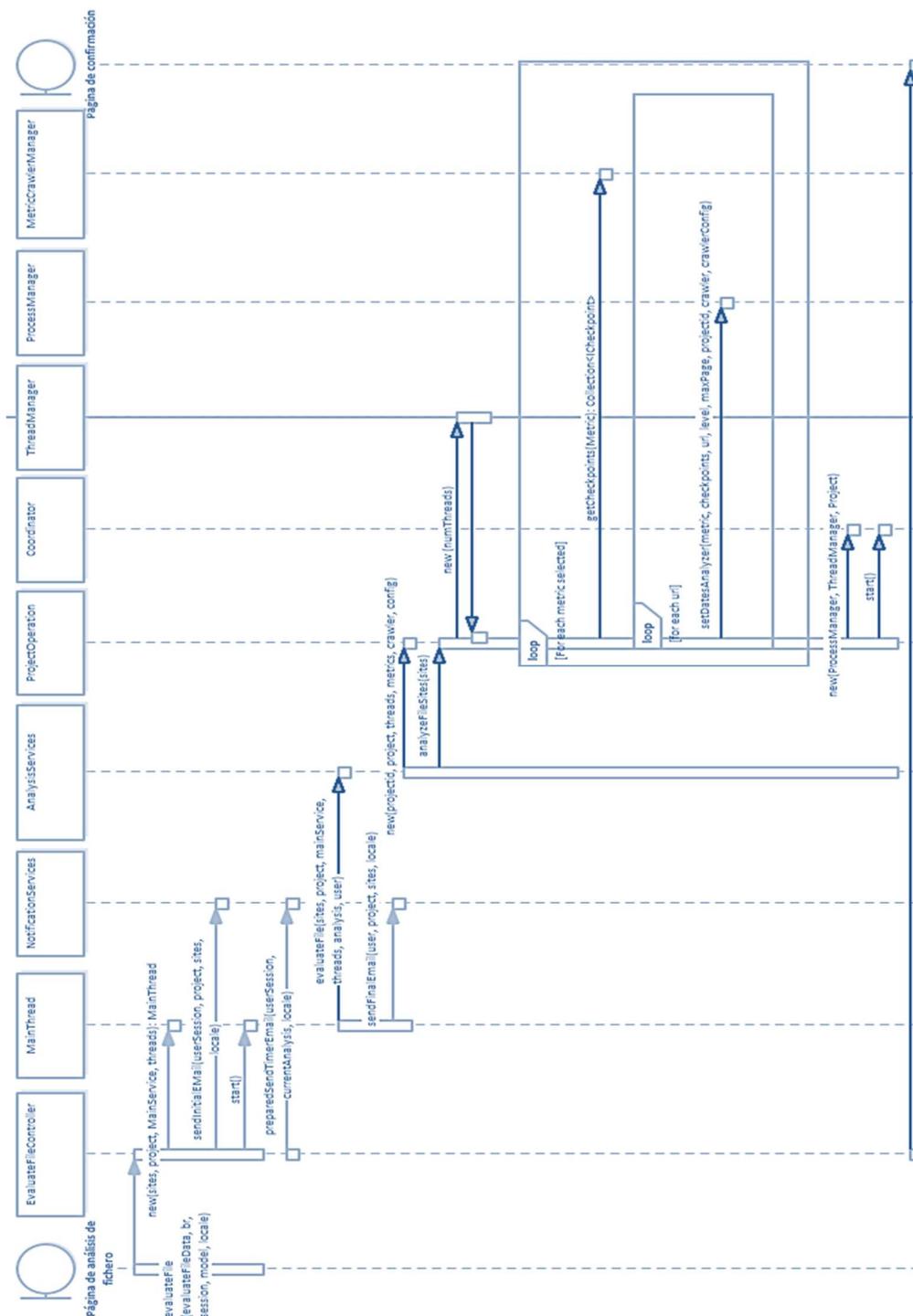


Imagen 57. Diagrama de interacción para analizar un conjunto de URLs

1. El usuario introduce la lista de URLs a analizar desde la página de análisis de la zona privada de la aplicación. En esta misma pantalla el usuario elige el número de hilos que realizarán el análisis. El resto de las características vendrán impuestas por el proyecto abierto/creado previamente por el usuario.
2. Un objeto del modelo asociado a la capa de presentación recoge esta información y se la traspa a la *controller* correspondiente. El controlador *EvaluateFileController* verifica que los datos son correctos e invoca al servicio necesario para iniciar la evaluación, en este caso, *AnalysisServices*.
3. Dentro de la capa de servicios *AnalysisServices* existe un método *evaluateFile* que se invoca para iniciar la evaluación. Este método maneja la clase *ProjectOperation* que se encarga de toda la gestión de una evaluación desde la zona privada.
4. Será un proceso en segundo plano el que se encargue de realizar la evaluación. En este punto el usuario ya habrá recibido la confirmación de que su análisis está en curso y podrá consultar el estado de su análisis en la pestaña “*Current Analysis*”.
5. *ProjectOperation* inicializa el *TreadManager* (con el número de hilos indicado durante la creación del análisis) y el *Coordinator*.
6. El *Coordinator* obtendrá los *Analyzers* para ejecutar. Concurrentemente se ejecutarán hasta el máximo número de hilos indicado para esta ejecución. En este punto, el *Coordinator* esperará a que termine alguno de los procesos de ejecución para empezar con la ejecución de los *Analyzer* restantes. Se creará un *Analyzer* diferente para cada sitio y métrica.
7. El *Analyzer* es el último encargado de realizar la ejecución de una URL con una determinada métrica. Almacenará todos los resultados obtenidos para cada caso.

En este diagrama aparece una nueva línea vertical que representa al componente encargado de enviar notificaciones.

Se enviará una notificación inicial al comenzar el análisis y otra final al terminar. Durante la ejecución de los análisis para cada métrica y cada sitio, el usuario recibirá un conjunto de notificaciones con el estado de los mismos, atendiendo a la frecuencia con la que este desea recibirlos.

6.3.4 Crear un nuevo proyecto

Con este diagrama se desea reflejar la interacción que existe entre la aplicación Atenea y la aplicación complementaria, que permite configurar los crawlers, a la hora de crear un nuevo proyecto.

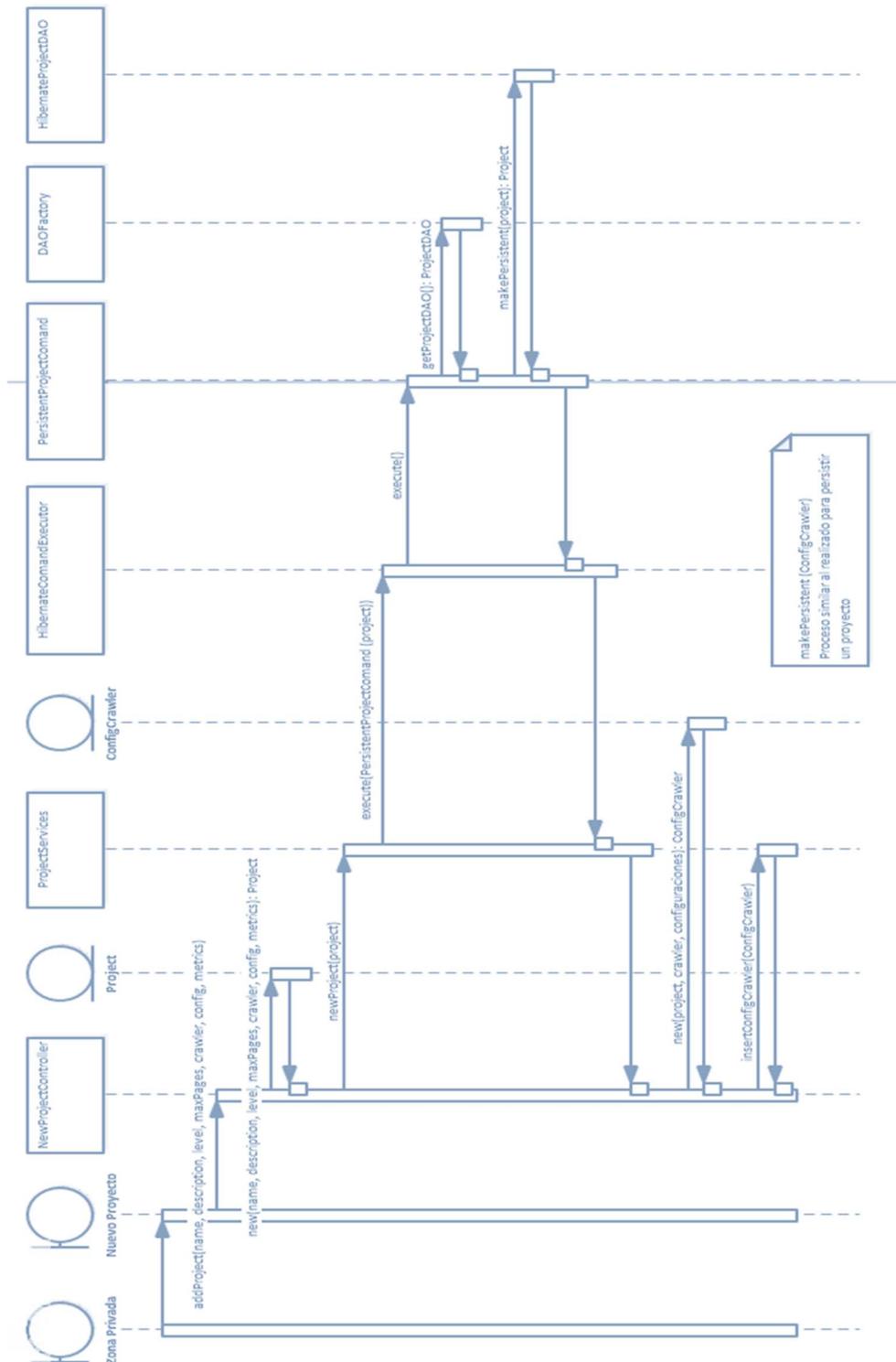


Imagen 58. Diagrama de interacción para crear un proyecto

1. El usuario registrado, accede a la zona privada de la aplicación y selecciona la pestaña *Proyectos*. Dentro de esta sección selecciona la opción de crear un nuevo proyecto.
2. En una nueva ventana se elige el nombre, descripción, número máximo de páginas para analizar, nivel de profundidad del análisis, crawler y métricas. También existe un botón que se utiliza para abrir la configuración del crawler seleccionado.
3. Una vez elegida la configuración del nuevo proyecto, se sigue adelante.
4. Un objeto del modelo asociado a la capa de presentación recoge esta información y se la traspasa al *controller* correspondiente. El controlador *AddProjectController* verifica que los datos son correctos y crea una nueva instancia de un objeto proyecto con dichos datos.
5. A continuación, se invoca a la capa de servicios para la gestión de los proyectos (*ProjectServices*). Dentro de esta capa se ofrece un servicio *newProject* que se encargará de todo el proceso de persistencia del proyecto en base de datos. Se utilizan las herramientas que ofrece *Hibernate* para persistir el proyecto.
6. Una vez creado el proyecto se crea una instancia de la configuración del crawler *ConfigCrawler*. La configuración del crawler también se guarda en base de datos.
7. La capa de servicios que gestiona los proyectos *ProjectServices* también ofrece un método para facilitar la inserción de una configuración de un crawler en base de datos. Por lo tanto, desde *NewProjectAction* se invoca este servicio *insertConfigCrawler*, el cual se encargará de todo el proceso de persistir la configuración del crawler. Este paso es similar al realizado anteriormente para insertar en base de datos un proyecto nuevo.

6.5 Diseño de la Base de Datos

Los sistemas de gestión de bases de datos son un conjunto de programas que permiten almacenar información para, posteriormente, poder modificarla, eliminarla o consultarla.

Existen sistemas de gestión de bases de datos tanto relacionales como no relacionales, esto es, mientras que unas trabajan los datos sobre tablas y sus distintas relaciones, las otras, trabajan los datos de forma independiente.

6.5.1 Descripción del SGBD Usado

Como sistema de gestión de base de datos se ha selecciona y utilizado el sistema relacional MySQL, en particular, una bifurcación de este sistema denominada mariaDB. Este sistema, desarrollado en el lenguaje de programación C++, es multiplataforma, multihilo, multiusuario y cuenta con un gran número de adeptos, gracias a las ventajas que proporciona:

- Alta velocidad al llevar a cabo las distintas transacciones.
- Bajo coste de utilización.
- Licencia pública.
- Fácil de instalar y configurar.

6.5.2 Integración del SGBD en Nuestro Sistema

Para realizar la comunicación entre la base de datos y el sistema, se implementa el patrón DAO, con el fin de encapsular las clases que acceden al sistema de gestión de base de datos, permitiendo el cambio de este, en un futuro.

En ocasiones resulta tedioso desarrollar el código necesario para poder recoger y almacenar información en la base de datos, por lo que, se ha utilizado JPA, ya que proporciona un sencillo puente entre el modelo de domino y el modelo relacional. En concreto, se ha utilizado *Hibernate* como herramienta de mapeo y se han anotado las clases del modelo dominio para establecer las relaciones pertinentes.

La configuración necesaria para comunicarse con la base de datos se sigue obtiene a través de un fichero declarativos XML. Todas las consultas necesarias, para el correcto funcionamiento del sistema, se crean utilizando las nuevas directrices impuestas por el lenguaje JPQL y se extraen todas a un fichero orm.xml, para mantenerlas juntas y organizadas.

6.5.3 Diagrama E-R

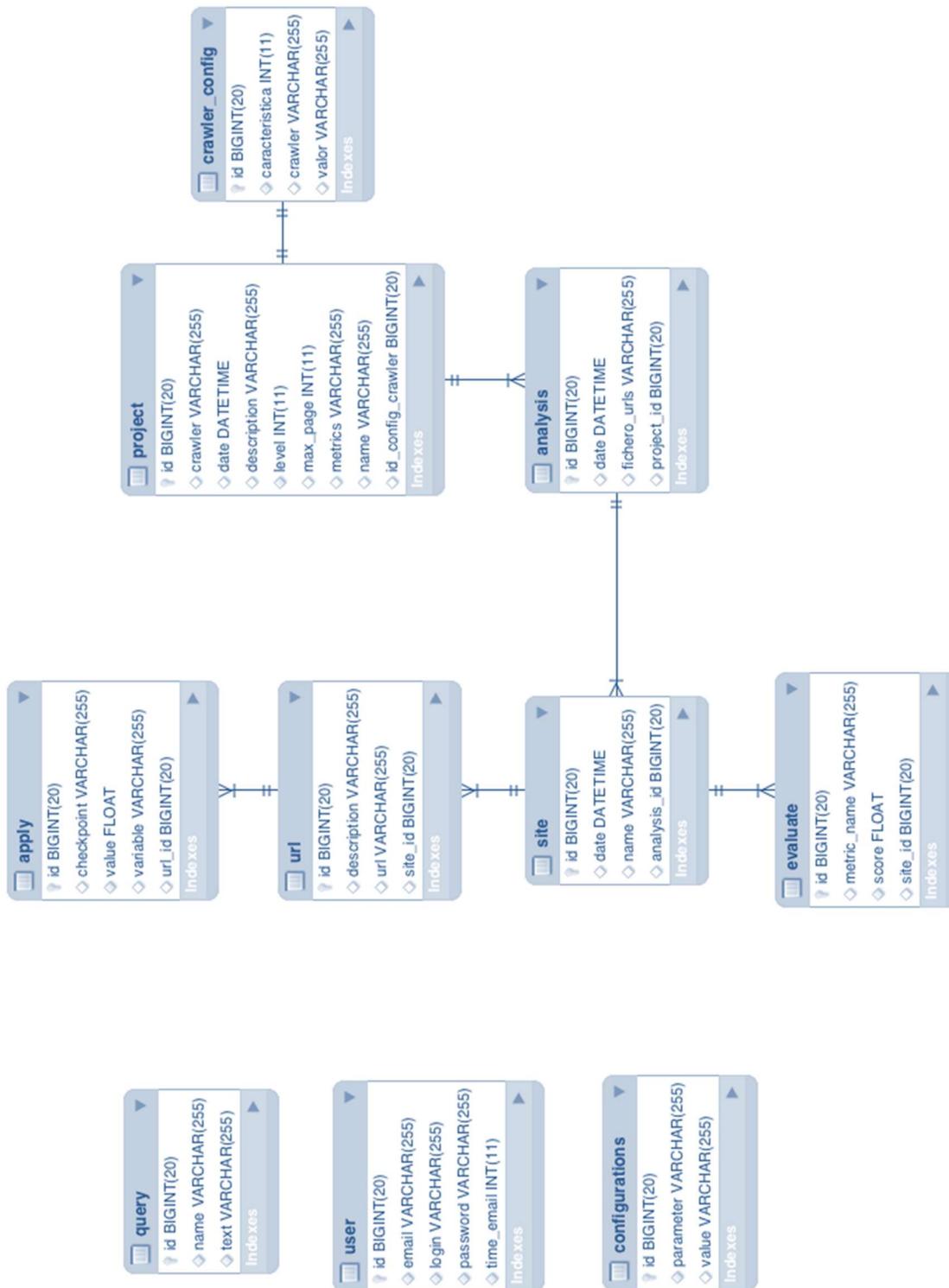


Imagen 60. Diagrama Entidad-Relación

6.6 Diseño de la Interfaz

El diseño de la interfaz se ha desarrollado siguiendo la estructura del sistema Atenea de partida, así como los modelos planteados en el análisis para las nuevas funciones.

6.6.1 Zona pública

Todas las páginas que componen la zona pública de la aplicación cuentan con una estructura predeterminada y similar.

- **Cabecera:** Lugar en el que aparece el logotipo de la aplicación, el título y un pequeño menú que contiene algunas opciones importantes. Dentro de este menú aparecen las opciones para modificar el idioma de la aplicación.
- **Cuerpo:** Zona dedicada a alojar el contenido principal de cada página. Esta zona cambia, según las necesidades.
- **Pie:** Submenú con opciones secundarias, pero que deben estar presentes en la zona pública de la aplicación.

6.6.1.1 Pantalla principal

La pantalla principal de la zona pública de Atenea tiene el siguiente aspecto:

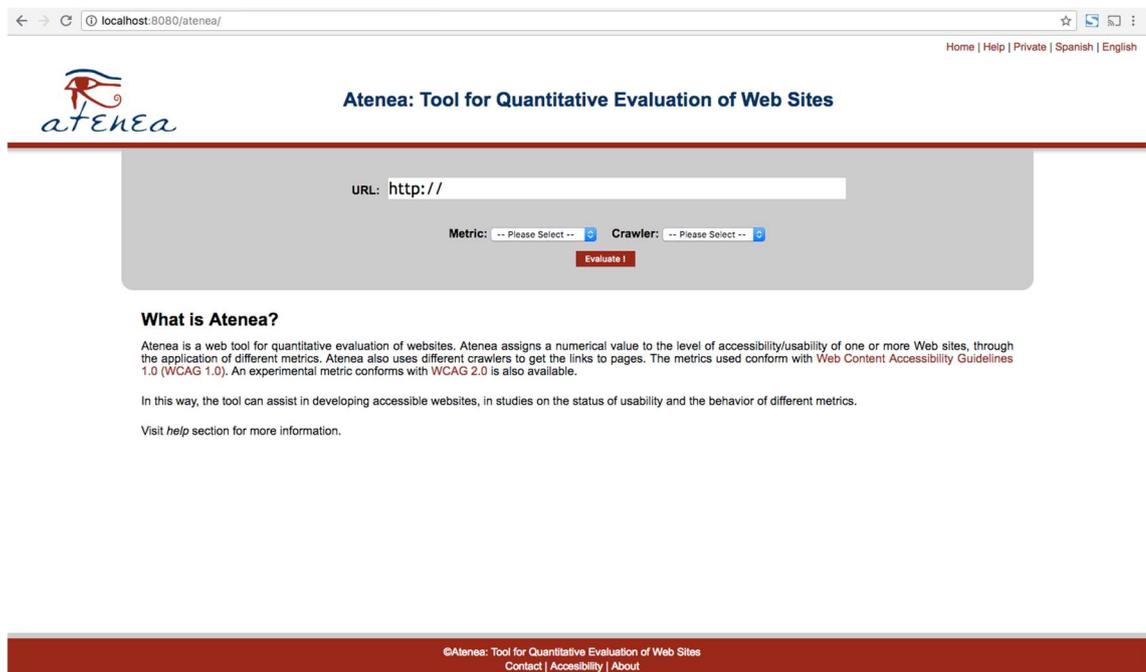


Imagen 61. Página principal zona pública

En este caso, el cuerpo de la página está destinado y estructurado, con el fin de poder realizar análisis de forma intuitiva. Además, cuenta con una pequeña explicación sobre la herramienta.

6.6.1.2 *Análisis en la zona pública*

En el Atenea de partida, mientras se realizaba un análisis, el sistema mostraba una nueva página con la estructura predeterminada y con un mensaje “cargando”. Ahora el sistema mantiene al usuario en la misma página del análisis para mostrar esta acción.

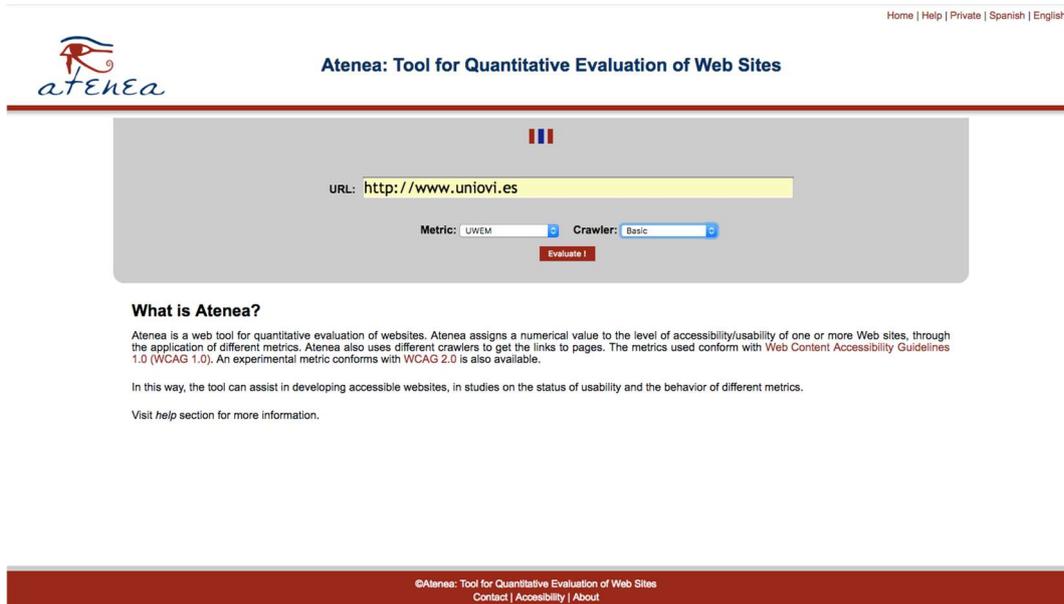


Imagen 62. Cargando análisis zona pública

Finalmente, transcurrido un tiempo, se mostrará la página con el resultado.

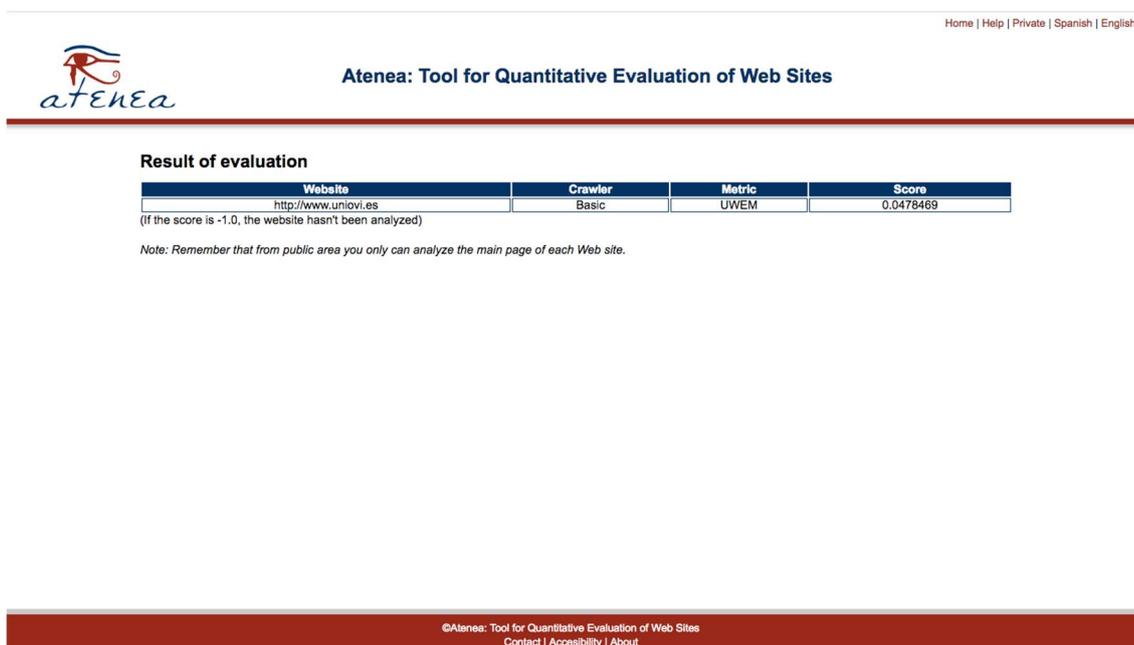


Imagen 63. Página de resultados zona pública

Como se puede observar, en este caso, la sección del cuerpo ya no contiene un formulario, sino una tabla con los resultados correspondientes.

6.6.1.3 Pantalla de inicio de sesión

Para acceder a la pantalla de inicio de sesión, desde la zona pública de la aplicación, debe pulsarse la opción Private situada en la parte superior derecha de la pantalla.

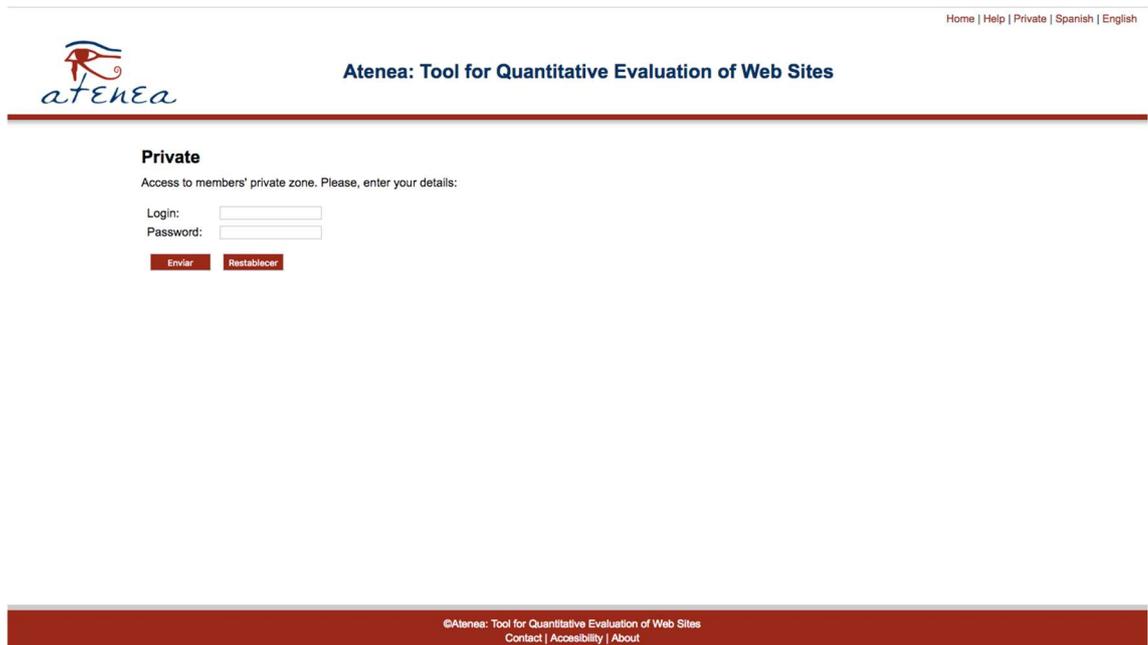


Imagen 64. Página de inicio de sesión

6.6.1.4 Otras páginas

Dentro de la zona pública de la web, existen otras páginas representativas. Su diseño será similar a las páginas anteriores, modificando su sección cuerpo. A continuación, se muestra la página de ayuda, a modo de ejemplo y guía.

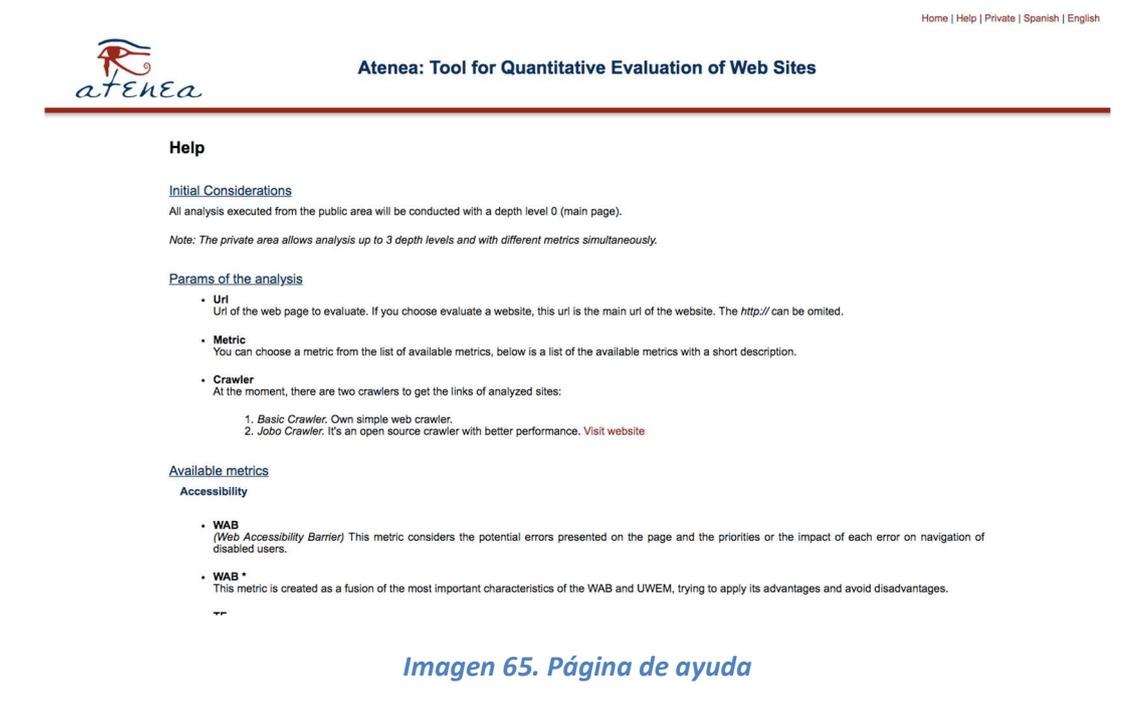


Imagen 65. Página de ayuda

6.6.2 Zona privada

Al igual que en la zona pública, la parte privada de la aplicación también presenta una estructura constante:

- **Cabecera:** Lugar en el que aparece el logotipo de la aplicación, el título y un pequeño sub-menú que contiene algunas opciones importantes.
- **Menú:** Incluye los elementos más importantes de la aplicación. Si no hay un proyecto abierto, algunos de los elementos estarán desactivados.
- **Cuerpo:** Zona donde se muestra toda la información de la sección elegida en el menú. Al igual que la zona pública, esta sección cambia, adaptándose a las necesidades de la funcionalidad.
- **Pie:** Información sobre la aplicación.

6.6.2.1 Pantalla principal

Si la validación de usuario y contraseña es correcta se le mostrará al usuario la página inicial de la zona privada que tendrá el aspecto que se muestra en la siguiente imagen.

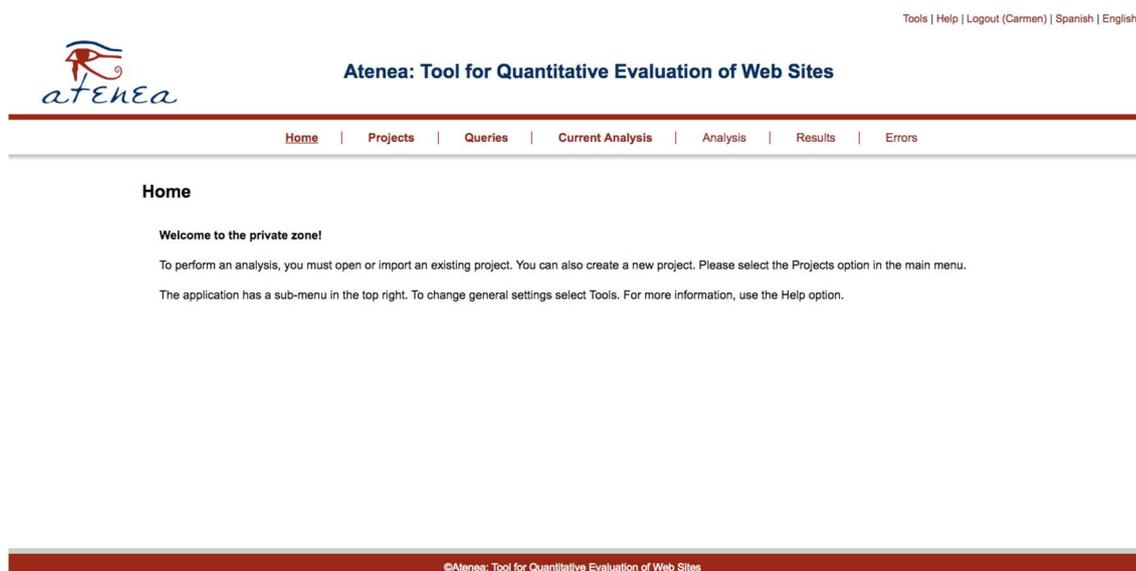


Imagen 66. Pantalla principal zona privada

En esta pantalla se da la bienvenida al usuario y se le indica la forma más habitual de uso de la aplicación. Se le ofrece también al usuario la posibilidad de visitar la sección de ayuda.

6.6.2.2 Sección de proyectos

Este elemento incluye las opciones de crear un nuevo proyecto, abrir un proyecto existente, eliminar un proyecto del sistema, importar y exportar un proyecto. A continuación, se muestra las páginas más significativas de esta sección y que dan al lector una idea de cómo es en conjunto esta sección. Esto es, las páginas destinadas a abrir, eliminar y exportar un proyecto presentan una estructura idéntica, mientras que la demás no.

6.6.2.2.1 Crear proyecto

Aquí el usuario debe introducir el nombre del proyecto, la descripción, escoger las métricas que desea, el nivel de profundidad del análisis, el máximo de páginas y el crawler. El botón Add Project iniciará el proceso de creación del nuevo proyecto.

The screenshot shows the 'Atenea: Tool for Quantitative Evaluation of Web Sites' interface. The top navigation bar includes 'Home', 'Projects', 'Queries', 'Current Analysis', 'Analysis', 'Results', and 'Errors'. The 'Projects' section is active, displaying a form for creating a new project. The form includes buttons for 'New Project', 'Open Project', 'Delete Project', 'Import Project', and 'Export Project'. Below these buttons, the user is prompted to 'Complete the following to create a new project:'. The form fields are: 'Name of project:' with a text input field; 'Short description:' with a text area; 'Metrics to use:' with a list of checkboxes for various metrics; 'Depth level:', 'Maximum pages:', and 'Crawler:' each with a dropdown menu. An 'Add project' button is located at the bottom of the form. The footer of the page reads '©Atenea: Tool for Quantitative Evaluation of Web Sites'.

Imagen 67. Pantalla nuevo proyecto

6.6.2.2.2 Importar proyecto

El cuerpo de esta sección contiene un formulario, donde se pueden adjuntar ficheros.

The screenshot shows the 'Atenea: Tool for Quantitative Evaluation of Web Sites' interface. The top navigation bar is the same as in the previous screenshot. The 'Projects' section is active, displaying a form for importing a project. The form includes buttons for 'Add project', 'Open Project', 'Delete Project', 'Import Project', and 'Export Project'. Below these buttons, the user is prompted to 'XML file:' with a file selection input field. An 'Import' button is located at the bottom of the form. The footer of the page reads '©Atenea: Tool for Quantitative Evaluation of Web Sites'.

Imagen 68. Pantalla importar proyecto

6.6.2.2.3 Abrir, eliminar y exportar proyecto

El cuerpo de estas secciones contiene una tabla con la información de cada proyecto a mostrar, así como una columna para poder seleccionar un proyecto y llevar a cabo la acción correspondiente.

Tools | Help | Logout (Carmen) | Spanish | English



Atenea: Tool for Quantitative Evaluation of Web Sites

Home
Projects
Queries
Current Analysis
Analysis
Results
Errors

Projects

[Add project](#) |
 [Open Project](#) |
 [Delete Project](#) |
 [Import Project](#) |
 [Export Project](#)
Search:

Show entries List projects to open

▲	Name of project	Short description	Data	Depth level	Maximum pages	Metrics	Crawler
<input type="radio"/>	Universidades	Universidades espa?olas a analizar	2018-06-18T14:49:25	1	10	EMPHASIZED TEXT,TOTAL IMAGES,TOTAL WORDS,WAB *	Basic
<input type="radio"/>	uno	uno	2018-06-15T12:27:14	0	5	WAB20,	Basic
<input type="radio"/>	a	a	2018-06-10T13:56:49	0	5	EMPHASIZED TEXT,IVORY,TEXT CLUSTER,TOTAL FONTS,TOTAL IMAGES,TOTAL WORDS,TE,UWEM,WAB *,WAB,WAB20,	Basic

Showing 1 to 3 of 3 entries Previous Next

©Atenea: Tool for Quantitative Evaluation of Web Sites

Imagen 69. Pantalla abrir proyectos

6.6.2.3 Sección de consultas

Este elemento incluye las opciones para hacer una consulta SQL a la base de datos (existe la opción de ver las tablas existentes en la base de datos), guardar una consulta para volver a realizarla en otro momento, abrir una consulta previamente guardada y eliminar una de estas consultas guardadas.

Al igual que en la sección de proyectos, para mostrar el diseño definitivo de las interfaces de usuario dirigidas a las funciones relacionadas con consultas a la base de datos, se mostrarán dos de las principales pantallas, ya que las otras dos comparten la misma estructura, respectivamente.

6.6.2.3.1 Añadir consulta

El cuerpo de esta sección contiene un formulario, donde se pueden añadir una nueva consulta.

The screenshot displays the 'Queries' interface. At the top, there is a navigation bar with buttons for 'New Query', 'Add Query', 'Open Query', and 'Delete Query'. Below this, a tree view shows the database schema under 'DataBase Tables'. The tables listed are:

- evaluate**: id, metric_name, score, site_id
- project**: id, crawler, date, description, level, max_page, metrics, name, id_config_crawler
- site**: id, date, name, analysis_id
- uri**: id, description, uri, site_id
- apply**: id, checkpoint, value, variable, uri_id

 To the right of the tree view is a form for adding a new query. It features a label 'Name of query:' followed by a text input field with the placeholder 'Insert name query...'. Below this is a large text area for entering the SQL query, with the placeholder 'Insert SQL query...'. A red 'Add' button is positioned at the bottom right of the form.

Imagen 70. Pantalla añadir consulta

Como todas las páginas de la aplicación, cuenta con su cabecera, su menú principal y el pie.

El diseño de la interfaz de usuario de la página para ejecutar una nueva consulta es similar a esta, pero con la diferencia de que no cuenta con un campo para introducir el nombre de la consulta.

6.6.2.3.2 Abrir y eliminar consulta

El cuerpo de estas secciones contiene una tabla con la información de cada consulta a mostrar, así como una columna para poder seleccionar una de ellas y llevar a cabo la acción correspondiente.



Atenea: Tool for Quantitative Evaluation of Web Sites

Home | Projects | **Queries** | Current Analysis | Analysis | Results | Errors

Queries

New Query | Add Query | **Open Query** | Delete Query

Show 10 entries Search:

List queries to open	
Name of query	Query
All Evaluates	select * from evaluate;

Showing 1 to 1 of 1 entries Previous Next

Imagen 71. Pantalla abrir consulta

6.6.2.4 Sección análisis actual

Para observar los análisis que están ejecutándose, en este momento en el sistema, se debe visitar este elemento del menú. Se trata de los análisis que han sido iniciados partiendo de una lista de URLs.

No es necesario que esté abierto ningún proyecto para poder visitar esta sección. Cualquier usuario registrado puede ver los análisis existentes en el sistema.



Atenea: Tool for Quantitative Evaluation of Web Sites

Home | Projects | Queries | **Current Analysis** | Analysis | Results | Errors

Analysis in Progress

Opened Project: a created at 2018-06-10T13:56:49 (Close)

Show 10 entries Search:

Delete	Project	Date	Completed	User	Trace
<input type="checkbox"/>	a	2018-06-26T14:16:14.644	9.09	Carmen	

Showing 1 to 1 of 1 entries Previous Next

```

http://www.uniovi.es/ - WAB - Running
http://www.uniovi.es/ - WAB20 - Terminated
    
```

Imagen 72. Pantalla análisis actual

Para cada análisis existe la opción de ver su traza y de detenerlo.

6.6.2.5 Sección análisis

Cabe destacar que para tener este elemento activo es necesario haber abierto un proyecto. Es decir, para realizar un análisis es necesario tener un proyecto asociado a dicho análisis.

Este elemento incluye las opciones para hacer un nuevo análisis de una URL, un análisis de una lista de URLs o exportar los análisis existentes en este proyecto en diferentes formatos. A continuación, se muestra la página más significativa de esta sección, evaluar una lista de URLs:

Imagen 73. Pantalla realizar análisis desde fichero

6.6.2.6 Sección de resultados

Cabe destacar que para tener este elemento activo es necesario haber abierto un proyecto previamente. Es decir, para ver los resultados de un proyecto es necesario tener un proyecto asociado a dichos resultados.

En esta sección aparece una tabla con los correspondientes resultados asociados al proyecto. Esta tabla incluye un mecanismo que permite buscar y filtrar los resultados tanto por sitio web como por métrica. En cualquiera de los dos casos es posible exportar los datos.

WebSite	Metric	score
http://www.uniovi.es/	IVORY	0.643722
http://www.uniovi.es/	EMPHASIZED TEXT	4.47761
http://www.uniovi.es/	TOTAL FONTS	0.0
http://www.uniovi.es/	TEXT CLUSTER	24.0
http://www.uniovi.es/	UWEM	0.0478469
http://www.uniovi.es/	TOTAL IMAGES	370.0
http://www.uniovi.es/	TE	0.428571
http://www.uniovi.es/	WAB *	9.04217
http://www.uniovi.es/	TOTAL WORDS	657.0
http://www.uniovi.es/	WAB	4.04217

Imagen 74. Pantalla de resultados

6.6.2.7 Sección de errores

Cabe destacar que para tener este elemento activo es necesario haber abierto un proyecto previamente. Es decir, para ver los errores de un proyecto es necesario tener un proyecto asociado a dichos errores.

Este elemento muestra los informes de errores asociados a cada una de las métricas del proyecto. El informe puede variar según la métrica, ya que cada una puede implementar esta función. Si la métrica no tiene una forma personalizada de mostrar los errores, el sistema proporcionará una tabla similar a la sección anterior.

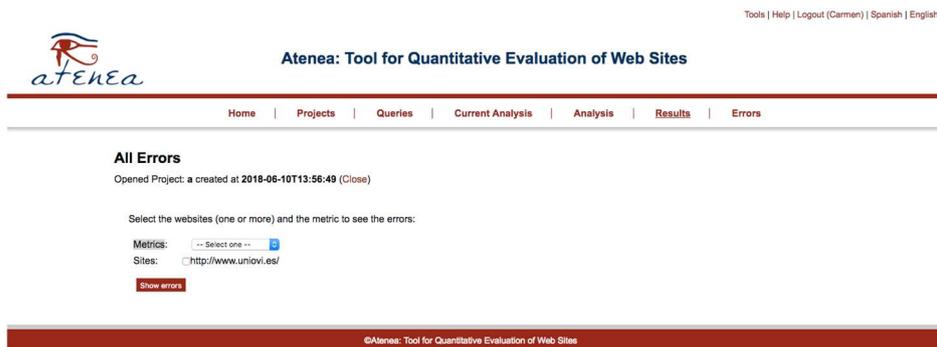


Imagen 75. Pantalla de errores

6.6.2.8 Sección de configuración

Este elemento, situado en la parte superior derecha de la pantalla, incluye las opciones de configuración básicas de la herramienta, la posibilidad de añadir un crawler o una métrica, la configuración de usuarios y, por último, la configuración de notificaciones. Todas estas opciones presentan una estructura muy similar. Tras haber diseñado el prototipo en la fase de análisis, finalmente, la sección para configurar las notificaciones presenta el siguiente aspecto.

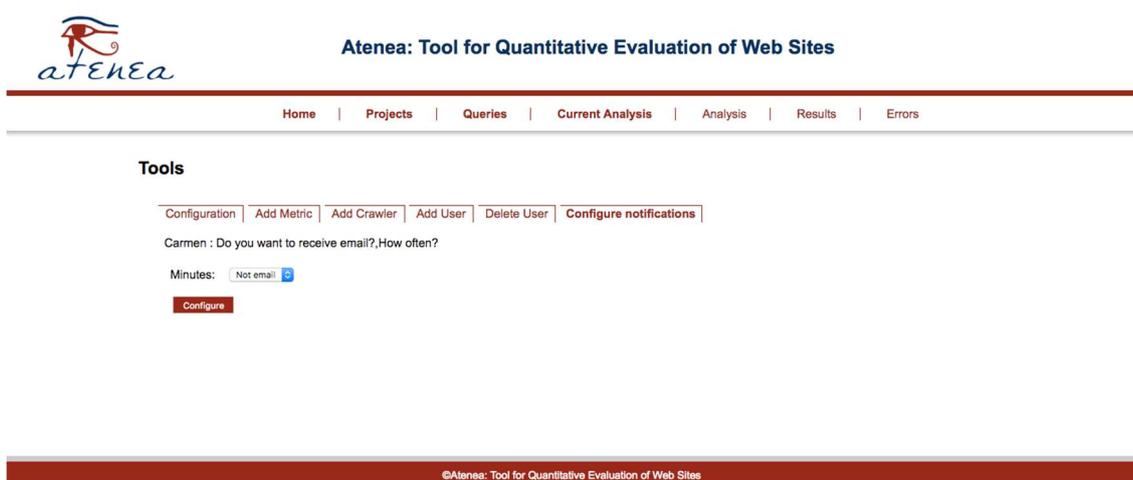


Imagen 76. Pantalla de configuración de notificaciones

6.6.2.9 Otros diseños

Finalmente, cabe destacar que el sistema cuenta con un mecanismo de recuperación, es decir, siempre que ocurre algún problema, el sistema es capaz de mostrar un mensaje y recuperarse.

Además, siempre que las funciones se ejecutan con éxito, el sistema muestra un mensaje al usuario, verificando la acción.



Imagen 77. Pantalla de verificación

6.7 Especificación Técnica del Plan de Pruebas

6.7.1 Pruebas de Integración y del Sistema

Escenario 1: Realizar un análisis desde la zona pública	
Prueba	Resultado Esperado
Realizar un análisis. Se introduce la URL, se selecciona la métrica y el crawler.	El sistema lleva al usuario a la pantalla de espera. Una vez finalizado el análisis, el usuario ve el resultado.
Prueba Técnica	
<ul style="list-style-type: none"> – URL: https://www.uniovi.es – Métrica: TE – Crawler: Basic 	
Prueba	Resultado Esperado
El usuario abandona la web antes de ver el resultado del análisis.	El sistema permanece sin cambios. El análisis continúa, pero el usuario nunca verá el resultado.
Prueba Técnica	
Se cierra el navegador.	
Prueba	Resultado Esperado
El usuario deja en blanco el campo de la URL.	El sistema muestra un error.
Prueba Técnica	
<ul style="list-style-type: none"> – URL: vacío – Métrica: TE – Crawler: Basic 	
Prueba	Resultado Esperado
El usuario introduce una URL con formato erróneo.	El Sistema muestra un error.
Prueba Técnica	
<ul style="list-style-type: none"> – URL: aabbcc – Métrica: TE – Crawler: Basic 	
Prueba	Resultado Esperado
El usuario no selecciona ninguna métrica o crawler.	El Sistema muestra un error.
Prueba Técnica	
<ul style="list-style-type: none"> – URL: https://www.uniovi.es – Métrica: vacío – Crawler: vacío 	

Tabla 100. Especificación técnica del plan de pruebas para análisis público

Escenario 2: Iniciar sesión	
Prueba	Resultado Esperado
El usuario se valida con su usuario y contraseña	El sistema dirige al usuario hacia la zona privada.
Prueba Técnica	
<ul style="list-style-type: none"> – Login: carmen – Password: 3123 	
Prueba	Resultado Esperado
El usuario no introduce usuario	El sistema muestra un error.

y/o contraseña.	
Prueba Técnica	
<ul style="list-style-type: none"> – Login: vacío – Password: vacío 	
Prueba	Resultado Esperado
El usuario introducido no existe en el sistema.	El sistema muestra un error.
Prueba Técnica	
<ul style="list-style-type: none"> – Login: noregistrado – Password: 3123 	
Prueba	Resultado Esperado
La contraseña del usuario es incorrecta.	El sistema muestra un error.
Prueba Técnica	
<ul style="list-style-type: none"> – Login: carmen – Password: incorrecta 	

Tabla 101. Especificación técnica del plan de pruebas para inicio de sesión

Escenario 3: Modificar configuración básica	
Prueba	Resultado Esperado
Se introducen todos los campos a modificar de forma correcta.	El sistema modifica la configuración básica de la herramienta.
Prueba Técnica	
<ul style="list-style-type: none"> – Nivel profundidad: 2 – Número máximo de páginas: 100 – Hilos por defecto: 15 	
Prueba	Resultado Esperado
Se deja algún un campo vacío.	El sistema devuelve un error.
Prueba Técnica	
<ul style="list-style-type: none"> – Nivel profundidad: vacío – Número máximo de páginas: 100 – Hilos por defecto: vacío 	
Prueba	Resultado Esperado
Cancelar la operación.	La configuración básica del sistema permanece sin cambios.
Prueba Técnica	
Se cambia de pantalla antes de presionar el botón actualizar.	
Prueba	Resultado Esperado
Se introducen valores numéricos negativos.	El sistema muestra un error.
Prueba Técnica	
<ul style="list-style-type: none"> – Nivel profundidad: 2 – Número máximo de páginas: 100 – Hilos por defecto: -15 	
Prueba	Resultado Esperado
Se introduce un carácter en un campo numérico.	El sistema no permite la acción.
Prueba Técnica	
<ul style="list-style-type: none"> – Nivel profundidad: 2 – Número máximo de páginas: 100 – Hilos por defecto: a 	
Prueba	Resultado Esperado
Para el nivel de profundidad, el usuario introduce	El sistema muestra un error.

un valor superior a 3.	
Prueba Técnica	
<ul style="list-style-type: none"> – Nivel profundidad: 4 – Número máximo de páginas: 100 – Hilos por defecto: 15 	

Tabla 102. Especificación técnica del plan de pruebas para configuración básica

Escenario 4: Añadir métrica	
Prueba	Resultado Esperado
Se sube un archivo JAR que contiene una métrica correctamente implementada.	El sistema muestra la confirmación de la operación realizada.
Prueba Técnica	
Se introduce la métrica correctamente implementada WAB20.jar	
Prueba	Resultado Esperado
Se sube un archivo JAR que no contiene una métrica correctamente implementada.	El sistema muestra un error.
Prueba Técnica	
Se introduce una métrica que no sigue los criterios necesarios.	
Prueba	Resultado Esperado
Se sube un archivo con un formato diferente al especificado.	El sistema no permite la acción.
Prueba Técnica	
Se intenta introducir el archivo universidades.txt .	
Prueba	Resultado Esperado
Cancelar la operación	El sistema permanece sin cambios.
Prueba Técnica	
Se cambia de pantalla antes de presionar el botón añadir.	
Prueba	Resultado Esperado
Se intenta subir una métrica que ya ha sido subida.	El sistema sobrescribe la información.
Prueba Técnica	
Se vuelve a introducir la métrica WAB20.jar	
Prueba	Resultado Esperado
No se introduce ningún archivo, es decir, el campo se deja vacío.	El sistema muestra un error.
Prueba Técnica	
Fichero: vacío	

Tabla 103. Especificación técnica del plan de pruebas para añadir una métrica

Escenario 5: Añadir crawler	
Prueba	Resultado Esperado
Se sube un archivo JAR que contiene un crawler correctamente implementado.	El sistema muestra la confirmación de la operación realizada.
Prueba Técnica	
Se introduce el crawler correctamente implementado Adapter.jar	
Prueba	Resultado Esperado
Se sube un archivo JAR que no contiene un crawler correctamente implementado.	El sistema muestra un error.

Prueba Técnica	
Se introduce un crawler que no sigue los criterios requeridos.	
Prueba	Resultado Esperado
Se sube un archivo con un formato diferente al especificado.	El sistema no permite la acción.
Prueba Técnica	
Se intenta introducir el archivo universidades.txt .	
Prueba	Resultado Esperado
Cancelar la operación	El sistema permanece sin cambios.
Prueba Técnica	
Se cambia de pantalla antes de presionar el botón añadir.	
Prueba	Resultado Esperado
Se intenta subir una crawler que ya ha sido subido.	El sistema sobrescribe la información.
Prueba Técnica	
Se vuelve a introducir el crawler Adapter.jar	
Prueba	Resultado Esperado
No se introduce ningún archivo, es decir, el campo se deja vacío.	El sistema muestra un error.
Prueba Técnica	
Fichero: vacío	

Tabla 104. Especificación técnica del plan de pruebas para añadir un crawler

Escenario 6: Añadir un usuario	
Prueba	Resultado Esperado
Se introducen todos los campos de forma correcta, para añadir un nuevo usuario.	El sistema muestra la confirmación de la operación.
Prueba Técnica	
<ul style="list-style-type: none"> – Nombre: nuevo – Correo electrónico: nuevo@gmail.com – Contraseña: nuevo – Repite contraseña: nuevo 	
Prueba	Resultado Esperado
Añadir un nombre de usuario existente en el sistema.	El sistema muestra un error.
Prueba Técnica	
<ul style="list-style-type: none"> – Nombre: nuevo – Correo electrónico: otro@gmail.com – Contraseña: nuevo – Repite contraseña: nuevo 	
Prueba	Resultado esperado
Añadir un correo electrónico existente en el sistema.	El sistema muestra un error.
Prueba Técnica	
<ul style="list-style-type: none"> – Nombre: otro – Correo electrónico: nuevo@gmail.com – Contraseña: nuevo – Repite contraseña: nuevo 	
Prueba	Resultado esperado
Se introduce un correo electrónico que no sigue el siguiente patrón aa@bb.cc	El sistema muestra un error.
Prueba Técnica	

Se introduce el siguiente correo electrónico: @gmail.com	
Prueba	Resultado Esperado
Dejar algún un campo sin rellenar.	El sistema muestra un error.
Prueba Técnica	
<ul style="list-style-type: none"> – Nombre: vacío – Correo electrónico: vacío – Contraseña: vacío – Repite contraseña: vacío 	
Prueba	Resultado Esperado
La contraseña y la repetición de la contraseña no coinciden.	El sistema muestra un error.
Prueba Técnica	
<ul style="list-style-type: none"> – Nombre: usuario – Correo electrónico: usuario @gmail.com – Contraseña: nuevo – Repite contraseña: viejo 	

Tabla 105. Especificación técnica del plan de pruebas para añadir usuarios

Escenario 7: Eliminar un usuario	
Prueba	Resultado Esperado
Eliminar un usuario de entre la lista de usuarios del sistema.	El sistema muestra la confirmación a la operación.
Prueba Técnica	
Se accede al listado de eliminación de usuarios para eliminar al primer usuario.	
Prueba	Resultado Esperado
Se accede a la sección de eliminar un usuario y no puede aparecer ni el usuario en sesión, ni el administrador.	El sistema no muestra estos dos usuarios.
Prueba Técnica	
Se accede al listado de eliminación de usuarios para eliminar al usuario Carmen.	
Prueba	Resultado Esperado
No se escoge ningún usuario a eliminar.	El sistema muestra un error.
Prueba Técnica	
Se presiona el botón eliminar sin haber seleccionado ningún usuario de la lista.	
Prueba	Resultado Esperado
Cancelar la operación.	El sistema no cambia.
Prueba Técnica	
Se cambia de pantalla antes de presionar el botón eliminar.	

Tabla 106. Especificación técnica del plan de pruebas para eliminar usuario

Escenario 8: Configuración de notificaciones	
Prueba	Resultado Esperado
El usuario escoge la frecuencia con la que desea obtener avisos. Puede escoger la opción de no recibir avisos.	El sistema actualiza esta información y muestra un mensaje de éxito.
Prueba Técnica	
El usuario está recibiendo notificaciones cada 15 minutos y se modifica el campo minutos del formulario a No deseo correos .	
Prueba	Resultado Esperado

Cancelar la operación.	El sistema permanece sin cambios.
Prueba Técnica	
Se cambia de pantalla antes de presionar el botón configurar.	

Tabla 107. Especificación técnica del plan de pruebas para configurar notificaciones

Escenario 9: Crear un proyecto	
Prueba	Resultado Esperado
Introducir y seleccionar los datos correctamente.	El sistema crea un nuevo proyecto, lo abre y muestra un mensaje de verificación.
Prueba Técnica	
<ul style="list-style-type: none"> – Nombre: Proyecto – Descripción: Soy un Proyecto nuevo – Métricas: WAB, TE, UWEM – Nivel de profundidad: 2 – Número máximo de páginas: 10 – Crawler: Básico – Configuración: - 	
Prueba	Resultado Esperado
No seleccionar o introducir algún los campos obligatorios.	El sistema no permite la creación del proyecto.
Prueba Técnica	
<ul style="list-style-type: none"> – Nombre: Proyecto – Descripción: vacío – Métricas: WAB, TE, UWEM – Nivel de profundidad: vacío – Número máximo de páginas: 10 – Crawler: vacío – Configuración: - 	
Prueba	Resultado Esperado
Cancelar la operación	El sistema permanece sin cambios.
Prueba Técnica	
Se cambia de pantalla antes de presionar el botón añadir.	

Tabla 108. Especificación técnica del plan de pruebas para crear un proyecto

Escenario 10: Abrir un proyecto	
Prueba	Resultado Esperado
Seleccionar un proyecto de entre la lista de proyectos del sistema.	El sistema muestra la confirmación a la operación y se abre un proyecto.
Prueba Técnica	
Se selecciona el proyecto con nombre Análisis de universidades para abrir.	
Prueba	Resultado Esperado
No se escoge ningún proyecto para ser abierto.	El sistema muestra un error.
Prueba Técnica	
Se presiona el botón abrir sin haber escogido ningún proyecto, previamente.	
Prueba	Resultado Esperado
Cancelar la operación.	El sistema no cambia.
Prueba Técnica	

Se cambia de pantalla antes de presionar el botón abrir.	
--	--

Tabla 109. Especificación técnica del plan de pruebas para abrir un proyecto

Escenario 11: Eliminar un proyecto	
Prueba	Resultado Esperado
Seleccionar un proyecto de entre la lista de proyectos del sistema.	El sistema muestra la confirmación a la operación y se elimina el proyecto.
Prueba Técnica Se selecciona el proyecto con nombre Análisis de universidades para eliminar.	
Prueba	Resultado Esperado
No se escoge ningún proyecto para ser eliminado.	El sistema muestra un error.
Prueba Técnica Se presiona el botón eliminar sin haber escogido ningún proyecto, previamente.	
Prueba	Resultado Esperado
Cancelar la operación.	El sistema no cambia.
Prueba Técnica Se cambia de pantalla antes de presionar el botón eliminar.	

Tabla 110. Especificación técnica del plan de pruebas para eliminar un proyecto

Escenario 12: Importar un proyecto	
Prueba	Resultado Esperado
Subir un archivo XML con un proyecto de Atenea.	El sistema crea un nuevo proyecto y lo abre.
Prueba Técnica Se sube un fichero XML con el siguiente contenido: <?xml version="1.0" encoding="UTF-8"?> <Project> <Name value="Jobo"/> <Description value="Jobo sin robots"/> <Metrics> <metric name="WAB20"/> <metric name="TOTAL IMAGES"/> <metric name="IVORY"/> </Metrics> <Depth_level value="0"/> <Max_number_pages value="5"/> <Crawler value="Jobo"> <characteristic id="1" value="true"/> </Crawler> </Project> En este proyecto se contemplan todos los posibles atributos de un proyecto. Una única línea.	
Prueba	Resultado Esperado
Subir un fichero con formato diferente a XML.	El sistema no permite esta acción.
Prueba Técnica Se sube un fichero TXT con el siguiente contenido: <?xml version="1.0" encoding="UTF-8"?>	

<pre><Project> <Name value="Jobo"/> <Description value="Jobo sin robots"/> <Metrics> <metric name="WAB20"/> <metric name="TOTAL IMAGES"/> <metric name="IVORY"/> </Metrics> <Depth_level value="0"/> <Max_number_pages value="5"/> <Crawler value="Jobo"> <characteristic id="1" value="true"/> </Crawler> </Project></pre>	
Prueba	Resultado Esperado
Subir un fichero en formato XML, pero con un contenido inadecuado.	El sistema no permite la creación del proyecto y muestra un mensaje de error.
Prueba Técnica	
<pre><?xml version="1.0" encoding="UTF-8"?> <Documento> <Proyecto value="Jobo"/> </Documento></pre>	
Prueba	Resultado Esperado
Cancelar la operación	El sistema permanece sin cambios.
Prueba Técnica	
Se cambia de pantalla antes de presionar el botón importar.	

Tabla 111. Especificación técnica del plan de pruebas para importar un proyecto

Escenario 13: Exportar un proyecto	
Prueba	Resultado Esperado
Seleccionar un proyecto de entre la lista de proyectos del sistema y pulsar el botón de exportar.	El sistema exporta al ordenador del cliente el fichero.
Prueba Técnica	
Se selecciona el proyecto con nombre Análisis de universidades para exportar.	
Prueba	Resultado Esperado
No se escoge ningún proyecto a exportar.	El sistema muestra un error y no se exporta ningún fichero.
Prueba Técnica	
Se presiona el botón exportar sin haber escogido ningún proyecto, previamente.	
Prueba	Resultado Esperado
Cancelar la operación.	El sistema no exporta nada al ordenador del cliente.
Prueba Técnica	
Se cambia de pantalla antes de presionar el botón exportar.	

Tabla 112. Especificación técnica del plan de pruebas para exportar un proyecto

Escenario 14: Cerrar un proyecto	
Prueba	Resultado Esperado
Seleccionar cerrar un proyecto.	El sistema muestra la confirmación

Prueba Técnica	de la operación y se cierra el proyecto.
El proyecto con nombre análisis de universidades está abierto y se puede analizar enlaces con él. Se decide cerrar dicho proyecto.	

Tabla 113. Especificación técnica del plan de pruebas para cerrar un proyecto

Escenario 15: Ejecutar una consulta SQL	
Prueba	Resultado Esperado
Ejecutar una consulta SQL correcta.	El sistema devuelve la información demandada.
Prueba Técnica SELECT * FROM project;	
Prueba	Resultado Esperado
Introducir una consulta SQL errónea.	El sistema devuelve un error.
Prueba Técnica SELECT * projects;	
Prueba	Resultado Esperado
No introducir ninguna consulta SQL, es decir, no rellenar un campo obligatorio.	El sistema devuelve un error.
Prueba Técnica Se presiona el botón ejecutar, sin haber introducido ninguna consulta en el campo correspondiente.	
Prueba	Resultado Esperado
Cancelar la operación.	El sistema permanece sin cambios.
Prueba Técnica Se cambia de pantalla antes de presionar el botón ejecutar.	
Prueba	Resultado Esperado
Intentar hacer un INSERT en una tabla de la base de datos.	El sistema devuelve un error.
Prueba Técnica INSERT INTO project (`id`, `crawler`, `date`, `description`, `level`, `max_page`, `metrics`, `name`) VALUES (1, 'Jobo', '2018-06-10 13:56:50', 'Prueba proyecto', '1', '5', 'TE', 'Proyecto');	

Tabla 114. Especificación técnica del plan de pruebas para ejecutar una consulta

Escenario 16: Añadir una consulta predeterminada	
Prueba	Resultado Esperado
Se introducen un nombre y una sentencia SQL correctos.	El sistema muestra la confirmación de la operación.
Prueba Técnica – Name: All projects – Sentencia: SELECT * FROM project;	
Prueba	Resultado Esperado
Añadir un nombre de consulta existente en el sistema.	El sistema muestra un error.
Prueba Técnica – Name: All projects – Sentencia: SELECT * FROM Project ORDER BY name;	
Prueba	Resultado esperado
Añadir una sentencia SQL existente en el sistema.	El sistema muestra un error.
Prueba Técnica – Name: All projects II	

– Sentencia: SELECT * FROM Project;	
Prueba	Resultado esperado
Se introduce una sentencia SQL con una sintaxis incorrecta.	El sistema muestra un error.
Prueba Técnica	
– Name: All projects – Sentencia: SELECT * project;	
Prueba	Resultado Esperado
Dejar algún un campo sin rellenar.	El sistema muestra un error.
Prueba Técnica	
– Name: vacío – Sentencia: SELECT * project;	
Prueba	Resultado Esperado
Cancelar la operación.	El sistema permanece sin cambios.
Prueba Técnica	
Se cambia de pantalla antes de presionar el botón añadir.	

Tabla 115. Especificación técnica del plan de pruebas para añadir una consulta

Escenario 17: Abrir una consulta	
Prueba	Resultado Esperado
Seleccionar una consulta de entre la lista de consultas existentes en el sistema.	El sistema ejecuta la consulta y muestra los resultados obtenidos.
Prueba Técnica	
Se selecciona la consulta con nombre All projects para abrirla.	
Prueba	Resultado Esperado
No se escoge ninguna consulta de la lista.	El sistema muestra un error.
Prueba Técnica	
Se presiona el botón abrir sin haber escogido ninguna consulta, previamente.	
Prueba	Resultado Esperado
Cancelar la operación.	El sistema no cambia.
Prueba Técnica	
Se cambia de pantalla antes de realizar las acciones que permiten abrir una consulta.	

Tabla 116. Especificación técnica del plan de pruebas para abrir una consulta

Escenario 18: Eliminar una consulta	
Prueba	Resultado Esperado
Seleccionar una consulta de entre la lista de consultas para eliminar.	El sistema muestra un mensaje de verificación y elimina la consulta.
Prueba Técnica	
Se selecciona la consulta con nombre All projects para eliminarla.	
Prueba	Resultado Esperado
No se escoge ninguna consulta de la lista.	El sistema muestra un error.
Prueba Técnica	
Se presiona el botón eliminar sin haber escogido ninguna consulta, previamente.	
Prueba	Resultado Esperado

Cancelar la operación.	El sistema no cambia.
Prueba Técnica	
Se cambia de pantalla antes de realizar las acciones que permiten eliminar una consulta.	

Tabla 117. Especificación técnica del plan de pruebas para eliminar consulta

Escenario 19: Ver el estado de los análisis en curso	
Prueba	Resultado Esperado
Seleccionar la pestaña de análisis en curso.	El sistema muestra todos los análisis en curso.
Prueba Técnica	
El usuario comienza un análisis y se dirige a la sección de análisis en curso para ver el estado de estos.	
Prueba	Resultado Esperado
Seleccionar la pestaña de análisis en curso cuando no existe ningún análisis en ejecución.	El sistema muestra un error.
Prueba Técnica	
Actualmente, no existe ningún análisis en curso, es decir, ni el usuario en sesión ni otro está ejecutando ningún análisis.	
Prueba	Resultado Esperado
Seleccionar la traza de un análisis en curso.	El sistema muestra la traza de dicho análisis.
Prueba Técnica	
Hay un análisis en curso y se presiona el botón de ver traza, con el fin de conocer qué enlaces han sido analizados y cuáles no y con qué métricas.	
Prueba	Resultado Esperado
Actualizar los análisis para ver el nuevo porcentaje completado.	El sistema actualiza la tabla con el nuevo porcentaje.
Prueba Técnica	
Hay un análisis en curso y se desea tener una idea del porcentaje de enlaces analizados y, por lo tanto, del resto. Para ello, se presiona el botón actualiza tabla de esta sección.	

Tabla 118. Especificación técnica del plan de pruebas para análisis en curso

Escenario 20: Detener un análisis en curso	
Prueba	Resultado Esperado
Seleccionar un análisis de la lista de los análisis en curso para detenerlo.	El sistema pide confirmación para detenerlo. Si la respuesta es afirmativa, lo detiene.
Prueba Técnica	
Hay varios análisis en curso y se selecciona uno para detenerlo.	
Prueba	Resultado Esperado
No se escoge ningún análisis de la lista.	El sistema muestra un error.
Prueba Técnica	
Se desea detener el análisis con nombre análisis de universidades, pero se presiona el botón detener antes de seleccionar dicho análisis.	
Prueba	Resultado Esperado
Cancelar la operación.	El sistema no cambia.
Prueba Técnica	
Se selecciona un análisis de la lista para eliminar, pero antes de presionar el botón detener, se cierra el navegador.	

Tabla 119. Especificación técnica del plan de pruebas para detener análisis

Escenario 21: Analizar una URL	
Prueba	Resultado Esperado
Introducir una URL correcta para analizar.	El sistema analiza la URL y devuelve el resultado.
Prueba Técnica	
Se abre un proyecto con las siguientes características: <ul style="list-style-type: none"> – Nivel de profundidad: 1 – Métrica: UWEM – Crawler: Basic – Número máximo de páginas: 10 Y se analiza el siguiente enlace: https://www.uniovi.es	
Prueba	Resultado Esperado
Introducir una URL con un formato erróneo.	El sistema muestra un error.
Prueba Técnica	
Se abre un proyecto con las siguientes características: <ul style="list-style-type: none"> – Nivel de profundidad: 1 – Métrica: UWEM – Crawler: Basic – Número máximo de páginas: 10 Y se analiza el siguiente enlace: aaabbbccc	
Prueba	Resultado Esperado
Dejar el campo de la URL vacío.	El sistema no permite esta acción.
Prueba Técnica	
Se abre un proyecto con las siguientes características: <ul style="list-style-type: none"> – Nivel de profundidad: 1 – Métrica: UWEM – Crawler: Basic – Número máximo de páginas: 10 Y se analiza el siguiente enlace: vacío	
Prueba	Resultado Esperado
Introducir números en la zona reservada para la URL	El sistema muestra un error.
Prueba Técnica	
Se abre un proyecto con las siguientes características: <ul style="list-style-type: none"> – Nivel de profundidad: 1 – Métrica: UWEM – Crawler: Basic – Número máximo de páginas: 10 Y se analiza el siguiente enlace: 1234	
Prueba	Resultado Esperado
Cerrar la aplicación sin haber terminado el análisis.	El análisis sigue ejecutándose en segundo plano. Cuando el análisis termina, el usuario puede ver el resultado.
Prueba Técnica	
Se abre un proyecto con las siguientes características: <ul style="list-style-type: none"> – Nivel de profundidad: 1 – Métrica: UWEM – Crawler: Basic – Número máximo de páginas: 10 Y se analiza el siguiente enlace: https://www.uniovi.es Antes de que el análisis finalice, se cierra la aplicación.	
Prueba	Resultado Esperado
Cerrar la ventana una vez terminado el análisis	El sistema permanece sin cambios.
Prueba Técnica	El resultado del análisis ya ha sido almacenado.
Se abre un proyecto con las siguientes características:	

<ul style="list-style-type: none"> – Nivel de profundidad: 1 – Métrica: UWEM – Crawler: Basic – Número máximo de páginas: 10 <p>Y se analiza el siguiente enlace: https://www.uniovi.es Al terminar el análisis, se cierra la aplicación.</p>	
---	--

Tabla 120. Especificación técnica del plan de pruebas para analizar una URL

Escenario 22: Analizar una lista de URLs	
Prueba	Resultado Esperado
Introducir un fichero TXT con un conjunto de URLs separadas por un retorno de carro.	El sistema comienza el análisis.
Prueba Técnica	
Se abre un proyecto con las siguientes características: <ul style="list-style-type: none"> – Nivel de profundidad: 1 – Métrica: UWEM – Crawler: Basic – Número máximo de páginas: 10 Se analiza un conjunto de URLs desde un fichero urls.txt con los siguientes enlaces: <ul style="list-style-type: none"> – http://www.uniovi.es/ – https://elpais.com/ 	
Prueba	Resultado Esperado
Introducir una lista de URLs con un formato incorrecto.	El sistema muestra un error.
Prueba Técnica	
Se abre un proyecto con las siguientes características: <ul style="list-style-type: none"> – Nivel de profundidad: 1 – Métrica: UWEM – Crawler: Basic – Número máximo de páginas: 10 Se analiza un conjunto de URLs desde un fichero urls.txt con los siguientes enlaces: <ul style="list-style-type: none"> – Aabc2313 – https://elpais.com/ 	
Prueba	Resultado Esperado
Introducir un fichero vacío.	El sistema muestra un error.
Prueba Técnica	
Se abre un proyecto con las siguientes características: <ul style="list-style-type: none"> – Nivel de profundidad: 1 – Métrica: UWEM – Crawler: Basic – Número máximo de páginas: 10 Se analiza un conjunto de URLs desde un fichero urls.txt con los siguientes enlaces: vacío	
Prueba	Resultado Esperado
Introducir un fichero con un formato diferente a	El sistema no permite esta acción.

TXT.	
Prueba Técnica	
Se abre un proyecto con las siguientes características: <ul style="list-style-type: none"> – Nivel de profundidad: 1 – Métrica: UWEM – Crawler: Basic – Número máximo de páginas: 10 Se analiza un conjunto de URLs desde un fichero urls.pdf con los siguientes enlaces: <ul style="list-style-type: none"> – http://www.uniovi.es/ – https://elpais.com/ 	
Prueba	Resultado Esperado
Si el usuario desea notificaciones y el análisis comienza.	
Prueba Técnica	
<ul style="list-style-type: none"> – Se abre un proyecto con las siguientes características: – Nivel de profundidad: 1 – Métrica: UWEM – Crawler: Basic – Número máximo de páginas: 10 Se analiza un conjunto de URLs desde un fichero urls.txt con los siguientes enlaces: <ul style="list-style-type: none"> – http://www.uniovi.es/ – https://elpais.com/ 	El sistema enviará varios correos electrónicos con la información asociada a dicho análisis.
Prueba	Resultado Esperado
Cerrar la aplicación sin haber terminado el análisis.	
Prueba Técnica	
Se abre un proyecto con las siguientes características: <ul style="list-style-type: none"> – Nivel de profundidad: 1 – Métrica: UWEM – Crawler: Basic – Número máximo de páginas: 10 Se analiza un conjunto de URLs desde un fichero urls.txt con los siguientes enlaces: <ul style="list-style-type: none"> – http://www.uniovi.es/ – https://elpais.com/ Antes de terminar el análisis, se cierra la aplicación.	El sistema continúa haciendo el análisis sin necesidad de atención por parte del usuario.
Prueba	Resultado Esperado
Cerrar la aplicación una vez terminado el análisis.	
Prueba Técnica	
Se abre un proyecto con las siguientes características: <ul style="list-style-type: none"> – Nivel de profundidad: 1 – Métrica: UWEM – Crawler: Basic – Número máximo de páginas: 10 Se analiza un conjunto de URLs desde un fichero urls.txt con los siguientes enlaces: <ul style="list-style-type: none"> – http://www.uniovi.es/ 	El sistema permanece sin cambios. Los resultados del análisis ya han sido almacenados.

– https://elpais.com/ Al terminar el análisis, se cierra la aplicación.	
--	--

Tabla 121. Especificación técnica del plan de pruebas para analizar un conjunto de URLs

Escenario 23: Exportar un análisis asociado un conjunto de URLs	
Prueba	Resultado Esperado
Seleccionar un análisis de entre la lista de los análisis ejecutados del sistema.	El sistema mostrará los resultados asociados a este análisis y permitirá su exportación.
Prueba Técnica	
Se abre un proyecto con las siguientes características: <ul style="list-style-type: none"> – Nivel de profundidad: 1 – Métrica: UWEM – Crawler: Basic – Número máximo de páginas: 10 Se analiza un conjunto de URLs desde un fichero urls.txt con los siguientes enlaces: <ul style="list-style-type: none"> – http://www.uniovi.es/ – https://elpais.com/ 	
Prueba	Resultado Esperado
No se escoge ningún análisis para ver y exportar sus resultados.	El sistema muestra un error.
Prueba Técnica	
Se accede a la sección de exportar análisis desde fichero y aparece un fichero urls.txt, antes de seleccionar dicho análisis se presiona el botón de ejecutar.	
Prueba	Resultado Esperado
Cancelar la operación.	El sistema no exporta nada al ordenador del cliente.
Prueba Técnica	
Se accede a la sección de exportar análisis desde fichero y aparece un fichero urls.txt, antes de seleccionar dicho análisis, el navegador se cierra inesperadamente.	

Tabla 122. Especificación técnica del plan de pruebas para exportar análisis

Escenario 24: Consultar y exportar resultados	
Prueba	Resultado Esperado
Seleccionar la pestaña de resultados para ver los resultados asociados a un proyecto.	El sistema muestra los resultados.
Prueba Técnica	
Se abre un proyecto con las siguientes características: <ul style="list-style-type: none"> – Nivel de profundidad: 1 – Métrica: UWEM – Crawler: Basic – Número máximo de páginas: 10 Se desea conocer el resultado de los análisis del enlace de la Universidad de Oviedo.	
Prueba	Resultado Esperado
Se intenta acceder a los resultados sin tener un proyecto abierto.	El sistema no permite esta acción, ya que no tiene la pestaña activa.
Prueba Técnica	
Se accede a la zona privada de la aplicación y sin abrir ningún proyecto existente, se intenta acceder a la zona de resultados.	

Prueba	Resultado Esperado
El sistema permite exportar los resultados y se selecciona el formato.	El sistema exporta el fichero con los resultados en el formato seleccionado al ordenador del cliente.
Prueba Técnica	
Se abre un proyecto con las siguientes características: <ul style="list-style-type: none"> – Nivel de profundidad: 1 – Métrica: UWEM – Crawler: Basic – Número máximo de páginas: 10 Se desea conocer el resultado de los análisis del enlace de la Universidad de Oviedo. Y, además, se exporta el análisis a formato XLS.	

Tabla 123. Especificación técnica del plan de pruebas para consultar resultados

Escenario 25: Consultar y exportar errores	
Prueba	Resultado Esperado
El usuario selecciona una métrica y uno o más sitios webs de forma correcta.	El sistema dependiendo de la métrica muestra o exporta los errores correctamente.
Prueba Técnica	
Se abre un proyecto con las siguientes características: <ul style="list-style-type: none"> Nivel de profundidad: 1 – Métrica: UWEM – Crawler: Basic – Número máximo de páginas: 10 Y se selecciona la métrica UWEM y el enlace de la universidad de Oviedo.	
Prueba	Resultado Esperado
No se selecciona la métrica o no se seleccionan los sitios o ambos a la vez.	El sistema no permite esta acción.
Prueba Técnica	
Se abre un proyecto con las siguientes características: <ul style="list-style-type: none"> – Nivel de profundidad: 1 – Métrica: UWEM – Crawler: Basic – Número máximo de páginas: 10 Y se desea exportar sus errores. No se selecciona ni la métrica y ningún enlace para conocer sus errores.	
Prueba	Resultado Esperado
No existen errores para una métrica o un determinado sitio.	El sistema muestra o exporta un listado vacío.
Prueba Técnica	
Se abre un proyecto con las siguientes características: <ul style="list-style-type: none"> – Nivel de profundidad: 1 – Métrica: UWEM – Crawler: Basic – Número máximo de páginas: 10 Y se selecciona la métrica UWEM y el enlace de la universidad de León.	
Prueba	Resultado Esperado
Cancelar operación	El sistema permanece sin cambios.
Prueba Técnica	

<p>Se abre un proyecto con las siguientes características:</p> <ul style="list-style-type: none"> – Nivel de profundidad: 1 – Métrica: UWEM – Crawler: Basic – Número máximo de páginas: 10 <p>Y se desea conocer si durante alguno de sus análisis se han producido errores, pero el navegador se cierra inesperadamente.</p>	
--	--

Tabla 124. Especificación técnica del plan de pruebas consultar errores

Escenario 26: Cerrar sesión	
Prueba	Resultado Esperado
Seleccionar cerrar sesión.	El sistema cierra la sesión y dirige al usuario a la página inicial de la aplicación.
Prueba Técnica	
Tras realizar varias funciones, se decide cerrar la sesión del sistema.	

Tabla 125. Especificación técnica del plan de pruebas cerrar sesión

6.7.2 Pruebas de Usabilidad y Accesibilidad

Se va a realizar un informe de auditoría, con la finalidad de medir aspectos relacionados con la usabilidad y accesibilidad.

Para dar una leve idea al lector de hasta donde se pretende llegar con este informe, en este apartado se van a describir las distintas tareas llevadas a cabo para la correcta redacción de dicho informe:

- Observación y análisis de la aplicación Atenea.
- Utilización de distintas herramientas de control de usabilidad y accesibilidad para escanear la aplicación.
- Reflexión acerca de aspectos a mejorar, modificar e incluso eliminar.

Además de comentar los aspectos en los que la web destaca y aquellos en los que falla, se dará una nota final global, es decir, contaremos con una evaluación tanto cuantitativa como cualitativa.

Las características principales a medir en un sitio web, atendiendo a estos aspectos citados son: usabilidad, accesibilidad, funcionalidad, contenidos, confiabilidad y, finalmente, eficiencia.

6.7.2.1 Usabilidad

Los aspectos que se van a estudiar atendiendo a la usabilidad son los siguientes:

- Comprensibilidad global del sitio.
- Mecanismos de ayuda y retroalimentación en línea.
- Aspectos de interfaces y estéticos.
- Adaptabilidad a dispositivos móviles y a idiomas.
- Usabilidad de los textos.
- Clasificación de la información y participación en redes sociales.

6.7.2.2 Accesibilidad

Los aspectos que se van a estudiar ateniendo a la accesibilidad son los siguientes:

- Accesibilidad para usuarios con discapacidades.
- Acceso a navegadores no gráficos.
- Se ejecutarán distintos test de accesibilidad.

6.7.2.3 *Funcionalidad*

Además de cumplir con pautas asociadas a la usabilidad y accesibilidad, los sitios web deben poner a disposición de los potenciales usuarios, una serie de funcionalidades que ayuden a mantener una experiencia enriquecida durante la utilización de los mismos.

Por esto, se comprobará si Atenea cuenta con mecanismos de búsqueda y se estudiarán aspectos de navegación y exploración.

6.7.2.4 *Contenidos*

Se analizará toda la información y temas tratados en la aplicación Atenea, con el fin de verificar que los contenidos de esta se adaptan al sentido de la aplicación. Es decir, se comprobará que la aplicación es fiel a lo que se espera de ella, que los contenidos ofrecidos se ajustan a aquello que la aplicación asegura proporcionar.

6.7.2.5 *Confiabilidad*

Para verificar la confiabilidad del sitio, se estudiarán aspectos como:

- Ausencia de deficiencias y errores.
- Utilización de estándares del W3C.
- Actualización periódica de la información.

6.7.2.6 *Eficiencia*

Las pruebas de eficiencia serán tratadas a parte en el apartado que sigue al presente.

6.7.3 Pruebas de Rendimiento

El sistema desarrollado consumirá una determinada cantidad de recursos (memoria y tiempo de proceso) que debemos procurar que sean los menores posibles. Por ello, la aplicación debe medirse para ver cuántos recursos consume y se debe intentar eliminar posibles cuellos de botella en el rendimiento que se puedan presentar en uno o varios subsistemas de la misma.

En su momento en la aplicación Atenea, desde la que se parte para desarrollar el presente proyecto, se realizaron pruebas de rendimiento y, en determinados casos, se llevaron a cabo las pertinentes modificaciones, con el fin de consumir la menor cantidad de recursos posibles.

Actualmente, ambas aplicaciones se encuentran en ejecución, es decir, la aplicación Atenea tanto inicial como la actual, están alojadas en varios servidores con las mismas características. Se realizarán varios análisis en igualdad de condiciones en ambas aplicaciones y, de este modo, se podrá verificar, de forma cuantitativa, como los cambios acometidos en esta aplicación han mejorado significativamente los tiempos de respuesta. Estos análisis se llevarán a cabo, desde la zona pública de la aplicación, así como desde su zona privada.

Finalmente, cabe destacar que, si se descubre algún fragmento de código con tiempos de ejecución muy grandes o complejidades muy altas se tratará de remediar esta situación refactorizando código, incluso si es necesario, buscando una nueva implementación.

Capítulo 7. Implementación del Sistema

7.1 Estándares y Normas Seguidos

En este apartado, se hace referencia a las distintas normas o estándares por las que se rige el presente proyecto:

- El sistema se ha desarrollado con el lenguaje de programación java, y éste, pone a disposición del programador un conjunto de directrices para crear un código limpio y elegante, estas directrices reciben el nombre de Java Code Conventions.
- Para crear páginas web, más usables y accesibles, se ha seguido una serie de estándares de diseño web. Entre estos se encuentra la recomendación WCAG (Apache Tomcat s.f.) (Web Content Accessibility Guidelines s.f.), que define un conjunto de normas que permiten adaptar las páginas web a personas con alguna discapacidad.
- Además, para la creación de las vistas que componen la aplicación, se han seguido los siguientes estándares: XHTML y CSS. El primero permite la creación de HTML expresado como XML válido. Es más estricto a nivel técnico, pero esto permite que posteriormente sea más fácil al hacer cambios o buscar errores entre otros. El segundo aporta una serie de reglas para la creación de hojas de estilo, que permitan formatear nuestra página web.

7.2 Patrones de diseño

Los patrones de diseño (Gamma, y otros 2009) son la base para la búsqueda de soluciones a problemas comunes en el desarrollo de software y otros ámbitos referentes al diseño de interacción o interfaces.

Un patrón de diseño es una solución a un problema de diseño no trivial que es efectiva (ya se resolvió el problema satisfactoriamente en ocasiones anteriores) y reusable (se puede aplicar a diferentes problemas de diseño en distintas circunstancias).

No es obligatorio utilizar los patrones siempre, solo en el caso de tener el mismo problema o similar que soluciona el patrón, siempre teniendo en cuenta que en un caso particular puede no ser aplicable. Abusar o forzar el uso de los patrones puede ser un error.

A continuación, se muestran una serie de patrones de diseño que han sido implementados en el desarrollo de este proyecto.

7.2.1 Patrón Command

7.2.1.1 Intención

Tener una interfaz abstracta de manejo de operaciones sobre cierto receptor, permitiendo a un cliente desarrollar las operaciones sin tener que conocer exactamente el tipo.

7.2.1.2 Motivo

A veces se quiere poder enviar solicitudes a objetos sin conocer exactamente la operación solicitada ni del receptor de la solicitud. En general un objeto, botón o menú ejecuta solicitudes, pero la solicitud no está implementada dentro del mismo.

7.2.1.3 Aplicación

Se puede decir que se utiliza en los siguientes casos:

- Encapsular un mensaje como un objeto, con lo que permite gestionar colas o registro de mensaje y deshacer operaciones.
- Soportar restaurar el estado a partir de un momento dado.
- Ofrecer una interfaz común que permita invocar las acciones de forma uniforme y extender el sistema con nuevas acciones de forma más sencilla.

7.2.1.4 Estructura

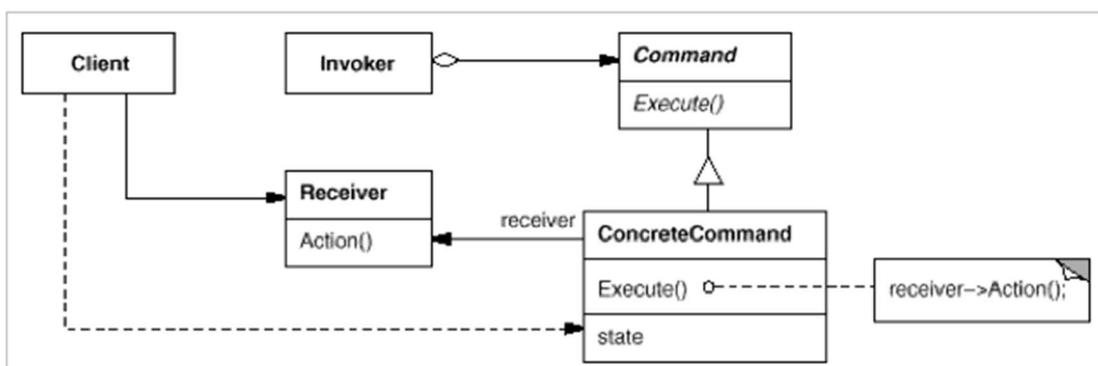


Imagen 78. Estructura del patrón Command

7.2.1.5 Uso dentro del proyecto

Este patrón se integra en el proyecto, con el fin de realizar las invocaciones a los diferentes DAOs. Se utiliza como intermediario entre la capa de negocio y la capa de persistencia.

7.2.2 Patrón Singleton

7.2.2.1 Intención

Obtener una única instancia de una clase, de forma que sólo se permita un único acceso a la misma.

7.2.2.2 Motivo

Modelo que garantiza que sólo haya una instancia de la clase en todo momento, y que el resto de los objetos pueden acceder a ella. Para ello, en lugar de tener una variable global, la instancia se almacena en un atributo estático de la clase y se accede a ella a través del método *instance()*, el cual también es un método estático de la clase.

7.2.2.3 Aplicación

Las situaciones más habituales de aplicación de este patrón son aquellos casos en los que se tenga que compartir recursos, es decir, aquellas situaciones en las que dicha clase controla el acceso a un recurso físico único o cuando cierto tipo de datos debe estar disponible para todos los demás objetos de la aplicación.

7.2.2.4 Estructura

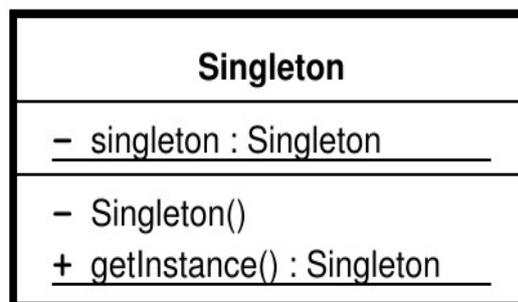


Imagen 79. Estructura del patrón Singleton

7.2.2.5 Uso dentro del proyecto

Este patrón se utiliza para permitir la creación de una única instancia de determinadas clases, como puede ser la clase que controla el fichero de log.

El objeto que controla la escritura en el fichero de log se utiliza desde varias capas de la aplicación, utilizando este patrón, se evita crear una instancia de este objeto por cada invocación al mismo. Se mantiene una única instancia y se permite su utilización desde cualquier implementación de la aplicación.

7.2.3 Patrón Facade

7.2.3.1 Intención

El patrón de diseño Facade sirve para proporcionar una interfaz unificada sencilla que haga de intermediaria entre un cliente y una interfaz o grupo de interfaces más complejas.

7.2.3.2 Motivo

Se trata de hacer una estructura fácilmente entendible que proporcione una forma de trabajar ordenada.

Se basa en proporcionar al programador una clase sencilla con un grupo de métodos, uno para cada operación permitida y de modo que sean estos métodos los que internamente hagan las operaciones con el fin de llevar a cabo la correcta lógica de la aplicación. A esta clase se le conoce como fachada.

7.2.3.3 Aplicación

Facade puede aplicarse en los siguientes casos, como principales, aunque también podrá usarse en otros muchos:

- Hacer una biblioteca de software más fácil de usar y entender, ya que Facade
- implementa métodos convenientes para tareas comunes.
- Hacer el código que usa la librería más legible, por la misma razón.
- Reducir la dependencia de código externo en los trabajos internos de una librería, ya que la mayoría del código lo usa Facade, permitiendo así más flexibilidad en el desarrollo de sistemas.
- Envuelve una colección mal diseñada de Apis con un solo API bien diseñado.

7.2.3.4 Estructura

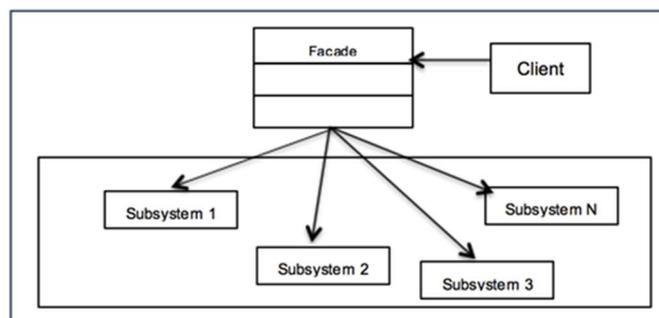


Imagen 80. Estructura del patrón Facade

7.2.3.5 *Uso dentro del proyecto*

Este patrón de diseño se ha utilizado para proporcionar una interfaz sencilla que haga de intermediaria entre las capas de presentación y de negocio; se proporcionan una serie de métodos que se invocan desde la otra capa.

De esta forma cuando un controlador de la capa de presentación necesite realizar alguna función lógica, se lo pedirá a su fachada asociada y no directamente a una clase de la capa de negocio. De esta forma cambios en las clases de una capa, no afectarán al resto.

En Atenea existe una fachada por cada conjunto de métodos asociados a una funcionalidad, por ejemplo, se crea una fachada que encapsula todos los métodos necesarios para implementar las acciones asociadas al envío de notificaciones.

7.2.4 Patrón Abstract Factory

7.2.4.1 *Intención*

Proporciona una interfaz para la creación de distintos tipos de objetos relacionados sin necesidad de especificar a qué clase concreta pertenecen.

7.2.4.2 *Motivo*

Un caso relativamente común de uso de este patrón se da en la creación de familias de interfaces gráficas en las cuales los elementos (productos) de la interfaz se mantienen constantes (por ejemplo labels, botones, cajas de texto...), pero el pintado de dichos elementos puede delegarse en distintas familias de forma que, en función de la fábrica seleccionada obtenemos unos botones u otros.

7.2.4.3 *Aplicación*

Este patrón se puede usar cuando:

- Un sistema debe ser independiente de los procesos de creación, composición y representación de sus productos.
- En situaciones en las que tenemos una familia de productos concretos y prevemos la inclusión de distintas familias de productos en un futuro.
- Los clientes sólo conozcan las interfaces de los productos y no sus implementaciones concretas.
- Se quiera proporcionar una librería de productos y solo se quieran revelar sus interfaces.

7.2.4.4 Estructura

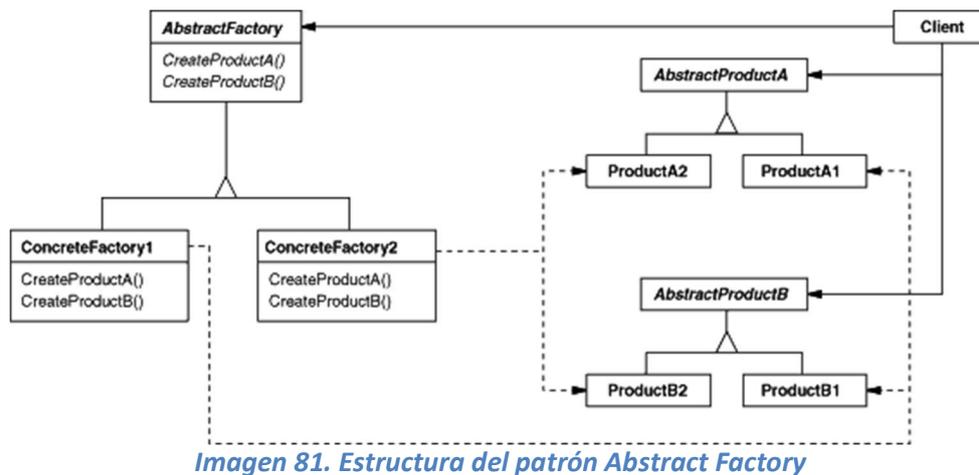


Imagen 81. Estructura del patrón Abstract Factory

7.2.4.5 Uso dentro del proyecto

Se usa para proporcionar una interfaz para la creación de distintos tipos de objetos, en el caso de este proyecto para los DAO, en este proyecto solo tendremos DAO de *Hibernate*, pero si hubiera otras clases DAO se proporcionaría esa interfaz.

7.2.5 Patrón DAO

7.2.5.1 Intención

Desacoplar la lógica de negocio de la lógica de acceso a datos de manera que se pueda cambiar la fuente de datos fácilmente.

7.2.5.2 Motivo

Almacenar y recuperar información persistente de diferentes sistemas de gestión de bases de datos.

7.2.5.3 Aplicación

El patrón *DAO* se puede usar para:

- Implementar un mecanismo para acceder y manipular los datos en un repositorio persistente.
- Desacoplar aplicación - repositorio.
- Proveer un *API* uniforme de acceso a datos que soporte varios tipos de repositorios.
- Organizar la lógica de acceso a datos para facilitar el mantenimiento y la portabilidad.

7.2.5.4 Estructura

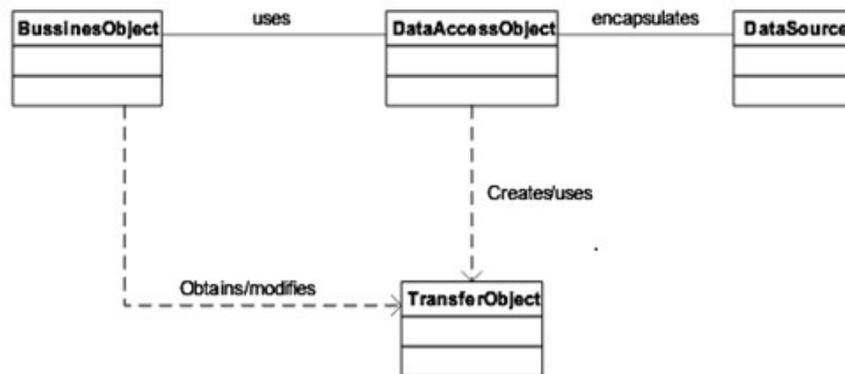


Imagen 82. Estructura del patrón DAO

7.2.5.5 Uso dentro del proyecto

Se utiliza en la capa de acceso a datos, que es la que comunica el programa con la base de datos, desacoplando así el acceso a datos de la capa de negocio dentro de la aplicación.

Actualmente, el proyecto se comunica con el sistema de gestión de bases de datos MySQL, pero se podría modificar en futuras implementaciones. Con la integración de este patrón, se podrán modificar y añadir nuevos sistemas de gestión de bases de datos. Esto hace que el acceso a los datos necesarios para el correcto funcionamiento de la aplicación sea totalmente independiente del motor de base de datos utilizado.

El patrón DAO trabaja de la siguiente manera: se crea una interfaz con las acciones que se desean realizar contra la base de datos por cada objeto del modelo de dominio. Por cada una de estas interfaces se crea una clase asociada que implementa cada uno de sus métodos. Es en estos métodos donde se realizan las consultas propias a la base de datos.

Además, las clases encargadas de realizar las consultas a la base de datos heredan de una clase que implementa una serie de métodos genéricos como son: añadir, eliminar, modificar y recuperar información.

7.2.6 Patrón Factory Method

7.2.6.1 Intención

Definir una interfaz para crear objetos de tipo genérico permitiendo a las subclases decidir qué tipo de objetos concretos crear. Es una simplificación del *Abstract Factory*, en la que la clase abstracta tiene métodos concretos que usan algunos de los abstractos; según se use una u otra hija de esta clase abstracta, se obtendrá uno u otro comportamiento.

7.2.6.2 *Motivo*

Crear objetos de tipo genérico permitiendo a las subclases decidir qué tipo de objetos concretos crear.

7.2.6.3 *Aplicación*

El patrón *Factory Method* se puede usar cuando:

- Una clase no puede anticipar el tipo concreto de objetos que debe crear.
- Una clase quiere que sus subclases especifiquen el tipo de objetos a crear
- Por el motivo que sea, se delega la responsabilidad de crear un tipo concreto de objeto.

7.2.6.4 *Estructura*

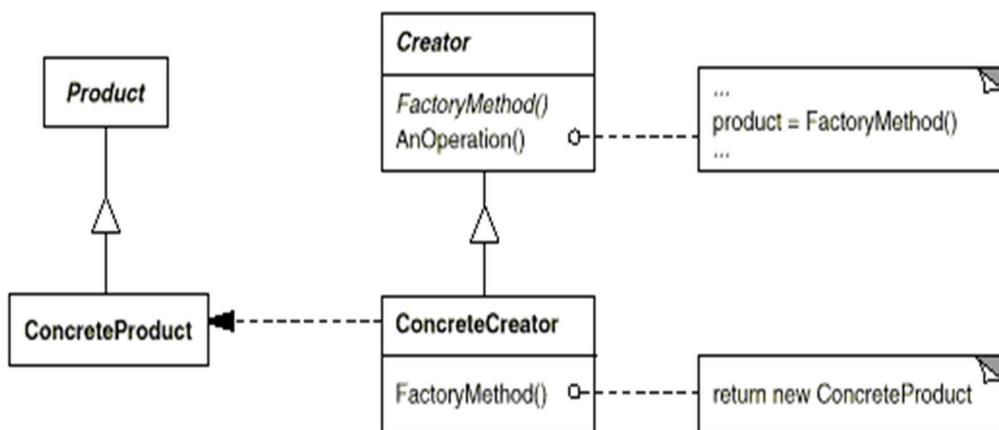


Imagen 83. Estructura del patrón Factory Method

7.2.6.5 *Uso dentro del proyecto*

El patrón *Factory Method* dentro del proyecto Atenea se utiliza para crear las factorías de los DAO de Hibernate, partiendo de una clase genérica.

Gracias a la integración de este patrón en Atenea, cuando la capa de negocio requiere utilizar una funcionalidad de la capa de persistencia, en vez de instanciarla directamente, se invocará a la factoría correspondiente que devolverá un objeto de la misma, permitiendo desacoplar aun mas la arquitectura.

7.2.7 Patrón Observer

7.2.7.1 Intención

Define una dependencia de uno a muchos entre objetos, de manera que cuando un objeto cambia todos los que tienen dependencia de él son informados.

7.2.7.2 Motivo

En casi todas las aplicaciones hay objetos que cuando cambian su valor deben informar a otros objetos para que cambien su estado en ese momento.

7.2.7.3 Aplicación

El patrón *Observer* se puede usar cuando:

- Una abstracción tiene diferentes aspectos y se desea reutilizar algunos de ellos de forma separada.
- Un cambio en un objeto necesita realizar cambios en otros y no se sabe a priori en cuantos objetos se hace este cambio.
- Un objeto debe notificar a otros de algún cambio y no se desea que se acoplen entre ellos.

7.2.7.4 Estructura

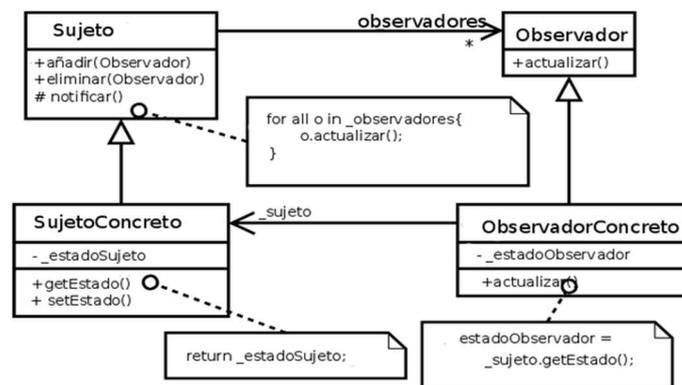


Imagen 84. Estructura del patrón observer

7.2.7.5 Uso dentro del proyecto

Este patrón se utiliza dentro de Atenea para la visualización de la traza durante los análisis del usuario. Cuando una evaluación termina, ella misma se encarga de notificar sobre su finalización.

7.2.8 Patrón MVC

7.2.8.1 Intención

Desacoplar el acceso a datos y la lógica de negocio (Modelo) de la presentación de los datos y la interacción con el usuario (Vista). Para ello, *MVC* introduce un componente intermedio, el *Controlador*.

7.2.8.2 Motivo

El motivo del uso de este patrón es el de desacoplar el acceso a los datos y la lógica de la aplicación de la presentación.

7.2.8.3 Aplicación

MVC se debe aplicar en los siguientes casos, prestar especial atención en no confundir con *3-Capas*:

- Separar los elementos de una aplicación en Modelo, Vista y Controlador.
- Desarrollo de *frameworks*
- Aplicaciones web

7.2.8.4 Estructura

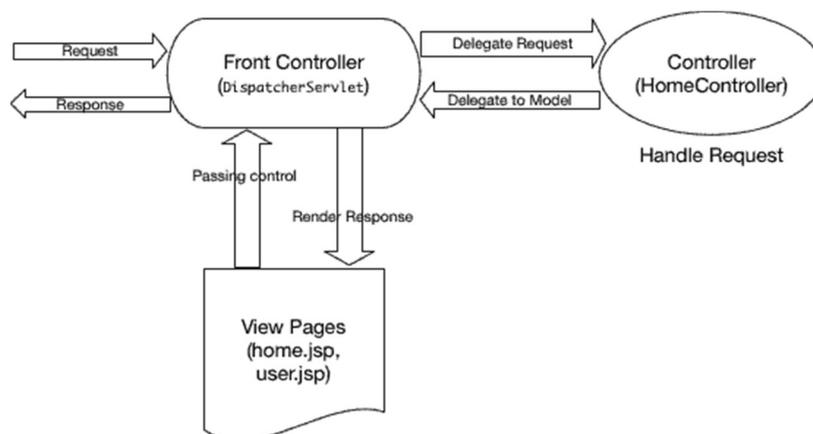


Imagen 85. Estructura patrón MVC

7.2.8.5 Uso dentro del proyecto

Para desarrollar la capa de presentación se ha utilizado el framework web Spring MVC, que presenta este tipo de arquitectura, con el fin de desacoplar la capa de presentación.

Para utilizar este patrón dentro del proyecto, la capa de presentación de Atenea encapsula un conjunto de componentes:

- Las diferentes JSP que conforman la vista o interfaz de usuario, que compone la información que se envía al cliente y los mecanismos interacción con éste.
- Los controladores que actúa como intermediario entre el Modelo y la Vista, gestionando el flujo de información entre ellos y las transformaciones para adaptar los datos a las necesidades de cada uno.
- El modelo que contiene una representación de los datos que maneja el sistema. Este modelo será utilizado, a su vez, por las demás capas que conforman la aplicación.
- Además, la utilización de este tipo de framework requiere la utilización de un modelo asociado, en exclusiva, a esta capa.

7.3 Lenguajes de Programación

Descripción de los lenguajes de programación usados para el desarrollo del presente sistema.

7.3.1 Java

El lenguaje de programación escogido, para el desarrollo del sistema Atenea, es Java más concretamente su versión 8.

Java es un lenguaje de programación orientado a objetos, multiplataforma y creado, principalmente, con el fin de desarrollar aplicaciones dirigidas a la web. Actualmente ha evolucionado y, este lenguaje, es utilizado para desarrollar cualquier tipo de proyecto.

Tiene una sintaxis muy parecida a la de C o C++, de hecho, podríamos decir que ambos lenguajes son los predecesores de Java, sin embargo, Java ha conseguido mejorar, con creces, sus propias expectativas.

Java ofrece unos mecanismos muy fuertes para garantizar la seguridad durante la ejecución del sistema. Esto, junto con su carácter multiplataforma ha hecho que se escoja.

7.3.2 JPQL y SQL

Para la interacción con la base de datos se ha utilizado tanto JPQL como SQL. Ambos lenguajes permiten especificar diversos tipos de operaciones sobre la base de datos. El primer lenguaje es utilizado por el propio sistema, es decir, todas las consultas declaradas en la aplicación se realizan mediante JPQL, ya que permite trabajar directamente con los objetos. El segundo es utilizado por los propios usuarios cuando desean añadir o ejecutar consultas desde la propia aplicación.

7.3.3 JavaScript y jQuery

Se ha utilizado JavaScript, así como una de sus bibliotecas, denominada jQuery. La integración de ambas permite crear páginas web dinámicas muy atractivas.

Java Script es un lenguaje de programación de alto nivel, orientado a objetos, dinámico, multiplataforma, que no necesita compilación y en el que las variables no necesitan ser introducidas antes de su uso y sus tipos se resuelven dinámicamente.

7.4 Herramientas y Programas Usados para el Desarrollo

7.4.1 Herramientas de ofimática

Para la elaboración del presente documento, así como para la distribución de la planificación y el posterior presupuesto, se han utilizado diversas herramientas de ofimática como son: Microsoft Office Word 2016, Microsoft Office Excel 2016 y Microsoft Office Project 2016.

7.4.2 Herramienta CASE

Se han utilizado varias hermanitas CASE para reflejar, de forma visual, las diferentes partes que conforman la arquitectura del sistema Atenea. Se han usado indistintamente diferentes hermanitas de esta familia, con el fin de dar claridad y comodidad a los posibles lectores de esta documentación.

Entre estas herramientas se encuentran: la versión 15 de Visual Paradigm, la versión 8 de Enterprise Architect y, finalmente, el plugin de eclipse ObjectAid.

7.4.3 Herramientas para el desarrollo del sistema

7.4.3.1 *Eclipse*

Eclipse es un entorno de desarrollo integrado de código abierto multiplataforma para desarrollar lo que el proyecto llama "Aplicaciones de Cliente Enriquecido", opuesto a las aplicaciones "Cliente-liviano" basadas en navegadores.

Para este proyecto se utiliza la versión Oxygen2.0 de Eclipse.

7.4.3.2 *MySQL*

Como sistema de gestión de base de datos se ha selecciona y utilizado el sistema relacional MySQL, en particular, una bifurcación de este sistema denominada mariaDB. Para este proyecto se ha utilizado la versión 10 de mariaDB.

7.4.3.3 *Hibernate*

En ocasiones resulta tedioso desarrollar el código necesario para poder recoger y almacenar información en la base de datos, por lo que, se ha utilizado JPA, ya que proporciona un sencillo puente entre el modelo de domino y el modelo relacional. En concreto, se ha utilizado Hibernate como herramienta de mapeo y se han anotado las clases del modelo dominio para establecer las relaciones pertinentes. Para este proyecto se ha usado la versión 5.

7.4.3.4 *Spring*

Como ya se ha mencionado, para el desarrollo y estructuración de la capa de presentación de Atenea se ha utilizado el framework web Spring MVC.

Para facilitar la creación y ejecución en desarrollo de este tipo de aplicaciones se ha utilizado un plugin de eclipse Spring Tool Suite (STS).

7.4.3.5 *Tomcat*

Como servidor de aplicaciones se ha utilizado la versión 8 de Tomcat.

7.4.3.6 *Atom*

Es un editor gratuito de código fuente, que soporta varios lenguajes de programación y se ejecuta en todos los sistemas operativos. Se ha utilizado su versión 1.28.0 para la edición de todo tipo de textos.

7.5 Creación del Sistema

7.5.1 Problemas Encontrados

El principal problema encontrado a la hora de llevar a cabo el presente proyecto, está directamente relacionado con el hecho de que no se trata de comenzar un proyecto desde cero, sino que se trabaja con un proyecto de punto de partida.

Este hecho hace que para poder llevar cabo modificaciones o añadir nuevas funcionalidades, se necesitará un largo periodo de tiempo en el que se estudiará el funcionamiento de la aplicación, así como la forma de implementarlo.

Es difícil trabajar con un código desarrollado por diferentes programadores. Aunque esto fue un problema, es necesario destacar que la correcta implementación y documentación del proyecto de partida ayudó mucho a resolver este problema.

7.5.2 Descripción Detallada de las Clases

A continuación, se incluye la información detallada de las clases más importantes de cada uno de los paquetes que comprenden la aplicación Atenea.

7.5.2.1 Clase HibernateComandExecutor

Nombre	Tipo	Descripción	Implementa de...
HibernateComandExecutor	-	Permite modificar el motor de persistencia.	ComandExecutor
<i>Responsabilidades</i>			
Número	Descripción		
1	Ejecuta un comando		
2	Permite que no se conozca que comando es previamente.		
<i>Métodos</i>			
Acceso Modo	Tipo de Retorno	Nombre	Parámetros y tipos
Público	Object	execute	Comand c
Observaciones			
Trabaja con el patrón Command			

Tabla 126. Clase HibernateComandExecutor

7.5.2.2 Clase HibernateDAOFactory

Nombre	Tipo	Descripción	Hereda de...
HibernateDAOFactory	-	Crea instancia de los objetos que interactúan con la base de datos.	DAOFactory
Responsabilidades			
Número	Descripción		
1	Es la factoría DAO para Hibernate		
2	Utilizada para instanciar cada uno de los DAOs de Hibernate		
Métodos			
Acceso Modo	Tipo de Retorno	Nombre	Parámetros y tipos
Público	AppliesDAO	getAppliesDAO	-
Público	EvaluateDAO	getEvaluateDAO	-
Público	SiteDAO	getSiteDAO	-
Público	UriDAO	getUriDAO	-
Público	UserDAO	getUserDAO	-
Público	ProjectDAO	getProjectDAO	-
Público	QueryDAO	getQueryDAO	-
Público	ConfigurationDAO	getConfigurationDAO	-
Público	AnalysisDAO	getAnalysisDAO	-
Público	ConfigCrawlerDAO	getConfigCrawlerDAO	-

Tabla 127. Clase HibernateDAOFactory

7.5.2.3 Clase Analyzer

Nombre	Tipo	Descripción	Hereda de...
Analyzer	-	Esta clase se encarga de analizar cada una de las URLs introducidas en el sistema	-
Responsabilidades			
Número	Descripción		
1	Evalúa la URL proporcionada por el usuario		
2	Almacena los resultados del análisis de una URL		
Métodos			
Acceso Modo	Tipo de Retorno	Nombre	Parámetros y tipos
Público	Coordinator	getCoordinator	-
Público	String	getMetricSelected	-
Público	String	getSiteName	-
Público	boolean	processUrl	-
Público	void	run	String y Site
Público	void	setCoordinator	Coordinator
Observaciones			
Cuenta con atributos que permiten a los métodos realizar sus funciones correctamente.			

Tabla 128. Clase Analyzer

7.5.2.4 Clase Coordinator

Nombre	Tipo	Descripción	Hereda de...
Coordinator	-	Esta clase maneja la ejecución de los hilos y procesos	Thread
Responsabilidades			
Número	Descripción		
1	Gestiona los <i>Analyzers</i> necesarios para una evaluación		
2	Crea la traza de ejecución		
3	Tiene en cuenta la posibilidad de múltiples ejecuciones simultáneas		
Métodos			
Acceso Modo	Tipo de Retorno	Nombre	Parámetros y tipos
Público	List<GenericThread>	getAliveThreads	-
Público	LocalDateTime	getDateBegin	-
Público	void	setDateBegin	LocalDateTime
Público	Project	getProject	-
Público	ObserverTrace	getTraza	-
Público	String	getUser	-
Público	boolean	isTerminate	-
Público	void	run	-
Público	void	setTerminate	boolean
Público	HashMap<String, Boolean>	sitesStatus	-
Observaciones			
Cuenta con atributos que permiten a los métodos realizar sus funciones correctamente.			

Tabla 129. Clase Coordinator

7.5.2.5 Clase ProjectOperation

Nombre	Tipo	Descripción	Hereda de...
ProjectOperation	-	Comprende el ciclo de un análisis en la zona privada	-
Responsabilidades			
Número	Descripción		
1	Gestiona la creación de un nuevo análisis para la zona privada de la web		
2	Análisis de una sola URL o de varias URLs		
Métodos			
Acceso Modo	Tipo de Retorno	Nombre	Parámetros y tipos
Público	void	analyzeOneSite	String
Público	void	analyzeFileSite	List<String>
Público	void	exportProjectXML	String
Público	List<Result>	getResultts	-
Observaciones			
Cuenta con atributos que permiten a los métodos realizar sus funciones correctamente.			

Tabla 130. Clase ProjectOperation

7.5.2.6 Clase *PublicOperation*

Nombre	Tipo	Descripción	Hereda de...
PublicOperation	-	Comprende el ciclo de un análisis en la zona pública	-
<u>Responsabilidades</u>			
Número	Descripción		
1	Análisis en la parte pública de la web		
<u>Métodos</u>			
Acceso Modo	Tipo de Retorno	Nombre	Parámetros y tipos
Público	void	evaluate	List<String>
Observaciones			
Una especie de <i>ProjectOperation</i> para la parte pública de la web			

Tabla 131. Clase *PublicOperation*

7.5.2.7 Clase *MetricCrawlerManager*

Nombre	Tipo	Descripción	Hereda de...
MetricCrawlerManager	-	Esta clase se encarga de cargar las métricas y los crawlers.	-
<u>Responsabilidades</u>			
Número	Descripción		
1	Carga las métricas y los crawlers del proyecto		
2	Carga informes de métricas y configuraciones de los crawlers		
3	Carga checkpoints asociados a métricas		
<u>Métodos</u>			
Acceso Modo	Tipo de Retorno	Nombre	Parámetros y tipos
Público	Collection<ICheckpoint>	getAllCheckpoints	-
Público	ArrayList<InterfaceWebModel>	getAllInterfaces	-
Público	Collection<IMetric>	getAllMetrics	-
Público	ArrayList<IReport>	getAllReports	-
Público	ICheckpoint	getCheckpoint	String
Público	String	getCheckpointNormative	String
Público	ArrayList<ICheckpoint>	getCheckpoints	-
Público	Collection<String>	getCheckpointsNames	-
Público	IConfig	getConfig	String
Público	ArrayList<ICrawler>	getCrawlers	-
Público	InterfaceWebModel	getInterface	String
Público	IMetric	getMetric	String
Público	ArrayList<ICheckpoint>	getMetricCheckpoints	String
Público	Collection<String>	getMetricCheckpointsNames	String
Público	String	getMetricNormative	String
Público	ArrayList<String>	getMetricsNames	-
Público	IReport	getReport	String
Público	void	setCheckpoints	List<ICheckpoint>

Observaciones			
Cuenta con atributos que permiten a los métodos realizar sus funciones correctamente.			

Tabla 132. Clase MetricCrawlerManager

7.5.2.8 Clase ServicesFactory

Nombre	Tipo	Descripción	Hereda de...
ServicesFactory	-	Crea instancias de objetos con funcionalidad de la capa de negocio	-
Responsabilidades			
Número	Descripción		
1	Ofrecer servicios ordenados por funcionalidad		
2	Intermediario entre presentación y negocio		
Métodos			
Acceso Modo	Tipo de Retorno	Nombre	Parámetros y tipos
Público	AnalysisServices	getAnalysisServices	-
Público	AnonimusUserServices	getAnonimusUserServices	-
Público	ConfigurationServices	getConfigurationServices	-
Público	ErrorServices	getErrorServices	-
Público	ExportServices	getExportServices	-
Público	ProjectServices	getProjectServices	-
Público	QueryServices	getQueryServices	-
Público	RegisterUserServices	getRegisterUserServices	-
Público	ResultServices	getResultServices	-

Tabla 133. Clase ServicesFactory

7.5.2.9 Clase ObservableTrace

Nombre	Tipo	Descripción	Hereda de...
ObservableTrace	-	Se responsabiliza de la ejecución del objeto Analyzer	Observable
Responsabilidades			
Número	Descripción		
1	Lanza Analyzer responsable de una ejecución y se bloquea		
2	Cuando termina notifica a los observadores		
Métodos			
Acceso Modo	Tipo de Retorno	Nombre	Parámetros y tipos
Público	void	start	-
Observaciones			
Trabaja con el patrón Observer			

Tabla 134. Clase ObservableTrace

Capítulo 8. Desarrollo de las Pruebas

8.1 Pruebas de Integración y del Sistema

A continuación, se ejecutan las pruebas funcionales ya diseñadas anteriormente y se anota el resultado obtenido, comparándolo con el resultado esperado, previamente justificado y documentado.

Escenario 1: Realizar un análisis desde la zona pública		
Prueba	Resultado Esperado	Resultado Obtenido
Realizar un análisis. Se introduce la URL, se selecciona la métrica y el crawler.	El sistema lleva al usuario a la pantalla de espera. Una vez finalizado el análisis, el usuario ve el resultado.	Tras la ventana de espera, aparece el resultado. En este caso el resultado es: 0.428571
Prueba Técnica <ul style="list-style-type: none"> – URL: https://www.uniovi.es – Métrica: TE – Crawler: Basic 		
El usuario abandona la web antes de ver el resultado del análisis.	El sistema permanece sin cambios. El análisis continúa, pero el usuario nunca verá el resultado.	Efectivamente, el sistema continúa con el análisis y nunca se llega a ver el resultado.
Prueba Técnica <ul style="list-style-type: none"> – Se cierra el navegador. 		
El usuario deja en blanco el campo de la URL.	El sistema muestra un error.	Efectivamente, el sistema no permite esta acción y muestra el siguiente mensaje: Por favor, introduzca un URL válida.
Prueba Técnica <ul style="list-style-type: none"> – URL: vacío – Métrica: TE – Crawler: Basic 		
El usuario introduce una URL con formato erróneo.	El Sistema muestra un error.	Efectivamente, el sistema no permite esta acción y muestra el siguiente mensaje: Por favor, introduzca un URL válida.
Prueba Técnica <ul style="list-style-type: none"> – URL: aabbcc – Métrica: TE – Crawler: Basic 		
El usuario no selecciona ninguna métrica o crawler.	El Sistema muestra un error.	Efectivamente, el sistema no permite esta acción y muestra el siguiente mensaje: Este campo no debe estar vacío.
Prueba Técnica <ul style="list-style-type: none"> – URL: https://www.uniovi.es – Métrica: vacío – Crawler: vacío 		

Tabla 135. Desarrollo plan de pruebas para análisis público

Escenario 2: Iniciar sesión		
Prueba	Resultado Esperado	Resultado Obtenido
El usuario se valida con su usuario y contraseña.	El sistema dirige al usuario hacia la zona privada.	Este usuario está registrado en el sistema, por lo que se le permite el acceso a la zona privada.
Prueba Técnica <ul style="list-style-type: none"> – Login: carmen – Password: 3123 		
Prueba	Resultado Esperado	Resultado Obtenido
El usuario no introduce usuario y/o contraseña.	El sistema muestra un error.	Efectivamente, el sistema no permite esta acción y muestra el siguiente mensaje: Usuario o contraseña inválidos.
Prueba Técnica <ul style="list-style-type: none"> – Login: vacío – Password: vacío 		
Prueba	Resultado Esperado	Resultado Obtenido
El usuario introducido no existe en el sistema.	El sistema muestra un error.	Efectivamente, el sistema no permite esta acción y muestra el siguiente mensaje: Usuario o contraseña inválidos.
Prueba Técnica <ul style="list-style-type: none"> – Login: noregistrado – Password: 3123 		
Prueba	Resultado Esperado	Resultado Obtenido
La contraseña del usuario es incorrecta.	El sistema muestra un error.	Efectivamente, el sistema no permite esta acción y muestra el siguiente mensaje: Usuario o contraseña inválidos.
Prueba Técnica <ul style="list-style-type: none"> – Login: carmen – Password: incorrecta 		

Tabla 136. Desarrollo plan de pruebas para inicio de sesión

Escenario 3: Modificar configuración básica		
Prueba	Resultado Esperado	Resultado Obtenido
Se introducen todos los campos a modificar de forma correcta.	El sistema modifica la configuración básica de la herramienta.	Se muestra el siguiente mensaje: La configuración ha sido cambiada y, efectivamente, al intentar modificar otra vez los datos, los campos aparecen rellenos con los nuevos datos.
Prueba Técnica <ul style="list-style-type: none"> – Nivel profundidad: 2 – Número máximo de páginas: 100 – Hilos por defecto: 15 		
Prueba	Resultado Esperado	Resultado Obtenido
Se deja algún un campo vacío.	El sistema devuelve un error.	Efectivamente, el sistema no permite la acción y muestra el siguiente mensaje: Completa este campo.
Prueba Técnica <ul style="list-style-type: none"> – Nivel profundidad: vacío – Número máximo de páginas: 100 – Hilos por defecto: vacío 		
Prueba	Resultado Esperado	Resultado Obtenido
Cancelar la operación.	La configuración básica del sistema permanece sin cambios.	Efectivamente, al volver a entrar en esta sección los campos siguen con los mismos datos.
Prueba Técnica		
Se cambia de pantalla antes de		

presionar el botón actualizar.		
Prueba	Resultado Esperado	Resultado Obtenido
Se introducen valores numéricos negativos.	El sistema muestra un error.	Esta validación no está contemplada y el sistema permite añadir números negativos.
Prueba Técnica		Acción Realizada
<ul style="list-style-type: none"> – Nivel profundidad: 2 – Número máximo de páginas: 100 – Hilos por defecto: -15 		Se realizan los cambios pertinentes para que el sistema no permita la acción y muestre el siguiente mensaje: Este campo debe ser igual o superior a 0
Prueba	Resultado Esperado	Resultado Obtenido
Se introduce un carácter en un campo numérico.	El sistema no permite la acción.	Efectivamente, el sistema no permite introducir en estos campos nada que no sea un número.
Prueba Técnica		
<ul style="list-style-type: none"> – Nivel profundidad: 2 – Número máximo de páginas: 100 – Hilos por defecto: a 		
Prueba	Resultado Esperado	Resultado Obtenido
Para el nivel de profundidad, el usuario introduce un valor superior a 3.	El sistema muestra un error.	Efectivamente, el sistema no permite la acción y muestra el siguiente mensaje: Este campo debe ser menor o igual a 3
Prueba Técnica		
<ul style="list-style-type: none"> – Nivel profundidad: 4 – Número máximo de páginas: 100 – Hilos por defecto: 15 		

Tabla 137. Desarrollo plan de pruebas para configuración básica

Escenario 4: Añadir métrica		
Prueba	Resultado Esperado	Resultado Obtenido
Se sube un archivo JAR que contiene una métrica correctamente implementada.	El sistema muestra la confirmación de la operación realizada.	Se puede observar como las clases de esta métrica se han subido al servidor, pero el sistema no las puede utilizar, ya que no se ha recargado.
Prueba Técnica		Acción Realizada
Se introduce la métrica correctamente implementada WAB20.jar		Se llama al método que recarga las métricas para poder utilizar la nueva métrica.
Prueba	Resultado Esperado	Resultado Obtenido
Se sube un archivo JAR que no contiene una métrica correctamente implementada.	El sistema muestra un error.	Efectivamente, el sistema no permite la acción y muestra el siguiente mensaje: La métrica no ha sido añadida.
Prueba Técnica		
Se introduce una métrica que no sigue los criterios necesarios.		
Prueba	Resultado Esperado	Resultado Obtenido
Se sube un archivo con un	El sistema no permite	Efectivamente, el sistema no permite

formato diferente al especificado.	la acción.	la acción y muestra el siguiente mensaje: Formato del fichero inválido.
Prueba Técnica		
Se intenta introducir el archivo universidades.txt .		
Prueba	Resultado Esperado	Resultado Obtenido
Cancelar la operación	El sistema permanece sin cambios.	En efecto, no aparece ninguna nueva métrica para escoger.
Prueba Técnica		
Se cambia de pantalla antes de presionar el botón añadir.		
Prueba	Resultado Esperado	Resultado Obtenido
Se intenta subir una métrica que ya ha sido subida.	El sistema sobrescribe la información.	Ocurre un error, ya que la métrica aparece duplicada.
Prueba Técnica		Acción Realizada
Se vuelve a introducir la métrica WAB20.jar		Se introduce la condición necesaria para evitar duplicados y sobrescribir las métricas.
Prueba	Resultado Esperado	Resultado Obtenido
No se introduce ningún archivo, es decir, el campo se deja vacío.	El sistema muestra un error.	Efectivamente, el sistema no permite la acción y muestra el siguiente mensaje: Por favor, selecciona un fichero para subir.
Prueba Técnica		
Fichero: vacío		

Tabla 138. Desarrollo plan de pruebas para añadir una métrica

Escenario 5: Añadir crawler		
Prueba	Resultado Esperado	Resultado Obtenido
Se sube un archivo JAR que contiene un crawler correctamente implementado.	El sistema muestra la confirmación de la operación realizada.	El crawler se añade al Sistema sin problema, se puede utilizar y se muestra el mensaje: El crawler se ha añadido correctamente.
Prueba Técnica		
Se introduce el crawler correctamente implementado Adapter.jar		
Prueba	Resultado Esperado	Resultado Obtenido
Se sube un archivo JAR que no contiene un crawler correctamente implementado.	El sistema muestra un error.	Efectivamente, el sistema no permite la acción y muestra el siguiente mensaje: El crawler no ha sido añadido.
Prueba Técnica		
Se introduce un crawler que no sigue los criterios requeridos.		
Prueba	Resultado Esperado	Resultado Obtenido
Se sube un archivo con un formato diferente al especificado.	El sistema no permite la acción.	Efectivamente, el sistema no permite la acción y muestra el siguiente mensaje: Formato del fichero inválido.
Prueba Técnica		
Se intenta introducir el archivo universidades.txt .		
Prueba	Resultado Esperado	Resultado Obtenido
Cancelar la operación	El sistema permanece sin cambios.	En efecto, no aparece ningún nuevo crawler para escoger.
Prueba Técnica		

Se cambia de pantalla antes de presionar el botón añadir.		
Prueba	Resultado Esperado	Resultado Obtenido
Se intenta subir una crawler que ya ha sido subido.	El sistema sobrescribe la información.	Se puede utilizar el crawler sin problemas y sin duplicados.
Prueba Técnica		
Se vuelve a introducir el crawler Adapter.jar		
Prueba	Resultado Esperado	Resultado Obtenido
No se introduce ningún archivo, es decir, el campo se deja vacío.	El sistema muestra un error.	Efectivamente, el sistema no permite la acción y muestra el siguiente mensaje: Por favor, selecciona un fichero para subir.
Prueba Técnica		
Fichero: vacío		

Tabla 139. Desarrollo plan de pruebas para añadir un crawler

Escenario 6: Añadir un usuario		
Prueba	Resultado Esperado	Resultado Obtenido
Se introducen todos los campos de forma correcta, para añadir un nuevo usuario.	El sistema muestra la confirmación de la operación.	Aparece u nuevo usuario en la base de datos y, por lo tanto, en el sistema. Además, se muestra el siguiente mensaje: El usuario ha sido añadido con éxito.
Prueba Técnica		
<ul style="list-style-type: none"> – Nombre: nuevo – Correo electrónico: nuevo@gmail.com – Contraseña: nuevo – Repite contraseña: nuevo 		
Prueba	Resultado Esperado	Resultado Obtenido
Añadir un nombre de usuario existente en el sistema.	El sistema muestra un error.	El sistema no permite esta acción, ya que el nombre de usuario ya existe. Se muestra el siguiente error: Nombre de usuario ya está en uso.
Prueba Técnica		
<ul style="list-style-type: none"> – Nombre: nuevo – Correo electrónico: otro@gmail.com – Contraseña: nuevo – Repite contraseña: nuevo 		
Prueba	Resultado esperado	Resultado Obtenido
Añadir un correo electrónico existente en el sistema.	El sistema muestra un error.	El sistema no permite esta acción, ya que el correo electrónico ya existe. Se muestra el siguiente error: Correo electrónico ya está en uso.
Prueba Técnica		
<ul style="list-style-type: none"> – Nombre: otro – Correo electrónico: nuevo@gmail.com – Contraseña: nuevo – Repite contraseña: nuevo 		
Prueba	Resultado esperado	Resultado Obtenido
Se introduce un correo	El sistema muestra	El sistema no permite este correo

electrónico que no sigue el siguiente patrón aa@bb.cc	un error.	electrónico, ya que no sigue el patrón requerido y muestra el siguiente error: Correo electrónico inválido.
Prueba Técnica		
Se introduce el siguiente correo electrónico: @gmail.com		
Prueba	Resultado Esperado	Resultado Obtenido
Dejar algún un campo sin rellenar.		
Prueba Técnica		
<ul style="list-style-type: none"> – Nombre: vacío – Correo electrónico: vacío – Contraseña: vacío – Repite contraseña: vacío 	El sistema muestra un error.	El sistema no permite añadir un usuario con todos los campos vacíos y muestra el error Completa los campos.
Prueba	Resultado Esperado	Resultado Obtenido
La contraseña y la repetición de la contraseña no coinciden.		
Prueba Técnica		
<ul style="list-style-type: none"> – Nombre: usuario – Correo electrónico: usuario @gmail.com – Contraseña: nuevo – Repite contraseña: viejo 	El sistema muestra un error.	Efectivamente, el sistema muestra el siguiente mensaje de error: Contraseñas no coinciden y no permite añadir el usuario.

Tabla 140. Desarrollo plan de pruebas para añadir usuarios

Escenario 7: Eliminar un usuario		
Prueba	Resultado Esperado	Resultado Obtenido
Eliminar un usuario de entre la lista de usuarios del sistema.	El sistema muestra la confirmación a la operación.	El sistema muestra el siguiente mensaje: El usuario ha sido eliminado con éxito y, efectivamente, ya no aparece este usuario en la lista.
Prueba Técnica		
Se accede al listado de eliminación de usuarios para eliminar al primer usuario.		
Prueba	Resultado Esperado	Resultado Obtenido
Se accede a la sección de eliminar un usuario y no puede aparecer ni el usuario en sesión, ni el administrador.	El sistema no muestra estos dos usuarios.	En el listado aparece el usuario administrador y el usuario en sesión.
Prueba Técnica		Acción Realizada
Se accede al listado de eliminación de usuarios para eliminar al usuario Carmen.		Se comprueba y modifica la condición que conlleva a este error.
Prueba	Resultado Esperado	Resultado Obtenido
No se escoge ningún usuario a eliminar.	El sistema muestra un error.	Efectivamente, el sistema no permite la acción y muestra el siguiente mensaje: No puede estar vacío.
Prueba Técnica		
Se presiona el botón eliminar sin haber seleccionado ningún		

usuario de la lista.		
Prueba	Resultado Esperado	Resultado Obtenido
Cancelar la operación.	El sistema no cambia.	En efecto, no se ha eliminado ningún usuario del sistema.
Prueba Técnica		
Se cambia de pantalla antes de presionar el botón eliminar.		

Tabla 141. Desarrollo plan de pruebas para eliminar usuario

Escenario 8: Configuración de notificaciones		
Prueba	Resultado Esperado	Resultado Obtenido
El usuario escoge la frecuencia con la que desea obtener avisos. Puede escoger la opción de no recibir avisos.	El sistema actualiza esta información y muestra un mensaje de éxito.	Se muestra el siguiente mensaje: Minutos actualizados. Ahora el usuario deja de recibir notificaciones.
Prueba Técnica		
El usuario está recibiendo notificaciones cada 15 minutos y se modifica el campo minutos del formulario a No deseo correos.		
Prueba	Resultado Esperado	Resultado Obtenido
Cancelar la operación.	El sistema permanece sin cambios.	El usuario sigue con la misma frecuencia de avisos que antes.
Prueba Técnica		
Se cambia de pantalla antes de presionar el botón configurar.		

Tabla 142. Desarrollo plan de pruebas para configurar notificaciones

Escenario 9: Crear un proyecto		
Prueba	Resultado Esperado	Resultado Obtenido
Introducir y seleccionar los datos correctamente.	El sistema crea un nuevo proyecto, lo abre y muestra un mensaje de verificación.	Efectivamente, el proyecto se ha creado con éxito y aparece, automáticamente, como abierto en el sistema. Se muestra el mensaje: El proyecto ha sido añadido con éxito.
Prueba Técnica		
<ul style="list-style-type: none"> – Nombre: Proyecto – Descripción: Soy un Proyecto nuevo – Métricas: WAB, TE, UWEM – Nivel de profundidad: 2 – Número máximo de páginas: 10 – Crawler: Básico – Configuración: - 		
Prueba	Resultado Esperado	Resultado Obtenido
No seleccionar o introducir algún los campos obligatorios.	El sistema no permite la creación del proyecto.	Efectivamente, el sistema no permite crear el proyecto, y obliga al usuario a rellenar todos los campos, mostrando los mensajes: No puede estar vacío.
Prueba Técnica		
<ul style="list-style-type: none"> – Nombre: Proyecto – Descripción: vacío – Métricas: WAB, TE, UWEM – Nivel de profundidad: vacío – Número máximo de 		

páginas: 10 – Crawler: vacío – Configuración: -		
Prueba	Resultado Esperado	Resultado Obtenido
Cancelar la operación	El sistema permanece sin cambios.	El sistema no abre ningún proyecto, ya que no se ha creado ninguno.
Prueba Técnica		
Se cambia de pantalla antes de presionar el botón añadir.		

Tabla 143. Desarrollo plan de pruebas para crear un proyecto

Escenario 10: Abrir un proyecto		
Prueba	Resultado Esperado	Resultado Obtenido
Seleccionar un proyecto de entre la lista de proyectos del sistema.	El sistema muestra la confirmación a la operación y se abre un proyecto.	Se muestra el mensaje de verificación: El proyecto se ha abierto con éxito. El proyecto con nombre Análisis de universidades está abierto en el sistema.
Prueba Técnica		
Se selecciona el proyecto con nombre Análisis de universidades para abrir.		
Prueba	Resultado Esperado	Resultado Obtenido
No se escoge ningún proyecto para ser abierto.	El sistema muestra un error.	Efectivamente, el sistema no permite la acción y muestra el siguiente mensaje: Seleccione una de estas opciones.
Prueba Técnica		
Se presiona el botón abrir sin haber escogido ningún proyecto, previamente.		
Prueba	Resultado Esperado	Resultado Obtenido
Cancelar la operación.	El sistema no cambia.	Efectivamente, el sistema no cambia, ya que no se abre ningún proyecto.
Prueba Técnica		
Se cambia de pantalla antes de presionar el botón abrir.		

Tabla 144. Desarrollo plan de pruebas para abrir un proyecto

Escenario 11: Eliminar un proyecto		
Prueba	Resultado Esperado	Resultado Obtenido
Seleccionar un proyecto de entre la lista de proyectos del sistema.	El sistema muestra la confirmación a la operación y se elimina el proyecto.	Se muestra el mensaje de verificación: El proyecto se ha eliminado con éxito. Se accede a la lista de nuevo y ya no aparece ningún proyecto con ese mismo nombre en la misma fecha, ya que se ha eliminado.
Prueba Técnica		
Se selecciona el proyecto con nombre Análisis de universidades para eliminar.		
Prueba	Resultado Esperado	Resultado Obtenido
No se escoge ningún proyecto para ser	El sistema muestra un error.	Efectivamente, el sistema no permite la acción y muestra el siguiente mensaje:

eliminado.		Seleccione una de estas opciones.
Prueba Técnica		
Se presiona el botón eliminar sin haber escogido ningún proyecto, previamente.		
Prueba	Resultado Esperado	Resultado Obtenido
Cancelar la operación.		
Prueba Técnica		
Se cambia de pantalla antes de presionar el botón eliminar.	El sistema no cambia.	Efectivamente, el sistema no cambia, ya que no se elimina ningún proyecto.

Tabla 145. Desarrollo plan de pruebas para eliminar un proyecto

Escenario 12: Importar un proyecto		
Prueba	Resultado Esperado	Resultado Obtenido
Subir un archivo XML con un proyecto de Atenea.		
Prueba Técnica		
Se sube un fichero XML con el siguiente contenido: <?xml version="1.0" encoding="UTF-8"?> <Project> <Name value="Jobo"/> <Description value="Jobo sin robots"/> <Metrics> <metric name="WAB20"/> <metric name="TOTAL IMAGES"/> <metric name="IVORY"/> </Metrics> <Depth_level value="0"/> <Max_number_pages value="5"/> <Crawler value="Jobo"> <characteristic id="1" value="true"/> </Crawler> </Project> En este proyecto se contemplan todos los posibles atributos de un proyecto. Una única línea.	El sistema crea un nuevo proyecto y lo abre.	Efectivamente, el proyecto se ha creado con éxito y aparece, automáticamente, como abierto en el sistema. Se muestra el mensaje: El proyecto ha sido añadido con éxito.
Prueba	Resultado Esperado	Resultado Obtenido
Subir un fichero con formato diferente a XML.		
Prueba Técnica		
Se sube un fichero TXT con el siguiente contenido: <?xml version="1.0" encoding="UTF-8"?> <Project> <Name value="Jobo"/> <Description value="Jobo sin robots"/> <Metrics> <metric name="WAB20"/> <metric name="TOTAL IMAGES"/> </Metrics>	El sistema no permite esta acción.	Efectivamente, el campo para introducir este tipo de fichero no admite un formato que no sea XML.

<pre><metric name="IVORY"/> </Metrics> <Depth_level value="0"/> <Max_number_pages value="5"/> <Crawler value="Jobo"> <characteristic id="1" value="true"/> </Crawler> </Project></pre>		
Prueba	Resultado Esperado	Resultado Obtenido
Subir un fichero en formato XML, pero con un contenido inadecuado.	El sistema no permite la creación del proyecto y muestra un mensaje de error.	El sistema no abre ni crea ningún proyecto y muestra el siguiente mensaje de error: El proyecto no ha sido importado.
Prueba Técnica		
<pre><?xml version="1.0" encoding="UTF-8"?> <Documento> <Proyecto value="Jobo"/> </Documento></pre>		
Prueba	Resultado Esperado	Resultado Obtenido
Cancelar la operación	El sistema permanece sin cambios.	El sistema no abre ningún proyecto, ya que no se ha importado ninguno.
Prueba Técnica		
Se cambia de pantalla antes de presionar el botón importar.		

Tabla 146. Desarrollo plan de pruebas para importar un proyecto

Escenario 13: Exportar un proyecto		
Prueba	Resultado Esperado	Resultado Obtenido
Seleccionar un proyecto de entre la lista de proyectos del sistema y pulsar el botón de exportar.	El sistema exporta al ordenador del cliente el fichero.	El sistema exporta el proyecto a través de fichero XML con el siguiente formato: <pre><Project> <Name value="Análisis de universidades"/> <Description value="Enlaces universidades"/> <Metrics> <metric name="WAB20"/> <metric name="TOTAL IMAGES"/> <metric name="TOTAL WORDS"/> <metric name="TOTAL FONTS"/> <metric name="WAB"/> </Metrics> <Depth_level value="1"/> <Max_number_pages value="15"/> <Crawler value="Basic"/> </Project></pre> Si el archivo tiene otro formato, la herramienta Atenea no podrá volver a importar este proyecto.
Prueba Técnica		
Se selecciona el proyecto con nombre Análisis de universidades para exportar.		
Prueba	Resultado Esperado	Resultado Obtenido
No se escoge ningún proyecto a exportar.	El sistema muestra un error y no se exporta ningún fichero.	Efectivamente, el sistema no permite la acción y muestra el siguiente mensaje: Seleccione una de estas opciones.
Prueba Técnica		
Se presiona el botón exportar		

sin haber escogido ningún proyecto, previamente.		
Prueba	Resultado Esperado	Resultado Obtenido
Cancelar la operación.	El sistema no exporta nada al ordenador del cliente.	Efectivamente, al no haber realizado ninguna operación tanto el servidor como el cliente permanecen sin cambios.
Prueba Técnica		
Se cambia de pantalla antes de presionar el botón exportar.		

Tabla 147. Desarrollo plan de pruebas para exportar un proyecto

Escenario 14: Cerrar un proyecto		
Prueba	Resultado Esperado	Resultado Obtenido
Seleccionar cerrar un proyecto.	El sistema muestra la confirmación de la operación y se cierra el proyecto.	Efectivamente, el proyecto se cierra, no se puede realizar más análisis con él y el sistema muestra el mensaje: El proyecto ha sido cerrado con éxito.
Prueba Técnica		
El proyecto con nombre análisis de universidades está abierto y se puede analizar enlaces con él. Se decide cerrar dicho proyecto.		

Tabla 148. Desarrollo plan de pruebas para cerrar un proyecto

Escenario 15: Ejecutar una consulta SQL		
Prueba	Resultado Esperado	Resultado Obtenido
Ejecutar una consulta SQL correcta.	El sistema devuelve la información demandada.	El sistema devuelve una tabla con todos los datos de todos los proyectos existentes en el sistema.
Prueba Técnica		
SELECT * FROM project;		
Prueba	Resultado Esperado	Resultado Obtenido
Introducir una consulta SQL errónea.	El sistema devuelve un error.	El sistema no devuelve ningún resultado, ya que la consulta tiene un error de sintaxis. Se muestra un mensaje informando de este error.
Prueba Técnica		
SELECT * projects;		
Prueba	Resultado Esperado	Resultado Obtenido
No introducir ninguna consulta SQL, es decir, no rellenar un campo obligatorio.	El sistema devuelve un error.	El sistema no permite esta acción y muestra el mensaje: Este campo no debe estar vacío.
Prueba Técnica		
Se presiona el botón ejecutar, sin haber introducido ninguna consulta en el campo correspondiente.		
Prueba	Resultado Esperado	Resultado Obtenido
Cancelar la operación.	El sistema permanece sin cambios.	Efectivamente, no se realiza ninguna transacción a la base de datos.
Prueba Técnica		
Se cambia de pantalla antes de presionar el botón ejecutar.		

Prueba	Resultado Esperado	Resultado Obtenido
Intentar hacer un INSERT en una tabla de la base de datos.	El sistema devuelve un error.	Se consulta la base de datos y se ha creado un nuevo proyecto.
Prueba Técnica		Acción Realizada
INSERT INTO project (`id`, `crawler`, `date`, `description`, `level`, `max_page`, `metrics`, `name`) VALUES (1, 'Jobo', '2018-06-10 13:56:50', 'Prueba proyecto', '1', '5', 'TE', 'Proyecto');		Por razones de seguridad esta acción no puede ser permitida en el sistema. Se desarrollan las medidas pertinentes para no permitir este error.

Tabla 149. Desarrollo plan de pruebas para ejecutar una consulta

Escenario 16: Añadir una consulta predeterminada		
Prueba	Resultado Esperado	Resultado Obtenido
Se introducen un nombre y una sentencia SQL correctas.	El sistema muestra la confirmación de la operación.	Se muestra el siguiente mensaje: La consulta ha sido añadida. Además, se accede a la zona de abrir una consulta y aparece en el listado.
Prueba Técnica – Name: All projects – Sentencia: SELECT * FROM project;		
Añadir un nombre de consulta existente en el sistema.	El sistema muestra un error.	No se puede añadir esta consulta predeterminada y se muestra el siguiente mensaje: El nombre de la consulta ya está en uso.
Prueba Técnica – Name: All projects – Sentencia: SELECT * FROM Project ORDER BY name;		
Añadir una sentencia SQL existente en el sistema.	El sistema muestra un error.	No se puede añadir esta consulta predeterminada y se muestra el siguiente mensaje: La consulta ya existe.
Prueba Técnica – Name: All projects II – Sentencia: SELECT * FROM Project;		
Se introduce una sentencia SQL con una sintaxis incorrecta.	El sistema muestra un error.	No se añade ninguna consulta al sistema, ya que tiene un error de sintaxis. Se muestra un mensaje informando de este error.
Prueba Técnica – Name: All projects – Sentencia: SELECT * project;		
Dejar algún un campo sin rellenar.	El sistema muestra un error.	Se había olvidado contemplar esta situación y el sistema permite añadir una consulta con nombre y sentencia vacías.
Prueba Técnica		Acción Realizada
– Name: vacío		Se modifica el sistema

– Sentencia: SELECT * project;		para obligar al usuario a introducir estos campos.
Prueba	Resultado Esperado	Resultado Obtenido
Cancelar la operación.	El sistema permanece sin cambios.	El resultado esperado coincide con el resultado obtenido, ya que no se presenta ningún cambio nuevo.
Prueba Técnica		
Se cambia de pantalla antes de presionar el botón añadir.		

Tabla 150. Desarrollo plan de pruebas para añadir una consulta

Escenario 17: Abrir una consulta		
Prueba	Resultado Esperado	Resultado Obtenido
Seleccionar una consulta de entre la lista de consultas existentes en el sistema.	El sistema ejecuta la consulta y muestra los resultados obtenidos.	El sistema devuelve una tabla con todos los datos de todos los proyectos existentes en el sistema.
Prueba Técnica		
Se selecciona la consulta con nombre All projects para abrirla.		
Prueba	Resultado Esperado	Resultado Obtenido
No se escoge ninguna consulta de la lista.	El sistema muestra un error.	El resultado esperado coincide con el resultado obtenido, ya que el sistema muestra el siguiente mensaje de error: Selecciona una de estas opciones.
Prueba Técnica		
Se presiona el botón abrir sin haber escogido ninguna consulta, previamente.		
Prueba	Resultado Esperado	Resultado Obtenido
Cancelar la operación.	El sistema no cambia.	Efectivamente, no se abre ni se ejecuta ninguna consulta.
Prueba Técnica		
Se cambia de pantalla antes de realizar las acciones que permiten abrir una consulta.		

Tabla 151. Desarrollo plan de pruebas para abrir una consulta

Escenario 18: Eliminar una consulta		
Prueba	Resultado Esperado	Resultado Obtenido
Seleccionar una consulta de entre la lista de consultas para eliminar.	El sistema muestra un mensaje de verificación y elimina la consulta.	El resultado esperado coincide con el resultado obtenido, ya que se ha eliminado la consulta correctamente y el sistema muestra un error.
Prueba Técnica		
Se selecciona la consulta con nombre All projects para eliminarla.		
Prueba	Resultado Esperado	Resultado Obtenido
No se escoge ninguna consulta de la lista.	El sistema muestra un error.	El resultado esperado coincide con el resultado obtenido, ya que el sistema muestra el siguiente mensaje de error: Selecciona una de estas opciones.
Prueba Técnica		
Se presiona el botón		

eliminar sin haber escogido ninguna consulta, previamente.		
Prueba	Resultado Esperado	Resultado Obtenido
Cancelar la operación.	El sistema no cambia.	Efectivamente, el sistema cuenta con el mismo número de consultas predeterminadas en el sistema.
Prueba Técnica		
Se cambia de pantalla antes de realizar las acciones que permiten eliminar una consulta.		

Tabla 152. Desarrollo plan de pruebas para eliminar consulta

Escenario 19: Ver el estado de los análisis en curso		
Prueba	Resultado Esperado	Resultado Obtenido
Seleccionar la pestaña de análisis en curso.	El sistema muestra todos los análisis en curso.	Efectivamente, el sistema muestra en formato de tabla todos los análisis en ejecución. Entre otras cosas, aparece el porcentaje de progreso.
Prueba Técnica		
El usuario comienza un análisis y se dirige a la sección de análisis en curso para ver el estado de estos.		
Prueba	Resultado Esperado	Resultado Obtenido
Seleccionar la pestaña de análisis en curso cuando no existe ningún análisis en ejecución.	El sistema muestra un error.	El resultado obtenido es el esperado, ya que se muestra un mensaje indicando que no hay ningún análisis en curso.
Prueba Técnica		
Actualmente, no existe ningún análisis en curso, es decir, ni el usuario en sesión ni otro está ejecutando ningún análisis.		
Prueba	Resultado Esperado	Resultado Obtenido
Seleccionar la traza de un análisis en curso.	El sistema muestra la traza de dicho análisis.	El resultado obtenido es el esperado, ya que se muestra, en una sección, los enlaces terminados en color rojo y los enlaces en curso en color verde.
Prueba Técnica		
Hay un análisis en curso y se presiona el botón de ver traza, con el fin de conocer qué enlaces han sido analizados y cuáles no y con qué métricas.		
Prueba	Resultado Esperado	Resultado Obtenido
Actualizar los análisis para ver el nuevo porcentaje completado.	El sistema actualiza la tabla con el nuevo porcentaje.	Efectivamente, al actualizar la tabla, el porcentaje de análisis se ha modificado.
Prueba Técnica		Nota
Hay un análisis en curso y se desea tener una idea del porcentaje de enlaces analizados y, por lo tanto, del resto. Para ello, se presiona el botón actualiza tabla de esta sección.		Si se actualizan los análisis en curso muy rápido, es posible que no se vea ningún cambio en dicho porcentaje.

Tabla 153. Desarrollo plan de pruebas para análisis en curso

Escenario 20: Detener un análisis en curso		
Prueba	Resultado Esperado	Resultado Obtenido
Seleccionar un análisis de la lista de los análisis en curso para detenerlo.	El sistema pide confirmación para detenerlo. Si la respuesta es afirmativa, lo detiene.	Efectivamente, el sistema pide confirmar dicha acción y, en caso afirmativo, se detiene el análisis seleccionado. Además, el sistema almacena el resultado de los enlaces ya analizados.
Prueba Técnica		
Hay varios análisis en curso y se selecciona uno para detenerlo.		
Prueba	Resultado Esperado	Resultado Obtenido
No se escoge ningún análisis de la lista.	El sistema muestra un error.	El sistema falla, ya que aparece un mensaje informando que se debe escoger algún enlace, pero el listado de análisis en curso desaparece.
Prueba Técnica		Acción Realizada
Se desea detener el análisis con nombre análisis de universidades, pero se presiona el botón detener antes de seleccionar dicho análisis.		El error se encontraba en un controlador y se ha modificado, con el fin de arreglar esta deficiencia.
Prueba	Resultado Esperado	Resultado Obtenido
Cancelar la operación.	El sistema no cambia.	El resultado obtenido coincide con el esperado, ya que existe el mismo número de análisis en curso, que antes de cerrar el navegador.
Prueba Técnica		
Se selecciona un análisis de la lista para eliminar, pero antes de presionar el botón detener, se cierra el navegador.		

Tabla 154. Desarrollo plan de pruebas para detener análisis

Escenario 21: Analizar una URL		
Prueba	Resultado Esperado	Resultado Obtenido
Introducir una URL correcta para analizar.	El sistema analiza la URL y devuelve el resultado.	El resultado obtenido es el esperado, ya que tras mostrar el mensaje de "cargando" y analizar la web, el sistema permite ver los resultados de dicho análisis. En este caso, el resultado ha sido: 0.0686471
Prueba Técnica		
Se abre un proyecto con las siguientes características: <ul style="list-style-type: none"> – Nivel de profundidad: 1 – Métrica: UWEM – Crawler: Basic – Número máximo de páginas: 10 Y se analiza el siguiente enlace: https://www.uniovi.es		
Prueba	Resultado Esperado	Resultado Obtenido
Introducir una URL con un formato erróneo.	El sistema muestra un error.	Efectivamente, el sistema no permite esta acción y muestra el siguiente mensaje: Por favor, introduzca un URL válida.
Prueba Técnica		
Se abre un proyecto con las siguientes características: <ul style="list-style-type: none"> – Nivel de profundidad: 1 – Métrica: UWEM – Crawler: Basic 		

<ul style="list-style-type: none"> – Número máximo de páginas: 10 Y se analiza el siguiente enlace: aaabbbccc		
Prueba	Resultado Esperado	Resultado Obtenido
Dejar el campo de la URL vacío.		
Prueba Técnica		
Se abre un proyecto con las siguientes características: <ul style="list-style-type: none"> – Nivel de profundidad: 1 – Métrica: UWEM – Crawler: Basic – Número máximo de páginas: 10 Y se analiza el siguiente enlace: vacío	El sistema no permite esta acción.	Efectivamente, el sistema no permite esta acción y muestra el siguiente mensaje: Por favor, introduzca un URL válida.
Prueba	Resultado Esperado	Resultado Obtenido
Introducir números en la zona reservada para la URL		
Prueba Técnica		
Se abre un proyecto con las siguientes características: <ul style="list-style-type: none"> – Nivel de profundidad: 1 – Métrica: UWEM – Crawler: Basic – Número máximo de páginas: 10 Y se analiza el siguiente enlace: 1234	El sistema muestra un error.	Efectivamente, el sistema no permite esta acción y muestra el siguiente mensaje: Por favor, introduzca un URL válida.
Prueba	Resultado Esperado	Resultado Obtenido
Cerrar la aplicación sin haber terminado el análisis.		
Prueba Técnica		
Se abre un proyecto con las siguientes características: <ul style="list-style-type: none"> – Nivel de profundidad: 1 – Métrica: UWEM – Crawler: Basic – Número máximo de páginas: 10 Y se analiza el siguiente enlace: https://www.uniovi.es Antes de que el análisis finalice, se cierra la aplicación.	El análisis sigue ejecutándose en segundo plano. Cuando el análisis termina, el usuario puede ver el resultado.	Efectivamente, el sistema continúa con el análisis y, al terminar, se almacena el resultado en la base de datos, para poder estudiarlo en cualquier momento.
Prueba	Resultado Esperado	Resultado Obtenido
Cerrar la ventana una vez terminado el análisis		
Prueba Técnica		
Se abre un proyecto con las siguientes características: <ul style="list-style-type: none"> – Nivel de profundidad: 1 – Métrica: UWEM – Crawler: Basic – Número máximo de páginas: 10 Y se analiza el siguiente enlace: https://www.uniovi.es Al terminar el análisis, se cierra la aplicación.	El sistema permanece sin cambios. El resultado del análisis ya ha sido almacenado.	Efectivamente, el sistema almacena el resultado en la base de datos, para poder estudiarlo en cualquier momento.

Tabla 155. Desarrollo plan de pruebas para analizar una URL

Escenario 22: Analizar una lista de URLs		
Prueba	Resultado Esperado	Resultado Obtenido
Introducir un fichero TXT con un conjunto de URLs separadas por un retorno de carro.	El sistema comienza el análisis.	El resultado obtenido es el esperado, ya que aparece un nuevo análisis en la sección de análisis en curso. En la traza aparecen los dos enlaces del fichero.
Prueba Técnica Se abre un proyecto con las siguientes características: <ul style="list-style-type: none"> – Nivel de profundidad: 1 – Métrica: UWEM – Crawler: Basic – Número máximo de páginas: 10 Se analiza un conjunto de URLs desde un fichero urls.txt con los siguientes enlaces: <ul style="list-style-type: none"> – http://www.uniovi.es/ – https://elpais.com/ 		
Prueba	Resultado Esperado	Resultado Obtenido
Introducir una lista de URLs con un formato incorrecto.	El sistema muestra un error.	El análisis comienza y el sistema muestra el siguiente mensaje: Ahora puede seguir el estado de las evaluaciones en la pestaña de análisis actual. Al terminar, se pueden visualizar los resultados y para la URL mal formada, el resultado es -1.
Prueba Técnica Se abre un proyecto con las siguientes características: <ul style="list-style-type: none"> – Nivel de profundidad: 1 – Métrica: UWEM – Crawler: Basic – Número máximo de páginas: 10 Se analiza un conjunto de URLs desde un fichero urls.txt con los siguientes enlaces: <ul style="list-style-type: none"> – Aabc2313 – https://elpais.com/ 		
Prueba	Resultado Esperado	Resultado Obtenido
Introducir un fichero vacío.	El sistema muestra un error.	El sistema devuelve un error, ya que se debe introducir en el fichero al menos una URL.
Prueba Técnica Se abre un proyecto con las siguientes características: <ul style="list-style-type: none"> – Nivel de profundidad: 1 – Métrica: UWEM – Crawler: Basic – Número máximo de páginas: 10 Se analiza un conjunto de URLs desde un fichero urls.txt con los siguientes enlaces: vacío		
Prueba	Resultado Esperado	Resultado Obtenido
Introducir un fichero con un formato diferente a TXT.	El sistema no permite esta acción.	El sistema no permite esta acción, ya que el mecanismo donde se introducen los ficheros solamente admite el formato .txt
Prueba Técnica Se abre un proyecto con las siguientes características:		

<ul style="list-style-type: none"> – Nivel de profundidad: 1 – Métrica: UWEM – Crawler: Basic – Número máximo de páginas: 10 <p>Se analiza un conjunto de URLs desde un fichero urls.pdf con los siguientes enlaces:</p> <ul style="list-style-type: none"> – http://www.uniovi.es/ – https://elpais.com/ 		
Prueba	Resultado Esperado	Resultado Obtenido
Si el usuario desea notificaciones y el análisis comienza.		
Prueba Técnica		
Se abre un proyecto con las siguientes características: <ul style="list-style-type: none"> – Nivel de profundidad: 1 – Métrica: UWEM – Crawler: Basic – Número máximo de páginas: 10 <p>Se analiza un conjunto de URLs desde un fichero urls.txt con los siguientes enlaces:</p> <ul style="list-style-type: none"> – http://www.uniovi.es/ – https://elpais.com/ 	El sistema enviará varios correos electrónicos con la información asociada a dicho análisis.	Efectivamente, se han recibido notificación al inicio y al final del análisis, así como durante la ejecución del mismo.
Prueba	Resultado Esperado	Resultado Obtenido
Cerrar la aplicación sin haber terminado el análisis.		
Prueba Técnica		
Se abre un proyecto con las siguientes características: <ul style="list-style-type: none"> – Nivel de profundidad: 1 – Métrica: UWEM – Crawler: Basic – Número máximo de páginas: 10 <p>Se analiza un conjunto de URLs desde un fichero urls.txt con los siguientes enlaces:</p> <ul style="list-style-type: none"> – http://www.uniovi.es/ – https://elpais.com/ <p>Antes de terminar el análisis, se cierra la aplicación.</p>	El sistema continúa haciendo el análisis sin necesidad de atención por parte del usuario.	Efectivamente, el sistema continúa con el análisis y, al terminar, se almacenan los resultados en la base de datos, para poder estudiarlos en cualquier momento.
Prueba	Resultado Esperado	Resultado Obtenido
Cerrar la aplicación una vez terminado el análisis.		
Prueba Técnica		
Se abre un proyecto con las siguientes características: <ul style="list-style-type: none"> – Nivel de profundidad: 1 – Métrica: UWEM – Crawler: Basic – Número máximo de páginas: 10 <p>Se analiza un conjunto de URLs desde un</p>	El sistema permanece sin cambios. Los resultados del análisis ya han sido almacenados.	Efectivamente, el sistema almacena los resultados en la base de datos, para poder estudiarlos en cualquier momento.

fichero urls.txt con los siguientes enlaces: <ul style="list-style-type: none"> – http://www.uniovi.es/ – https://elpais.com/ Al terminar el análisis, se cierra la aplicación.		
---	--	--

Tabla 156. Desarrollo plan de pruebas para analizar un conjunto de URLs

Escenario 23: Exportar un análisis asociado un conjunto de URLs		
Prueba	Resultado Esperado	Resultado Obtenido
Seleccionar un análisis de entre la lista de los análisis ejecutados del sistema.		
Prueba Técnica		
Se abre un proyecto con las siguientes características: <ul style="list-style-type: none"> – Nivel de profundidad: 1 – Métrica: UWEM – Crawler: Basic – Número máximo de páginas: 10 Se analiza un conjunto de URLs desde un fichero urls.txt con los siguientes enlaces: <ul style="list-style-type: none"> – http://www.uniovi.es/ – https://elpais.com/ 	El sistema mostrará los resultados asociados a este análisis y permitirá su exportación.	Efectivamente, al finalizar este análisis, en la sección de exportar análisis desde fichero, aparece la opción de seleccionar el fichero url.txt y se puede observar y exportar los resultados obtenidos. En este caso, los resultados son 0.0686471 y 0.151188, respectivamente.
Prueba	Resultado Esperado	Resultado Obtenido
No se escoge ningún análisis para ver y exportar sus resultados.		
Prueba Técnica		
Se accede a la sección de exportar análisis desde fichero y aparece un fichero urls.txt, antes de seleccionar dicho análisis se presiona el botón de ejecutar.	El sistema muestra un error.	El resultado obtenido es el esperado, ya que el sistema no permite esta acción y obliga al usuario, mediante un mensaje, a seleccionar al menos un análisis desde fichero.
Prueba	Resultado Esperado	Resultado Obtenido
Cancelar la operación.		
Prueba Técnica		
Se accede a la sección de exportar análisis desde fichero y aparece un fichero urls.txt, antes de seleccionar dicho análisis, el navegador se cierra inesperadamente.	El sistema no exporta nada al ordenador del cliente.	Efectivamente, no se muestra ningún resultado y, por lo tanto, no se exporta nada al navegador del cliente.

Tabla 157. Desarrollo plan de pruebas para exportar análisis

Escenario 24: Consultar y exportar resultados		
Prueba	Resultado Esperado	Resultado Obtenido
Seleccionar la pestaña de resultados para ver los resultados asociados a un proyecto.		
Prueba Técnica		
Se abre un proyecto con las siguientes características: <ul style="list-style-type: none"> – Nivel de profundidad: 1 – Métrica: UWEM – Crawler: Basic 	El sistema muestra los resultados.	El resultado obtenido coincide con el resultado esperado, ya que al acceder a la sección de resultados, el sistema muestra una tabla con el resultado final. En este caso, el resultado es:

<ul style="list-style-type: none"> Número máximo de páginas: 10 Se desea conocer el resultado de los análisis del enlace de la Universidad de Oviedo.		
Prueba	Resultado Esperado	Resultado Obtenido
Se intenta acceder a los resultados sin tener un proyecto abierto.		Efectivamente, esta acción no es posible, ya que esta opción del menú solamente se habilita si hay un proyecto abierto.
Prueba Técnica	El sistema no permite esta acción, ya que no tiene la pestaña activa.	
Se accede a la zona privada de la aplicación y sin abrir ningún proyecto existente, se intenta acceder a la zona de resultados.		
Prueba	Resultado Esperado	Resultado Obtenido
El sistema permite exportar los resultados y se selecciona el formato.		Se ha detectado un error, puesto que, al abrir el fichero XLS, se ha observado que no muestra los resultados al completo.
Prueba Técnica	El sistema exporta el fichero con los resultados en el formato seleccionado al ordenador del cliente.	Acción Realizada
Se abre un proyecto con las siguientes características: <ul style="list-style-type: none"> Nivel de profundidad: 1 Métrica: UWEM Crawler: Basic Número máximo de páginas: 10 Se desea conocer el resultado de los análisis del enlace de la Universidad de Oviedo. Y, además, se exporta el análisis a formato XLS.		Se ha modificado el método encargado de este tipo de exportación para corregir este error.

Tabla 158. Desarrollo plan de pruebas para consultar resultados

Escenario 25: Consultar y exportar errores		
Prueba	Resultado Esperado	Resultado Obtenido
El usuario selecciona una métrica y uno o más sitios webs de forma correcta.		
Prueba Técnica	El sistema dependiendo de la métrica muestra o exporta los errores correctamente.	Como esta métrica, cuenta con una implementación propia para exportar los errores, el sistema exportará en el ordenador del cliente un fichero PDF con todos los errores.
Se abre un proyecto con las siguientes características: <ul style="list-style-type: none"> Nivel de profundidad: 1 Métrica: UWEM Crawler: Basic Número máximo de páginas: 10 y se selecciona la métrica UWEM y el enlace de la universidad de Oviedo.		
Prueba	Resultado Esperado	Resultado Obtenido
No se selecciona la métrica o no se seleccionan los sitios o ambos a la vez.		Efectivamente, el resultado obtenido es el esperado, ya que el sistema no permite la acción y obliga al a usuario a escoger una
Prueba Técnica	El sistema no permite esta acción.	
Se abre un proyecto con las siguientes características: <ul style="list-style-type: none"> Nivel de profundidad: 1 		

<ul style="list-style-type: none"> – Métrica: UWEM – Crawler: Basic – Número máximo de páginas: 10 y se desea exportar sus errores. No se selecciona ni la métrica y ningún enlace para conocer sus errores.		métrica y al menos un enlace.
Prueba	Resultado Esperado	Resultado Obtenido
No existen errores para una métrica o un determinado sitio.	El sistema muestra o exporta un listado vacío.	Efectivamente, como el análisis no ha presentado ningún error, en este caso, la métrica cuenta con una implementación propia para los errores y exporta un fichero PDF con la lista de errores vacía.
Prueba Técnica		
Se abre un proyecto con las siguientes características: <ul style="list-style-type: none"> – Nivel de profundidad: 1 – Métrica: UWEM – Crawler: Basic – Número máximo de páginas: 10 y se selecciona la métrica UWEM y el enlace de la universidad de León.		
Prueba	Resultado Esperado	Resultado Obtenido
Cancelar operación	El sistema permanece sin cambios.	Efectivamente, el sistema no exporta ni muestra ningún error.
Prueba Técnica		
Se abre un proyecto con las siguientes características: <ul style="list-style-type: none"> – Nivel de profundidad: 1 – Métrica: UWEM – Crawler: Basic – Número máximo de páginas: 10 y se desea conocer si durante alguno de sus análisis se han producido errores, pero el navegador se cierra inesperadamente.		

Tabla 159. Desarrollo plan de pruebas consultar errores

Escenario 26: Cerrar sesión		
Prueba	Resultado Esperado	Resultado Obtenido
Seleccionar cerrar sesión.	El sistema cierra la sesión y dirige al usuario a la página inicial de la aplicación.	Al cerrar la sesión, efectivamente, el sistema muestra la página de inicio de la zona pública y ya no hay un usuario en el objeto Session.
Prueba Técnica		
Tras realizar varias funciones, se decide cerrar la sesión del sistema.		

Tabla 160. Desarrollo plan de pruebas cerrar sesión

8.2 Pruebas de Usabilidad y Accesibilidad

A continuación, se desarrollará el informe de auditoría especificado en el apartado de diseño del presente documento.

8.2.1 Identificación del sitio web

DATOS DE LA WEB	
INSTITUCIÓN	Atenea
URL	http://156.35.94.110/Atenea/
EVALUADOR	Carmen Peñalver San Cristóbal
FECHA	11 de junio de 2018
CALIFICACIÓN	7,32

Tabla 161. Datos para el informe de auditoría

8.2.2 Usabilidad

A continuación, se estudiarán aspectos asociados a la usabilidad que el sitio web de Atenea debe presentar.

8.2.2.1 Comprensibilidad global del sitio

En primer lugar, se realiza un análisis de aspectos general del sitio web, así como su organización y la facilidad con que se permite a un usuario identificar los recursos que busca.

8.2.2.1.1 Esquema de organización global

Actualmente, la página web no cuenta con ninguno de los tres elementos que en este apartado se analizan, es decir, no se puede observar ni un mapa web del sitio, ni un índice global como tal, ni tampoco un listado de todos los contenidos. Si bien hay que destacar que, tanto en la zona pública como la privada de la aplicación, se puede apreciar uno o varios menús que indican y proporcionan de forma organizada todos los apartados con los que cuenta Atenea.

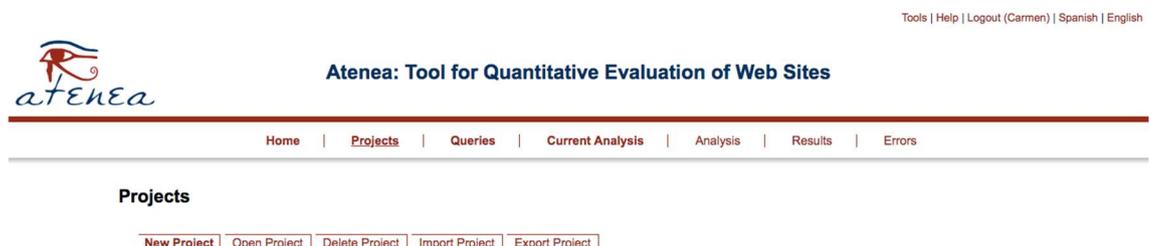


Imagen 86. Menú privado de Atenea

©Atenea: Tool for Quantitative Evaluation of Web Sites
 Contact | Accesibility | About

Imagen 87. Pie de página Atenea

8.2.2.1.2 Calidad en el sistema de etiquetado



Imagen 88. Calidad de etiquetado

A lo largo de toda la página web se observan muchos enlaces, todos ellos, presentan un correcto etiquetado y dan una idea preliminar del sitio a dónde se dirigen.

Únicamente, el enlace de inicio “Atenea” no presenta un correcto etiquetado, ya que no te da una idea clara del contenido de la página a dónde se dirige, pero al ser un enlace común en todas las webs se puede intuir.

8.2.2.1.3 Visita guiada

En este momento no se observa ningún tutorial para utilizar la web adecuadamente, y con el fin de respetar este aspecto, se debería mejorar la web para ayudar a los usuarios a utilizarla de forma adecuada y facilitar su navegación.

8.2.2.1.4 Página principal

Atenea cuenta con una zona tanto pública como privada y las dos cuentan con una página principal. Desde cualquiera de estas dos páginas principales, se refleja claramente el contexto global de la web, es decir, con solo echar un vistazo se puede entender que se trata de un sitio dedicado al análisis de la accesibilidad y usabilidad web.

Además, la web cuenta con un menú que permite una correcta navegación por la página y en cada sección existen enlaces para poder acceder a cualquier contenido relacionado con la misma.

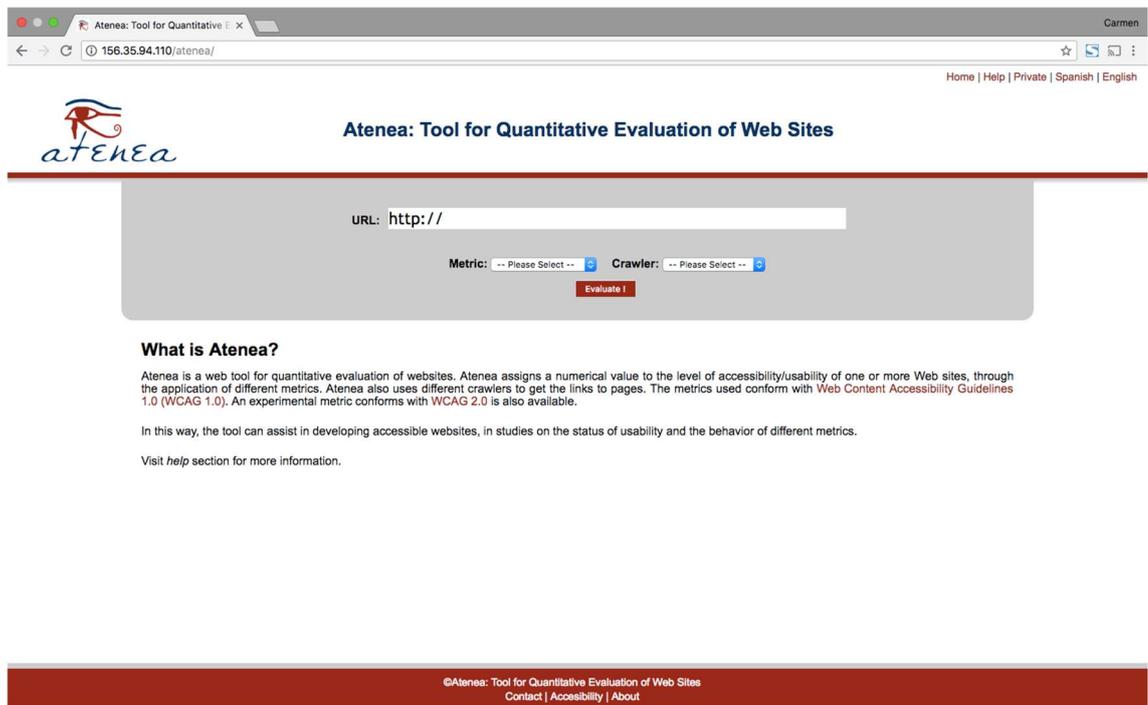


Imagen 89. Página principal pública

La página de Atenea, en general, no cuenta con muchas imágenes. De todas formas, es necesario comprobar que, si se desactiva la opción del navegador “no permitir imágenes”, la página web se sigue visualizando correctamente y mostrando su sentido original.

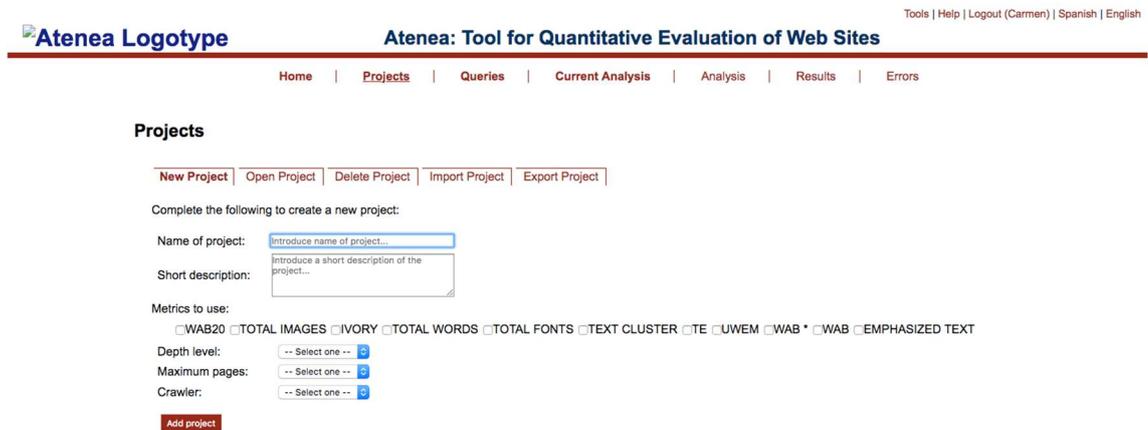


Imagen 90. Atenea sin imágenes

8.2.2.1.5 Consistencia en la navegación

Para el funcionamiento global de Atenea, es necesario utilizar dos aplicaciones complementarias. Ambas aplicaciones mantienen el mismo aspecto global, es decir, en conjunto proporcionan una navegación constante. Este aspecto ayuda considerablemente a la correcta y cómoda utilización de Atenea.

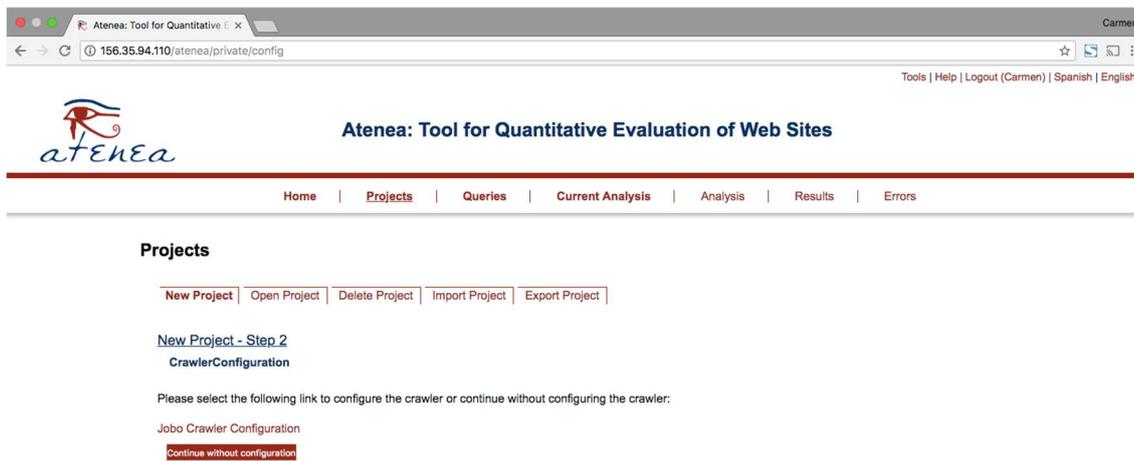


Imagen 91. Navegación consistente

Desde Atenea también se puede acceder a distintas webs, que ayudan a entender se funcionamiento global, pero no todas siguen una consistencia homogénea, lo que hace que la navegación sea incomoda y difícil de utilizar.

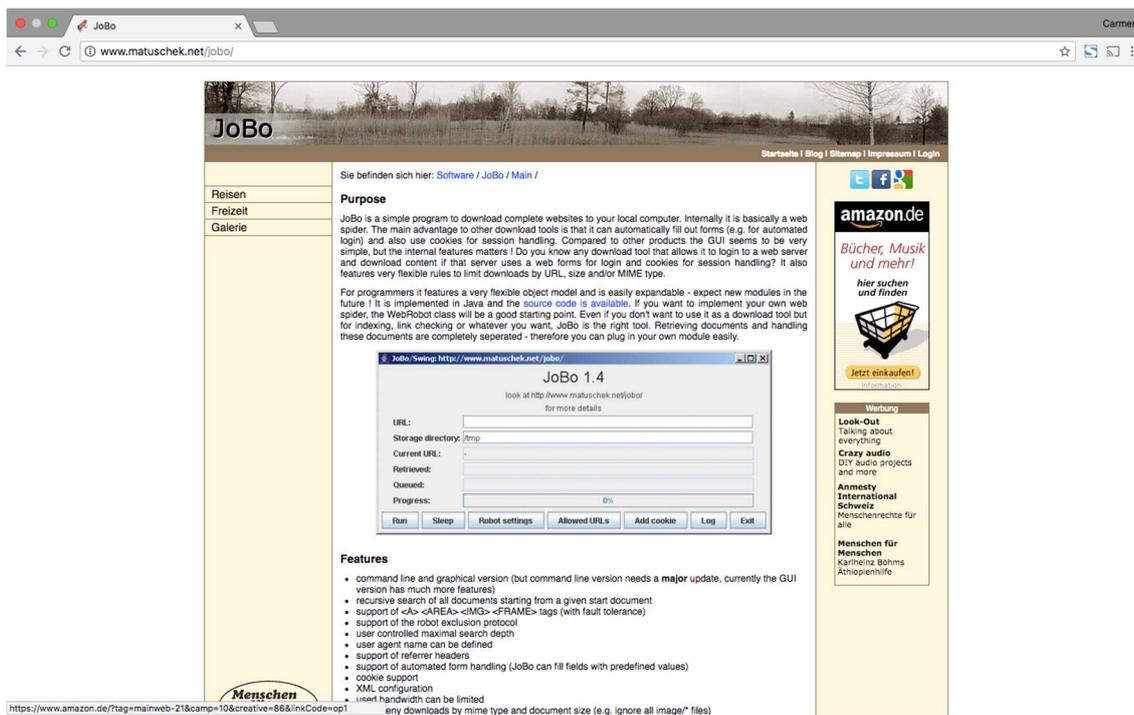


Imagen 92. Especificación JoBo

8.2.2.2 Mecanismo de ayuda y retroalimentación en línea

Actualmente, el sitio web bajo análisis cuenta, tanto en la zona pública como en la zona privada, con un mecanismo de ayuda, que permite facilitar su manejo. Por lo contrario, no incluye ningún indicador que muestra sus posibles actualizaciones.

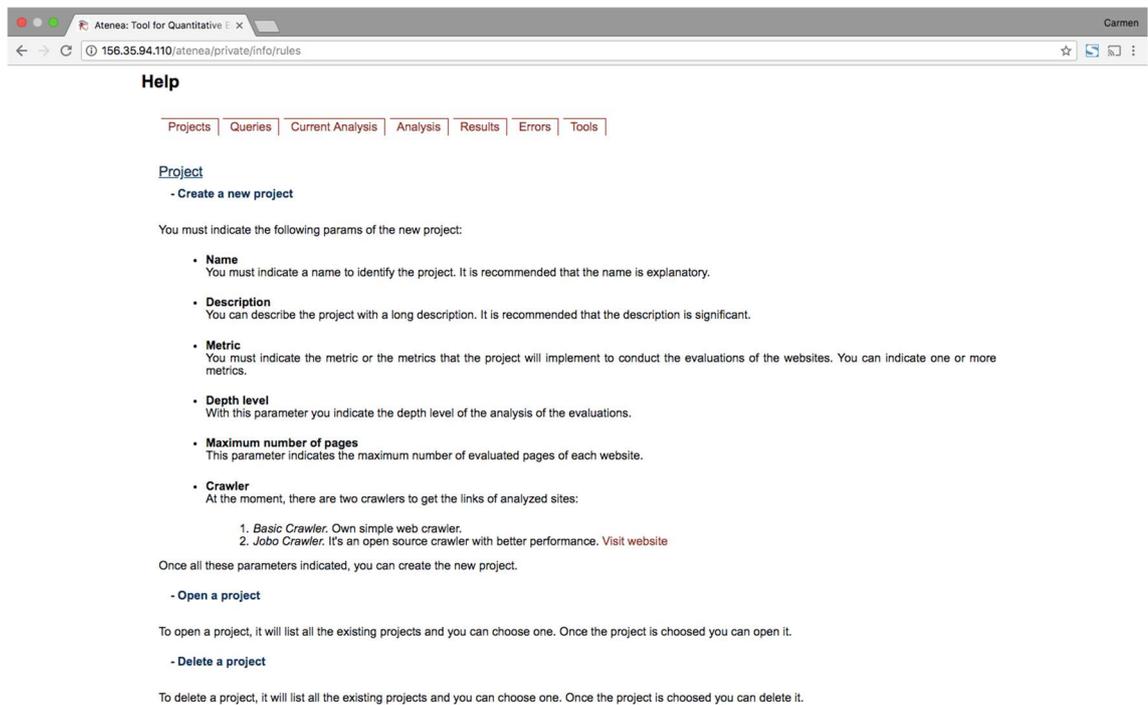


Imagen 93. Ayuda en la zona privada

Sería necesario incluir un apartado con las preguntas más frecuentes de los usuarios. Como ya se ha comentado, Atenea se ayuda de otros enlaces de interés, para entender su completo funcionamiento.

A pesar de que no cuenta con enlaces a redes sociales, Atenea cuenta con un apartado de contacto, que cuenta con la información necesaria para contactar con sus administradores.

Contact

For all site related questions, please feel free to contact us:

ANA BELEN MARTINEZ PRIETO

Address: Calvo Sotelo S/N, Oviedo - Spain. Despacho 62

Telephone: +34 985 104 337

E-mail: **belenmp [at] uniovi.es**

AQUILINO ADOLFO JUAN FUENTE

Address: Calvo Sotelo S/N, Oviedo - Spain. Despacho 197

Telephone: +34 985 103 172

E-mail: **aajuan [at] uniovi.es**

Imagen 94. Información de contacto

8.2.2.3 Aspectos de interfaces y estéticos

El sitio web cuenta con un aspecto adecuado y homogéneo, los colores que presentan son estables y adecuados para las preferencias estéticas de hoy en día.

En cuanto a la cohesividad de los objetos principales, cabe destacar que siempre se muestran en la misma zona de la pantalla y siempre son coherentes e iguales. Esto es, la página web presenta una estabilidad y uniformidad en cuanto a estética se refiere.

Como se ha comentado anteriormente, desde la página de Atenea se puede acceder a otras páginas webs relacionados con su contenido, pero la estética cambia ligeramente.

8.2.2.4 Adaptabilidad a dispositivos móviles y a idiomas

La página web, con el fin de llegar a un mayor número de usuarios, se ha internacionalizado a dos idiomas distintos: español e inglés.

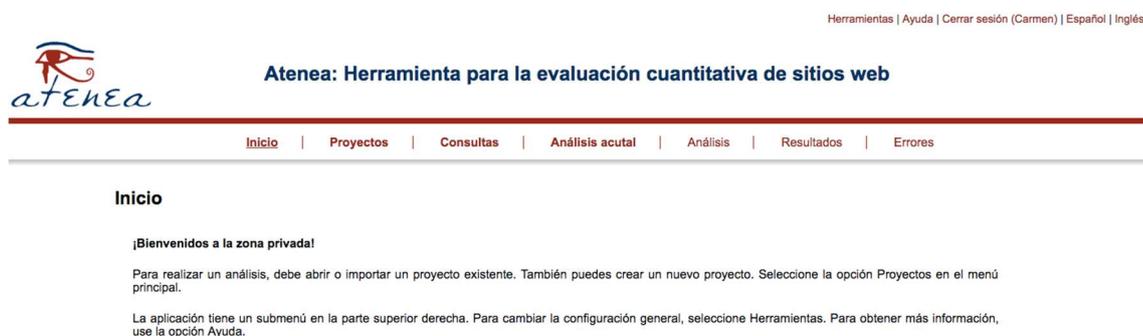


Imagen 95. Atenea en español

Desde el dispositivo móvil del evaluador, en este caso un iPhone, se ha podido verificar que la descarga de contenidos se ha realizado correctamente. Por ejemplo, se ha probado a descargar el fichero XML con la configuración de un proyecto.

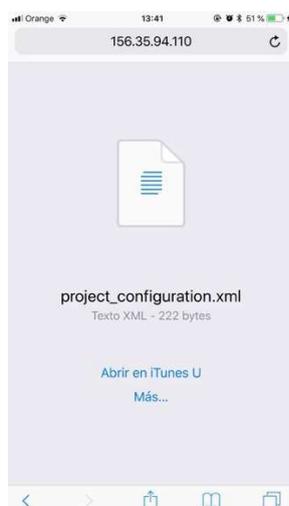


Imagen 96. Descarga de contenidos

8.2.2.4.1 Test de adaptabilidad

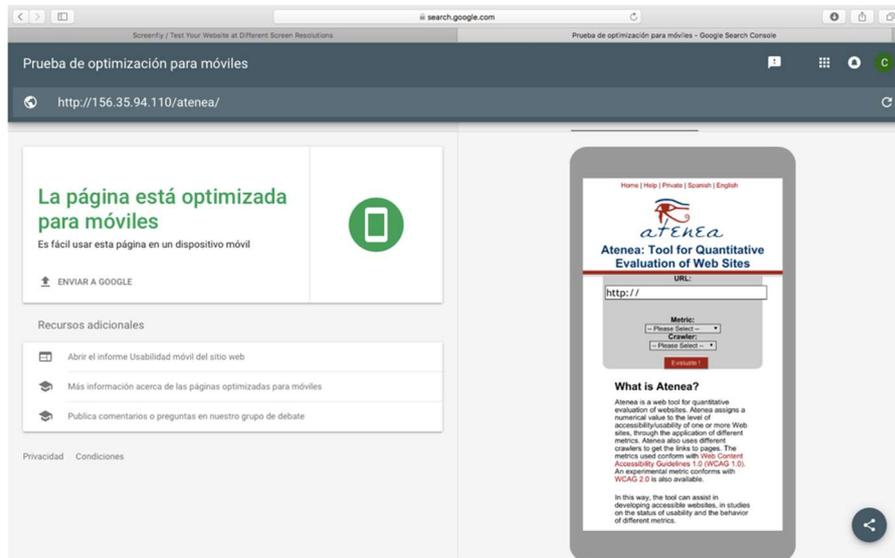


Imagen 97. Test adaptabilidad I

Para realizar un test correcto, no nos podemos fiar únicamente de una herramienta, así que se pasa otro test con otra una herramienta diferente.

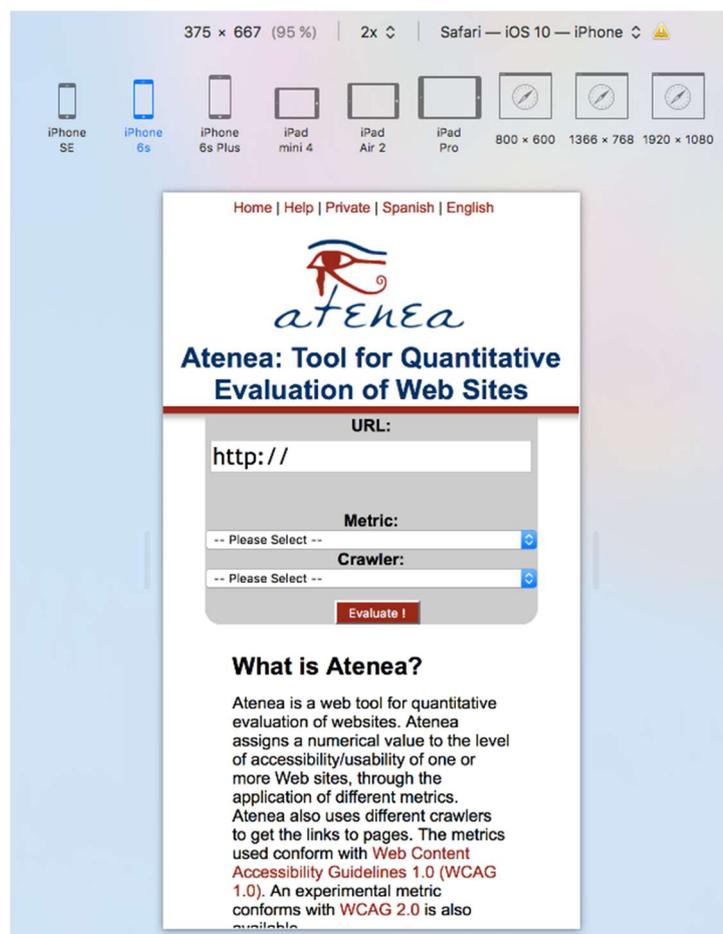


Imagen 98. Test de adaptabilidad II

8.2.2.5 Usabilidad de los textos

En todas las páginas del sitio web, los textos son en general breves, ya que se limitan a dar una información clara y concisa.

Por lo general, el texto se adecua al principio de escaneabilidad, ya que toda la información común resalta por mantenerse agrupada y, además, se usan distintos colores y tamaños de letra en los títulos con el fin de atraer la atención de los usuarios.

8.2.2.6 Clasificación de la información y participación en redes sociales

La información está correctamente clasificada, ya que la página cuenta con distintas secciones claramente diferenciadas y agrupadas por su contenido, como se puede observar en las siguientes imágenes que se adjuntan.

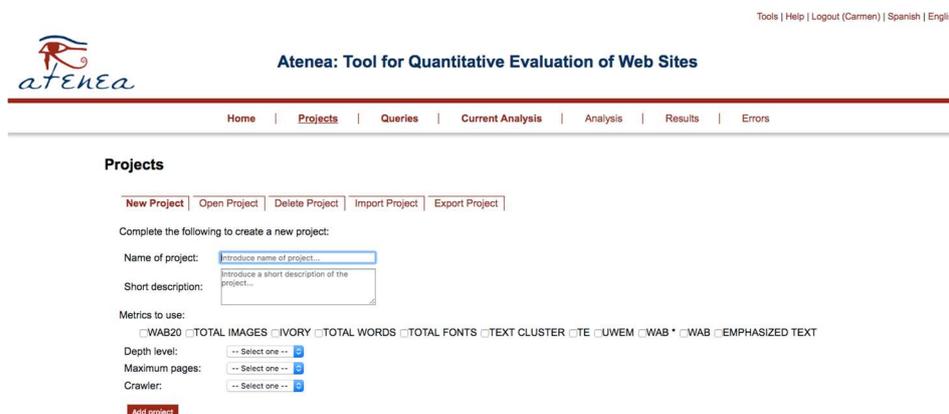


Imagen 99. Sección de proyectos

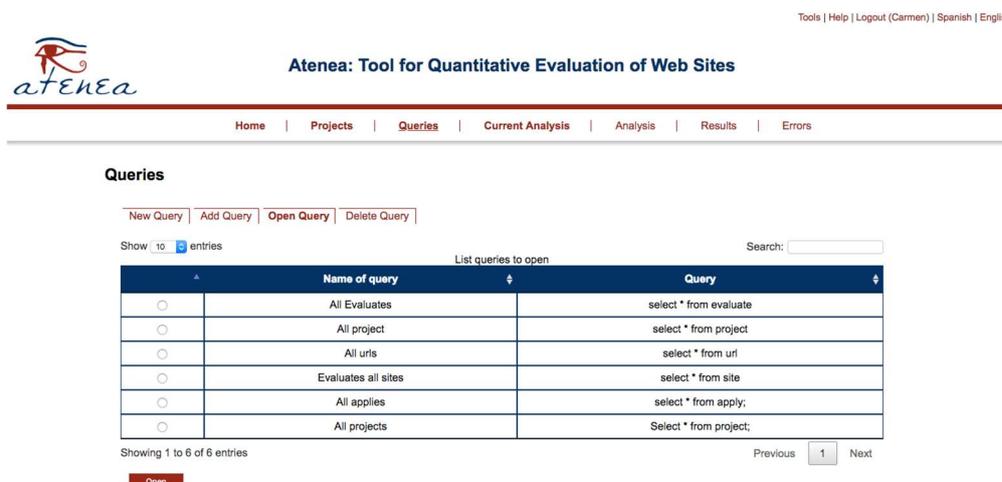


Imagen 100. Sección de consultas

En cuanto a la participación en redes sociales, se debería añadir una sección que encapsule los enlaces a las distintas redes sociales, con el fin de conseguir llegar a un mayor número de usuarios. Si no existen cuentas de estas nuevas tecnologías, se deberían crear y compartir.

8.2.3 Accesibilidad

A continuación, se estudiarán aspectos asociados a la accesibilidad que el sitio web de Atenea debe presentar.

8.2.3.1 Accesibilidad para usuarios con discapacidad

En este apartado, se analizará la página web atendiendo a discapacidades visuales y auditivas.

8.2.3.1.1 Visuales

Con la propia funcionalidad de los navegadores se puede aumentar el tamaño de letra del sitio web, para facilitar el visionado a personas con alguna discapacidad visual, sin ningún problema y la estructura de la página no varía.

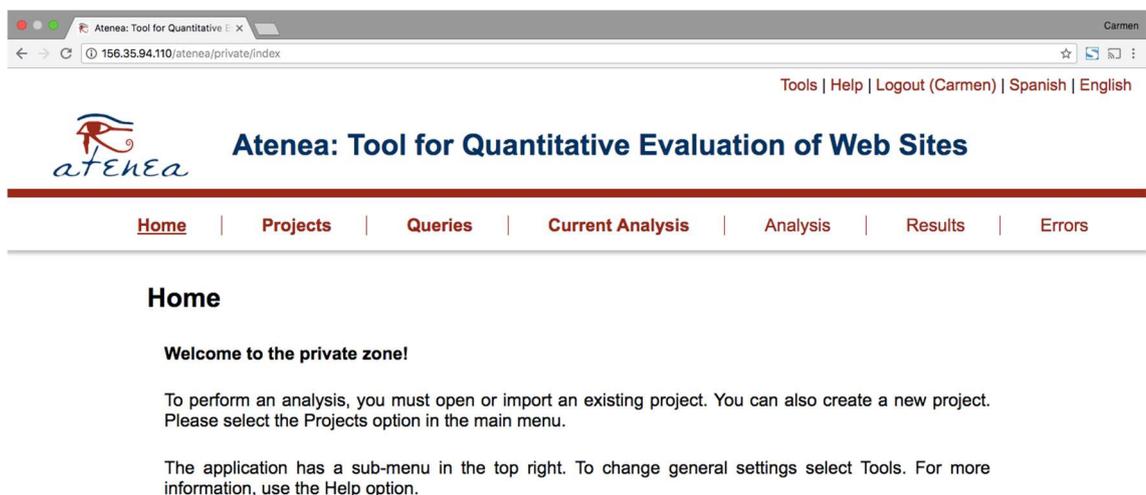


Imagen 101. Aumento del tamaño de letra

En cuanto a los colores utilizados en la página web, se hace un análisis con la herramienta CCA (Colour Contrast Analyzer s.f.) que nos permite identificar si dicha web es apta para personas con problemas visuales.

En general, los colores principales de la página son el azul y el rojo, así como el blanco y el negro.

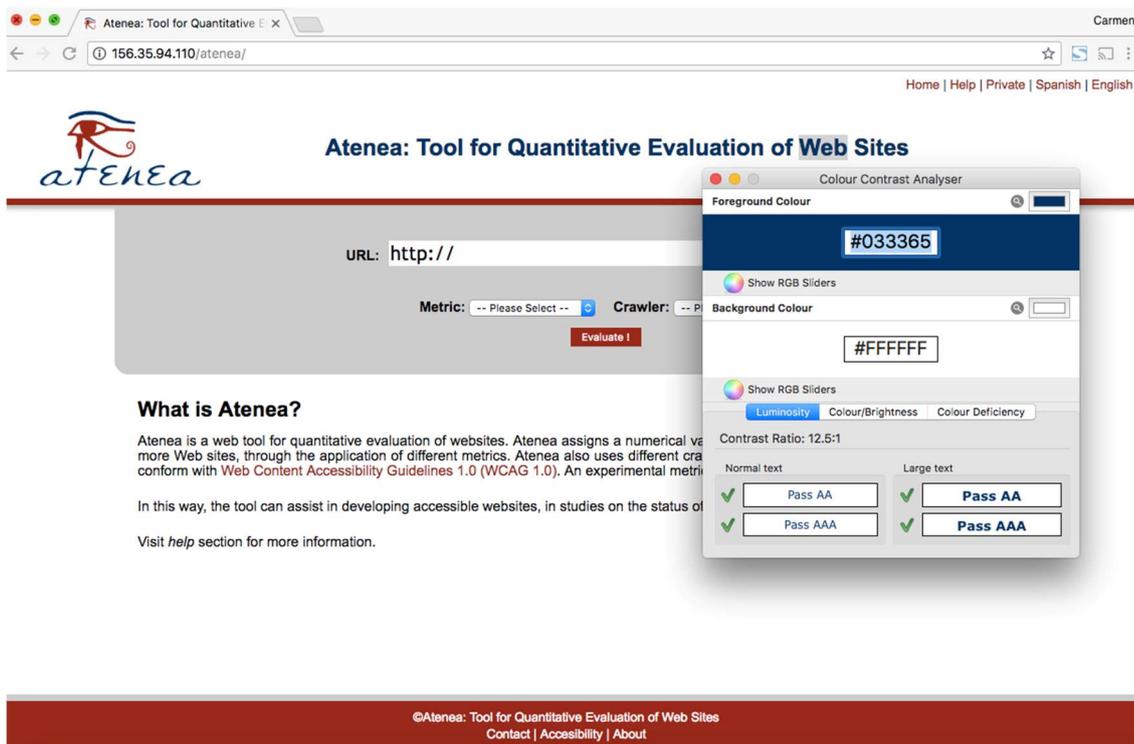


Imagen 102. Contraste azul y blanco

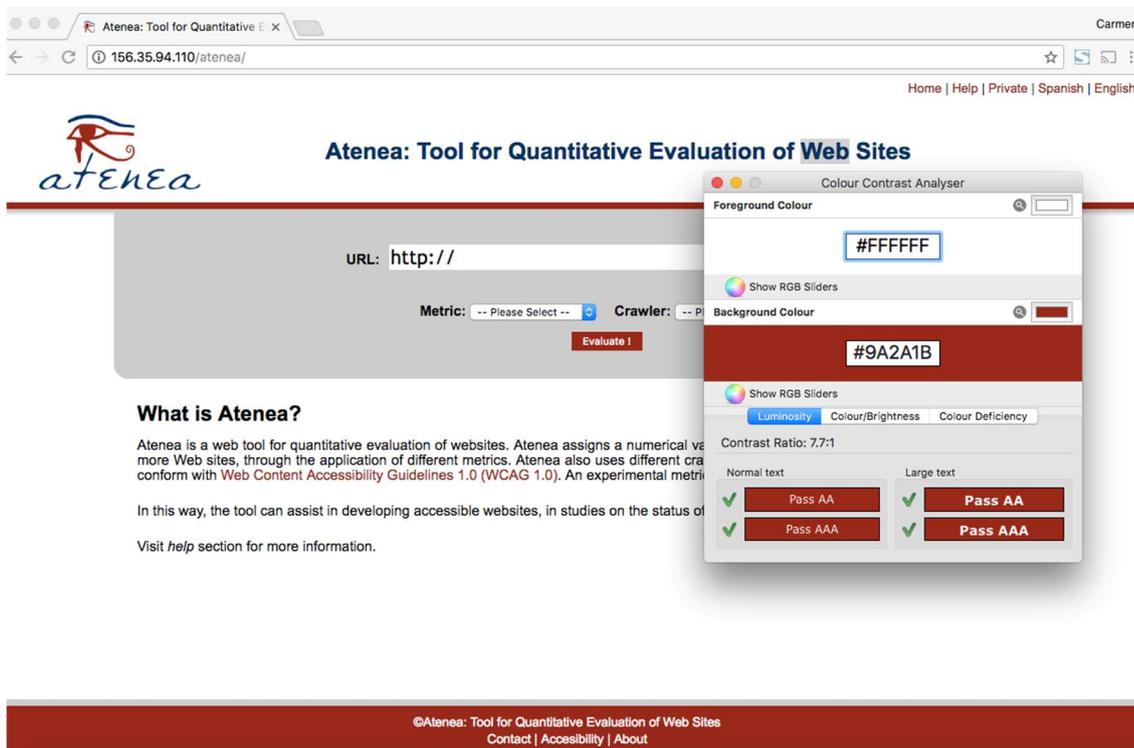


Imagen 103. Contraste rojo y blanco

A continuación, se adjunta una imagen, con el fin de que nos hagamos una idea de cómo se vería la página web si se tuviera las enfermedades visuales más comunes:

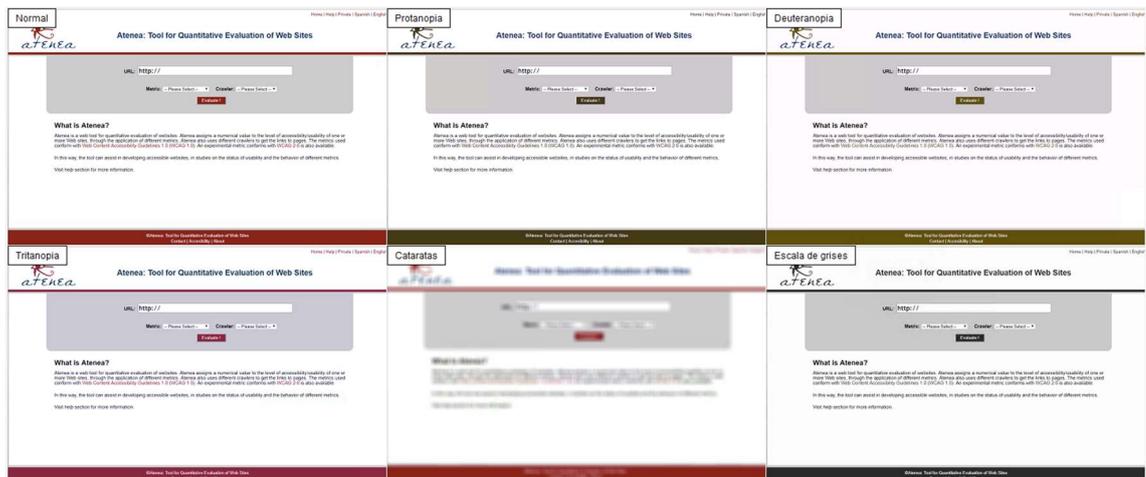


Imagen 104. Atenea desde enfermedades visuales comunes

8.2.3.1.2 Auditivas

Se ha utilizado el narrador propio del ordenador, para comprobar que, efectivamente, se puede leer todo el contenido web, para que las personas sin visión o con visión limitada puedan conocer tanto la página web como su contenido. Y así, poder utilizarla.

8.2.3.2 Acceso a navegadores no gráficos

Se utilizan dos navegadores no gráficos para comprobar que Atenea es accesible a través de este tipo de navegadores. A continuación, se puede visualizar como se ve la web en ambos navegadores Elinks y Lynx, respectivamente.

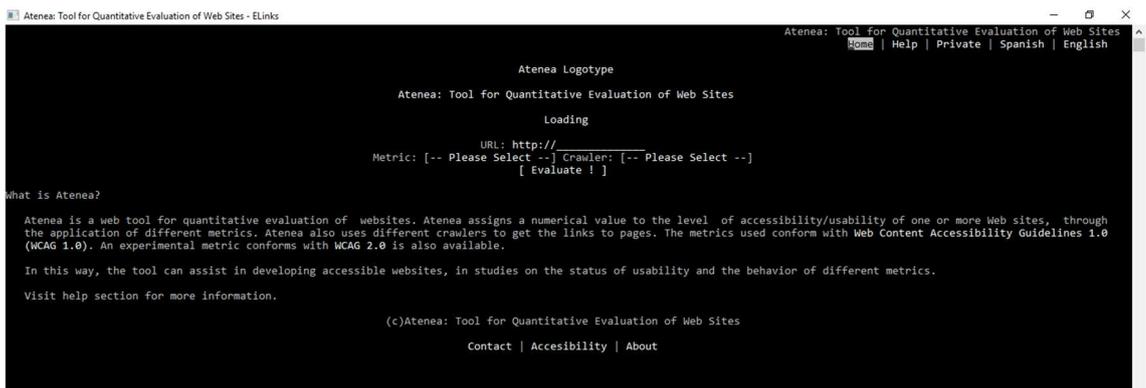


Imagen 105. Atenea desde navegador de texto Elinks

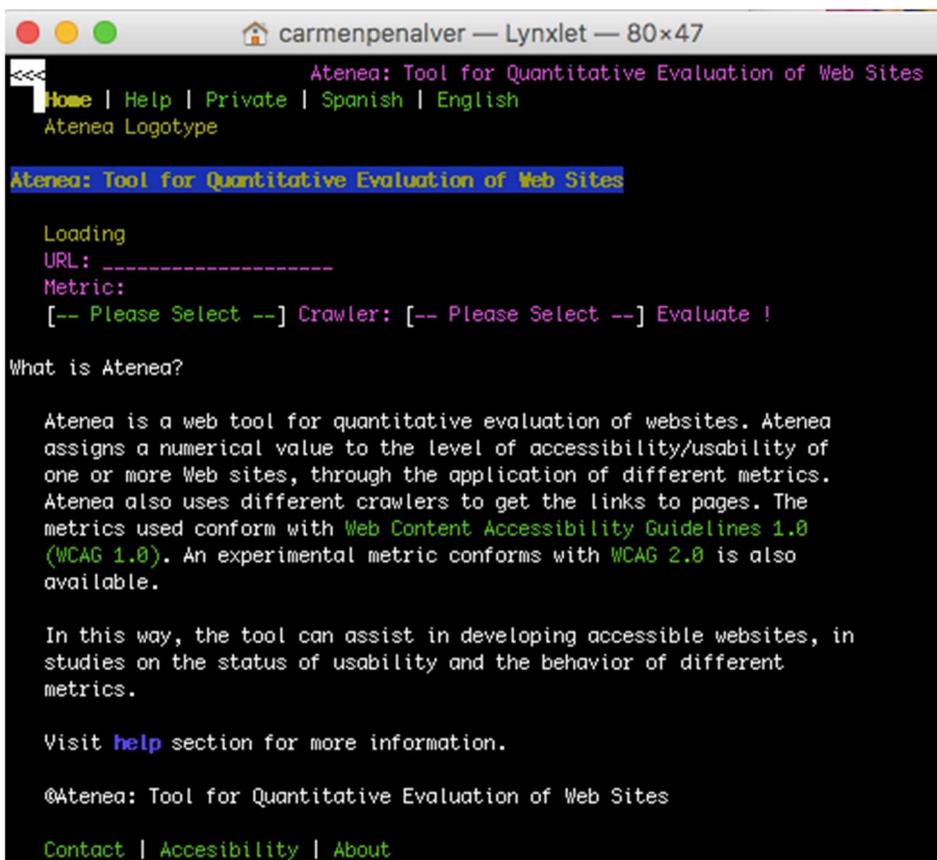


Imagen 106. Atenea desde navegador de texto Lynx

8.2.3.3 Test de accesibilidad

En este apartado se hará un examen al sitio web de Atenea, atendiendo a la accesibilidad que debe tener una web y, para ello, se utilizarán varias herramientas disponibles para este fin, ya que no nos debemos fiar de una única herramienta.

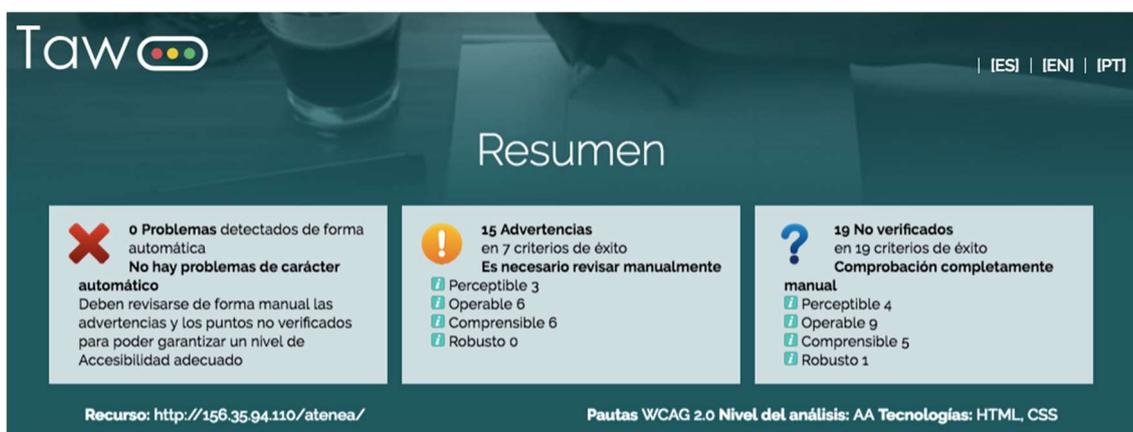


Imagen 107. Test accesibilidad TAW

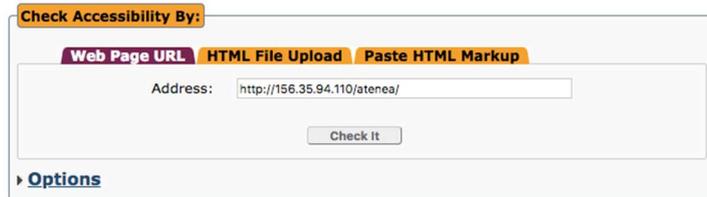


Imagen 108. Test accesibilidad AChecker

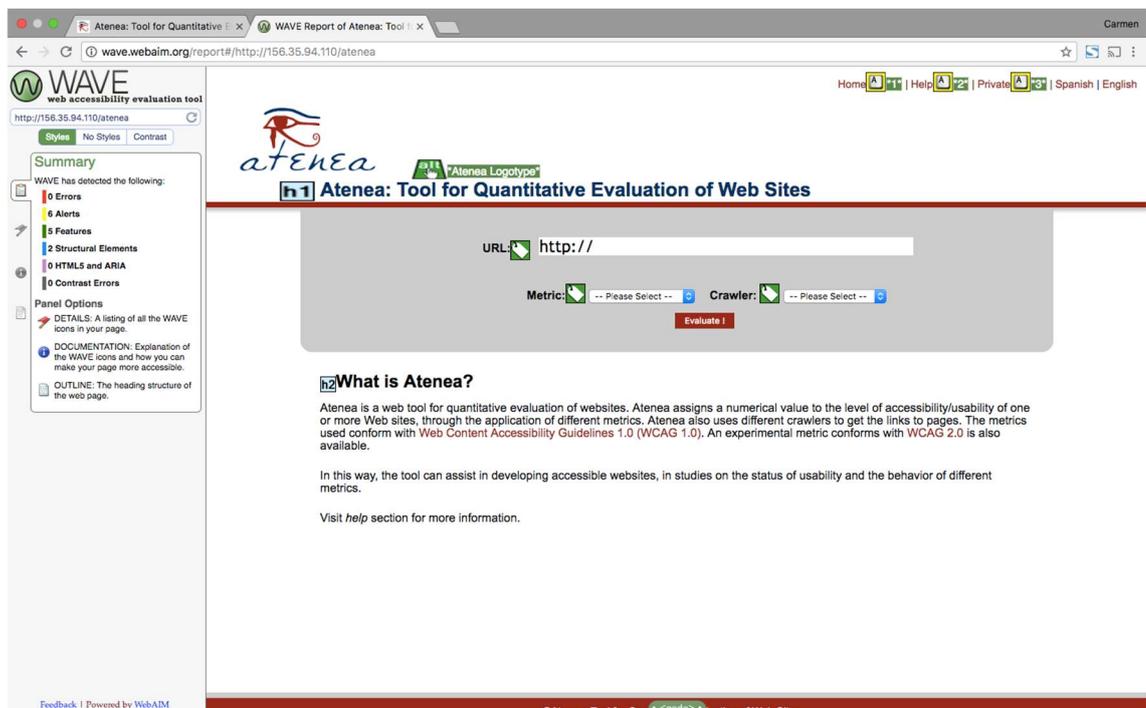


Imagen 109. Test accesibilidad WAVE

8.2.4 Funcionalidad

A continuación, se estudiarán aspectos asociados a la funcionalidad que el sitio web de Atenea debe presentar.

8.2.4.1 Aspectos de búsqueda

Atenea no cuenta con un mecanismo de búsqueda global, sin embargo, cada vez que se desea mostrar algún tipo de información amplia, se proporciona un sistema individual de búsqueda y filtrado. Esto es, cuando Atenea muestra al usuario el listado de los proyectos, las consultas,

los resultados de los análisis o sus errores, cuenta con un sistema que permite realizar búsquedas rápidas.

Results

Opened Project: **Universidades** created at 2018-06-26T15:35 (Close)

[Export XML](#) [Export PDF](#) [Export XLS](#)

Show entries Search:

Results for analysis

WebSite	Metric	score
http://www.uniovi.es/	TOTAL WORDS	1407.5
http://www.uniovi.es/	TOTAL FONTS	0.0
http://www.uniovi.es/	TOTAL IMAGES	22.8

Showing 1 to 3 of 3 entries (filtered from 610 total entries)
 (If the score is -1.0, the website hasn't been analyzed)

Previous Next

Imagen 110. Mecanismos de búsqueda y filtrado

8.2.4.2 Aspectos de navegación y exploración

No existe una zona habilitada para poder conocer en que parte de la web estás, es decir, no cuenta con una miga de pan, pero el título de cada página se identifica con la sección en la que uno se encuentra y los enlaces se activan.



Atenea: Tool for Quantitative Evaluation of Web Sites

Home | Projects | Queries | Current Analysis | Analysis | Results | Errors

All Errors

Opened Project: **a** created at 2018-06-10T13:56:49 (Close)

Imagen 111. Posición actual

Hay que comentar que existe un scroll vertical, ya que en algunas páginas tienen bastante contenido a lo largo y es necesario que se vean completamente.

Por lo general los enlaces que contiene la página web son todos bastante descriptivos y clarificativos.

8.2.5 Contenidos

Tras probar y analizar la web de Atenea, se puede concluir que los contenidos son coherentes, ya que se trata de una web para analizar aspecto de accesibilidad y usabilidad y, efectivamente, todas las secciones de la web tienen sentido.

Cuenta diferentes secciones propias de este ámbito, entre éstas, aparecen secciones para analizar.

8.2.6 Confiabilidad

Tras navegar por la web se puede confirmar que no existen enlaces rotos, inválidos ni no implementados.

La información y los contenidos parece estar correctamente actualizada, ya que las métricas con las que se pueden medir los sitios son actuales, pero es recomendable seguir, en un futuro, actualizando estos aspectos.

Finalmente, se pasarán dos nuevos test para comprobar si el sitio web cumple la utilización de estándares del W3C (World Wide Web Consortium s.f.).

Errors found while checking this document as XHTML 1.0 Strict!	
Result:	5 Errors
Address:	<input type="text" value="http://156.35.94.110/atenea/"/>
Encoding:	iso-8859-1 (detect automatically)
Doctype:	XHTML 1.0 Strict (detect automatically)
Root Element:	html
Root Namespace:	http://www.w3.org/1999/xhtml

Imagen 112. XHTML válido

La validación anterior no es correcta, debido al uso de atributos y etiquetas introducidas por el código generado mediante los tags de Spring.

![\"¡CSS Válido!\"](\"http://jigsaw.w3.org/css-validator/images/vcss\")

Resultados del Validador CSS del W3C para <http://156.35.94.110/atenea/> (CSS versión 3 + SVG)

¡Enhorabuena! No error encontrado.

¡Este documento es CSS versión 3 + SVG válido!

Puede mostrar este icono en cualquier página que valide para que los usuarios vean que se ha preocupado por crear una página Web interoperable. A continuación se encuentra el XHTML que puede usar para añadir el icono a su página Web:

```
<p>
<a href="http://jigsaw.w3.org/css-validator/check/referer">

</a>
</p>
```

```
<p>
<a href="https://jigsaw.w3.org/css-validator/check/referer">

</a>
</p>
```

Imagen 113. CSS válido

8.3 Pruebas de Rendimiento

8.3.1 Comparativa de rendimiento: Zona pública

Se han seleccionado aleatoriamente un conjunto de métricas, así como un crawler determinado, para realizar diversos análisis sobre el sitio web de la universidad de Oviedo.

APLICACIÓN	ENLACE	MÉTRICA	TIPO	CRAWLER	TIEMPO DE RESPUESTA (SEG)
ATENEA 3.0	http://www.uniovi.es	EMPHASIDE Z TEXT	Cualitativa	Basic	12,66
ATENEA 4.0	http://www.uniovi.es	EMPHASIDE Z TEXT	Cualitativa	Basic	10,76
ATENEA 3.0	http://www.uniovi.es	TOTAL IMAGES	Cualitativa	Basic	12,14
ATENEA 4.0	http://www.uniovi.es	TOTAL IMAGES	Cualitativa	Basic	8,27
ATENEA 3.0	http://www.uniovi.es	TE	Cuantitativa	Basic	12,76
ATENEA 4.0	http://www.uniovi.es	TE	Cuantitativa	Basic	8,89
ATENEA 3.0	http://www.uniovi.es	WAB*	Cuantitativa	Basic	12,75
ATENEA 4.0	http://www.uniovi.es	WAB*	Cuantitativa	Basic	11,03

Tabla 162. Comparativa de rendimiento: zona pública

Como se puede observar en el siguiente gráfico, efectivamente, los cambios acometidos en el sistema de Atenea han desembocado en una mejora de los tiempos de respuesta en los análisis de la zona pública de la aplicación.

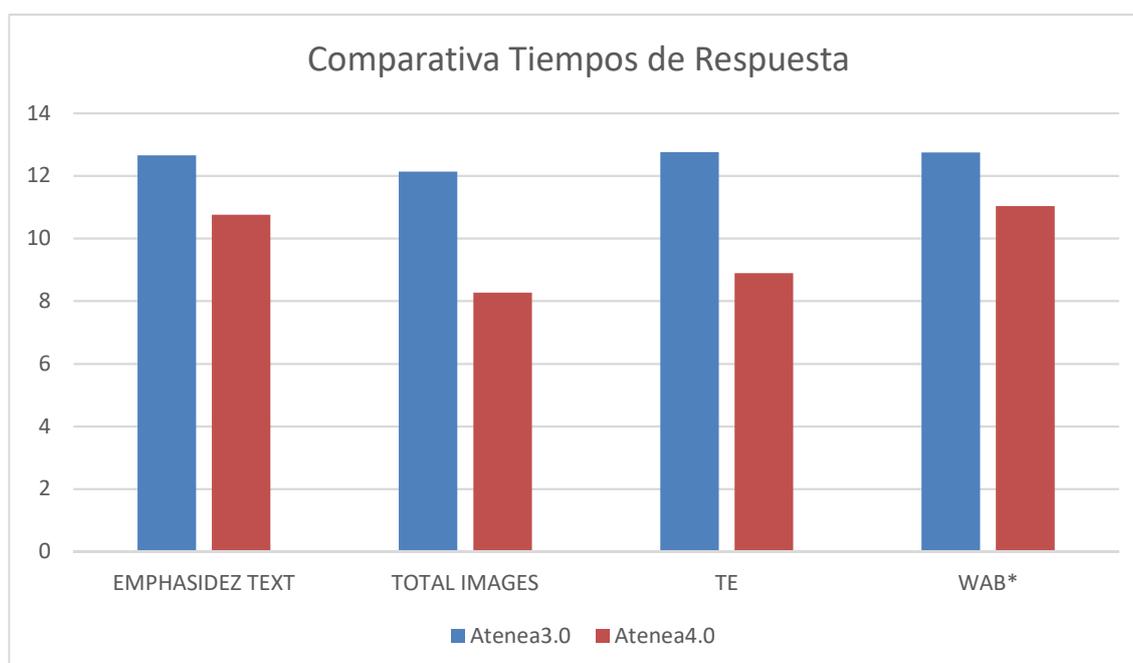


Imagen 114. Comparativa tiempos de respuesta zona pública

8.3.2 Comparativa de rendimiento: Zona privada

Para estudiar el rendimiento de la zona privada, en primer lugar, se crea un proyecto con las siguientes características:

- Nivel de profundidad: 2
- Número máximo de páginas: 15
- Métricas: EMPHASIZED TEXT, TOTAL IMAGES, TOTAL WORDS, TOTAL FONTS, TEXT CLUSTER, IVORY, TE, UWEM, WAB, WAB* y WAB20.
- Crawler: Basic
- Número de hilos: 15

APLICACIÓN	ENLACE	FICHERO	TIEMPO DE RESPUESTA (SEG)
ATENEA 3.0	http://www.uniovi.es	No	100,15
ATENEA 4.0	http://www.uniovi.es	No	73,17
ATENEA 3.0	http://www.uniovi.es	Si	95,76
ATENEA 4.0	http://www.uniovi.es	Si	81,99

Imagen 115. Comparativa de rendimiento: zona privada

Como se puede observar en el siguiente gráfico, efectivamente, los cambios acometidos en el sistema de Atenea han desembocado en una mejora de los tiempos de respuesta en los análisis de la zona privada de la aplicación.

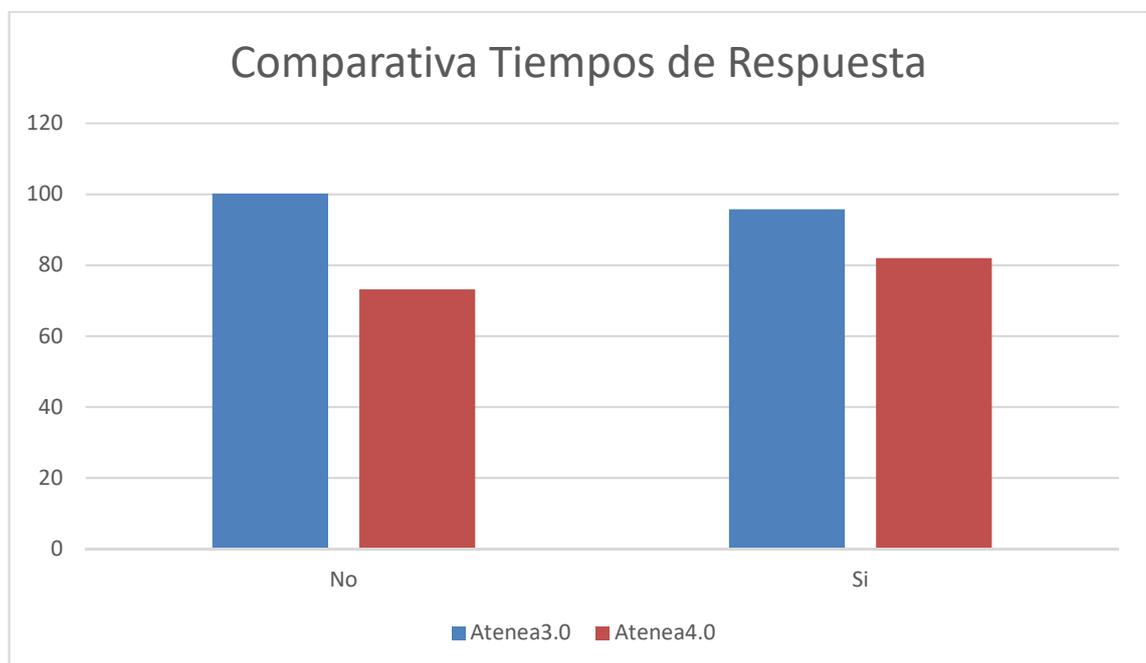


Imagen 116. Comparativa tiempos de respuesta zona privada

Capítulo 9. Estudio de la accesibilidad y usabilidad de las universidades españolas utilizando Atenea

Tras finalizar el desarrollo del presente proyecto, es decir, cuando se cuenta con una nueva implementación de Atenea, mejorada y actualizada, se decide hacer uso de sus funciones, con la finalidad de estudiar aspectos de usabilidad y accesibilidad, sobre un conjunto de datos.

El conjunto de datos escogidos consiste en una muestra significativa de las páginas webs asociadas a las principales universidades españolas.

	SITIO WEB
UNIVERSIDAD DE ALCALÁ DE HENARES	https://www.uah.es/es/
UNIVERSIDAD ALFONSO X EL SABIO	http://www.uax.es/
UNIVERSIDAD DE ALICANTE	http://www.ua.es
UNIVERSIDAD ANTONIO DE NEBRIJA	https://www.nebrija.com/
UNIVERSIDAD AUTÓNOMA DE BARCELONA	http://www.uab.es/
UNIVERSIDAD AUTÓNOMA DE MADRID	http://www.uam.es/
UNIVERSIDAD DE BARCELONA	http://www.ub.edu/web/ub/ca/
UNIVERSIDAD DE BURGOS	http://www.ubu.es/
UNIVERSIDAD DE CADÍZ	http://www.uca.es/
UNIVERSIDAD DE CANTABRIA	http://www.unican.es/
UNIVERSIDAD CARLOS III DE MADRID	http://www.uc3m.es/
UNIVERSIDAD DE CASTILLA LA MANCHA	http://www.uclm.es/
UNIVERSIDAD COMPLUTENSE DE MADRID	http://www.ucm.es/
UNIVERSIDAD DE CÓRDOBA	http://www.uco.es/
UNIVERSIDAD DE LA CORUÑA	https://www.udc.es/
UNIVERSIDAD DE DEUSTO	http://www.deusto.es
UNIVERSIDAD EUROPEA DE MADRID	http://www.uem.es/
UNIVERSIDAD DE EXTREMADURA	http://www.unex.es/
UNIVERSITAT DE GIRONA	http://www.udg.es/
UNIVERSIDAD DE GRANADA	http://www.ugr.es/
UNIVERSIDAD DE HUELVA	http://www.uhu.es/
UNIVERSIDAD INTERNACIONAL DE ANDALUCIA	https://www.unia.es/
UNIVERSIDAD INTERNACIONAL MENÉNDEZ PELAYO	http://www.uimp.es/
UNIVERSIDAD DE LAS ISLAS BALEARES	http://www.uib.cat/
UNIVERSIDAD DE JAÉN	http://www.ujaen.es/
UNIVERSITAT JAUME I	http://www.uji.es/
UNIVERSIDAD DE LA LAGUNA	http://www.ull.es/
UNIVERSIDAD DE LEÓN	http://www.unileon.es
UNIVERSIDAD DE LLEIDA	http://www.udl.es
UNIVERSIDAD DE MÁLAGA	http://www.uma.es/
UNIVERSIDAD MIGUEL HERNÁNDEZ DE ELCHE	http://www.umh.es/
UNIVERSIDAD DE MONDRAGON	https://www.mondragon.edu/es/inicio

UNIVERSIDAD DE MURCIA	http://www.um.es/
UNIVERSIDAD NACIONAL DE EDUCACIÓN A DISTANCIA	http://www.uned.es/
UNIVERSIDAD DE NAVARRA	http://www.unav.edu/
UNIVERSIDAD DE OVIEDO	http://www.uniovi.es/
UNIVERSITAT OBERTA DE CATALUNYA	http://www.uoc.edu/portal/es/
UNIVERSIDAD DEL PAÍS VASCO	https://www.ehu.eus/es/
UNIVERSIDAD DE LAS PALMAS DE GRAN CANARIA	http://www.ulpgc.es/
UNIVERSITAT POLITÈCNICA DE CATALUNYA	http://www.upc.es/
UNIVERSIDAD POLITÈCNICA DE MADRID	http://www.upm.es/
UNIVERSITAT POLITÈCNICA DE VALENCIA	http://www.upv.es/
UNIVERSIDAD POMPEU FABRA	http://www.upf.edu/
UNIVERSIDAD PONTIFICIA COMILLAS (ICAI-ICADE)	http://www.comillas.edu/es/
UNIVERSIDAD PONTIFICIA DE SALAMANCA	http://www.upsa.es/
UNIVERSIDAD PÚBLICA DE NAVARRA	http://www.unavarra.es/
UNIVERSITAT RAMON LLULL	http://www.url.es/
UNIVERSIDAD DE LA RIOJA	http://www.unirioja.es/
UNIVERSITAT ROVIRA I VIRGILI	http://www.urv.cat/ca/
UNIVERSIDAD DE SALAMANCA	http://www.usal.es/
UNIVERSIDAD SAN PABLO CEU	http://www.ceu.es/
UNIVERSIDAD DE SANTIAGO DE COMPOSTELA	http://www.usc.es/
UNIVERSIDAD DE SEVILLA	http://www.us.es/
UNIVERSITAT DE VALENCIA	http://www.uv.es/
UNIVERSIDAD DE VALLADOLID	http://www.uva.es/export/sites/uva/
UNIVERSIDAD DE VIGO	http://www.uvigo.es/uvigo/Universidad.html
UNIVERSIDAD DE ZARAGOZA	http://www.unizar.es

Tabla 163. Conjunto de enlaces a analizar

Para realizar este análisis, se ha creado un proyecto exclusivo con las siguientes características:

- Nombre: Universidades
- Descripción: Análisis sobre enlace de universidades
- Nivel de profundidad: 2
- Número máximo de páginas: 20
- Crawler: Basic
- Métricas: EMPHASIZED TEXT, TOTAL IMAGES, TOTAL WORDS, TOTAL FONTS, TEXT CLUSTER, IVORY, TE, UWEM, WAB, WAB* y WAB20.

Una vez creado el proyecto, este, se ha abierto automáticamente y se ha procedido a analizar un fichero con el conjunto de enlaces seleccionados.

A continuación, se muestra el resultado del análisis con todas las puntuaciones que han obtenido todos y cada uno de los enlaces de las universidades, atendiendo a todas las métricas asociadas al proyecto.

SITES	WAB20	TOTAL IMAGES	IVORY	TOTAL WORDS	TOTAL FONTS	TEXT CLUSTER	TE	UWEM	WAB *	WAB	EMPHASIZED TEXT
HTTP://PORTAL.UNED.ES/PORTAL/PAGE?_PAGEID=93,1&_DAD=PORTAL&_SCHEMA=PORTAL	-1.0	-1.0	-1.0	-1.0	-1.0	-1.0	-1.0	-1.0	-1.0	-1.0	-1.0
HTTP://UNIVERSIDADEUROPEA.ES	1.0	0.0	0.274169	8.0	0.0	0.0	-1.0	0.714286	11.0	3.0	58240.0
HTTP://WWW.CEU.ES/	-1.0	-1.0	-1.0	-1.0	-1.0	-1.0	-1.0	-1.0	-1.0	-1.0	-1.0
HTTP://WWW.COMILLAS.EDU/ES/	-1.0	-1.0	-1.0	-1.0	-1.0	-1.0	-1.0	-1.0	-1.0	-1.0	-1.0
HTTP://WWW.DEUSTO.ES	2.0	0.0	0.129	0.0	0.0	0.0	-1.0	0.5	9.0	5.0	0.0
HTTP://WWW.UA.ES	-1.0	-1.0	-1.0	-1.0	-1.0	-1.0	-1.0	-1.0	-1.0	-1.0	-1.0
HTTP://WWW.UAB.CAT/	2.12027	24.25	0.407745	1573.65	2.0	1779.65	0.365462	0.137267	15.6712	5.70468	-1.0
HTTP://WWW.UAM.ES/UAM/HOME.HTM?LANGUAGE=ES	-1.0	-1.0	-1.0	-1.0	-1.0	-1.0	-1.0	-1.0	-1.0	-1.0	-1.0
HTTP://WWW.UAX.ES/	-1.0	-1.0	-1.0	-1.0	-1.0	-1.0	-1.0	-1.0	-1.0	-1.0	-1.0
HTTP://WWW.UB.EDU/WEB/UB/CA/	-1.0	-1.0	-1.0	-1.0	-1.0	-1.0	-1.0	-1.0	-1.0	-1.0	-1.0
HTTP://WWW.UBU.ES/	1.58407	66.95	1.35126	1515.8	0.0	63.75	0.357938	0.0875711	9.69672	4.35332	-1.0
HTTP://WWW.UC3M.ES/	2.0	0.0	0.129	0.0	0.0	0.0	-1.0	0.5	9.0	5.0	0.0
HTTP://WWW.UCA.ES/	1.73774	46.95	0.409421	1119.05	0.0	399.9	0.584845	0.170611	13.6966	6.23498	438.454
HTTP://WWW.UCLM.ES/	2.0	0.0	0.129	0.0	0.0	0.0	-1.0	0.5	9.0	5.0	0.0
HTTP://WWW.UCM.ES/	0.529659	9.0	0.346231	498.9	0.0	15.2	0.398148	0.0241566	1.88038	0.933713	1592.66
HTTP://WWW.UCO.ES/	3.0644	37.2	0.411518	733.2	1.0	8.85	0.429326	0.197467	-1.0	-1.0	3950.64
HTTP://WWW.UDG.EDU/	2.69248	8.4	0.750123	1039.55	0.0	30.3	0.405882	0.0754047	11.9689	4.63915	12930.4
HTTP://WWW.UDL.ES/CA/	1.42657	6.65	0.437214	1198.0	0.0	22.4	0.431078	0.0559006	6.37915	3.77016	247.968
HTTP://WWW.UGR.ES/	0.184513	8.25	0.419494	539.15	0.0	13.3	0.390805	0.0311516	2.49064	0.521421	867.597
HTTP://WWW.UHU.ES/	2.0	0.0	0.14926	10.0	0.0	0.0	-1.0	0.75	11.0	5.0	0.0
HTTP://WWW.UIB.CAT/	0.229431	8.65	0.499977	962.5	0.0	16.6	0.353741	0.110794	3.77143	1.39913	387.166
HTTP://WWW.UIIMP.ES/	2.53715	65.05	0.310993	2017.75	1.0	54.9	0.435484	0.175907	8.84312	3.5635	-1.0
HTTP://WWW.UJAEN.ES/	2.0	0.0	0.129	0.0	0.0	0.0	-1.0	0.5	9.0	5.0	0.0
HTTP://WWW.UJI.ES/	-1.0	-1.0	-1.0	-1.0	-1.0	-1.0	-1.0	-1.0	-1.0	-1.0	-1.0
HTTP://WWW.ULL.ES/	1.0	0.0	0.263441	7.0	0.0	0.0	-1.0	0.714286	11.0	3.0	59825.0

SITES	WAB20	TOTAL IMAGES	IVORY	TOTAL WORDS	TOTAL FONTS	TEXT CLUSTER	TE	UWEM	WAB *	WAB	EMPHASIZ ED TEXT
HTTP://WWW.ULPGC.ES/	-1.0	-1.0	-1.0	-1.0	-1.0	-1.0	-1.0	-1.0	-1.0	-1.0	-1.0
HTTP://WWW.UM.ES/	-1.0	-1.0	-1.0	-1.0	-1.0	-1.0	-1.0	-1.0	-1.0	-1.0	-1.0
HTTP://WWW.UMA.ES/	-1.0	-1.0	-1.0	-1.0	-1.0	-1.0	-1.0	-1.0	-1.0	-1.0	-1.0
HTTP://WWW.UMH.ES/	1.58157	37.25	1.16656	1414.6	1.0	431.45	0.38861	0.027702	7.53473	4.69521	-1.0
HTTP://WWW.UNAV.EDU/	2.403	154.7	0.433824	1536.7	0.0	185.4	0.482667	0.0942549	8.63695	5.0114	9600.02
HTTP://WWW.UNAVARRA.ES/	1.95476	14.85	0.354321	765.15	0.0	26.1	0.421479	0.0950245	10.8022	5.75989	-1.0
HTTP://WWW.UNEX.ES/	-1.0	-1.0	-1.0	-1.0	-1.0	-1.0	-1.0	-1.0	-1.0	-1.0	-1.0
HTTP://WWW.UNICAN.ES/	1.0	0.0	0.291127	10.0	0.0	0.0	-1.0	0.333333	7.0	3.0	39675.0
HTTP://WWW.UNILEON.ES	2.0	0.0	0.129	0.0	0.0	0.0	-1.0	0.5	9.0	5.0	0.0
HTTP://WWW.UNIOVI.ES/	2.12617	22.8	0.644865	1407.5	0.0	28.2	0.39207	0.0611401	9.72277	3.37761	-1.0
HTTP://WWW.UNIRIOJA.ES/	4.14035	19.7	0.511419	263.6	1.0	3.25	0.466555	0.306186	11.3424	7.51435	2627.34
HTTP://WWW.UNIZAR.ES	1.65165	15.65	0.203369	1563.55	0.0	18.1	0.491468	0.12476	7.84699	3.02825	222.85
HTTP://WWW.UOC.EDU/PORTAL/ES/	1.92259	21.1	0.645073	967.65	0.0	21.1	0.432712	0.38271	11.7998	4.68236	-1.0
HTTP://WWW.UPC.EDU/CA	0.853523	32.95	0.456586	911.85	0.0	35.95	0.397672	0.196041	6.88591	2.86858	-1.0
HTTP://WWW.UPF.EDU/	-1.0	-1.0	-1.0	-1.0	0.0	-1.0	-1.0	-1.0	-1.0	-1.0	-1.0
HTTP://WWW.UPM.ES/	1.38665	7.65	0.972568	1300.2	0.0	47.8	0.398693	0.0452098	4.78743	2.46036	110.999
HTTP://WWW.UPSA.ES/	-1.0	-1.0	-1.0	-1.0	-1.0	-1.0	-1.0	-1.0	-1.0	-1.0	-1.0
HTTP://WWW.UPV.ES/	0.65618	33.15	0.319446	1096.0	0.0	18.45	0.368881	0.139891	8.79705	2.84649	-1.0
HTTP://WWW.URL.EDU	3.14066		0.346704	624.95	0.0	5.25	0.478411	0.211828	8.22682	4.63548	-1.0
HTTP://WWW.URV.CAT/CA/	1.51587	12.5	1.02019	2855.15	0.0	78.1	0.550667	0.126251	10.5944	3.47208	38.5592
HTTP://WWW.US.ES/	1.24867	43.0	0.425453	378.35	1.0	10.9	0.367893	0.107271	8.76117	4.21557	192.683
HTTP://WWW.USAL.ES/	1.68773	4.55	0.321598	5879.45	1.0	96.8	0.619048	0.167701	9.64814	5.60927	138.09
HTTP://WWW.USC.ES/	0.565302	18.75	0.232418	820.25	1.0	21.95	0.392	0.13312	6.15944	1.37063	-1.0
HTTP://WWW.UV.ES/	-1.0	-1.0	-1.0	-1.0	-1.0	-1.0	-1.0	-1.0	-1.0	-1.0	-1.0
HTTP://WWW.UVA.ES/EXPORT/SITES/UVA/	1.95704	4.5	0.188623	1421.95	0.0	13.35	0.455172	0.105461	9.57892	3.59896	15.1773
HTTP://WWW.UVIGO.ES/UVIGO/UNIVERSIDAD.HTML	2.0	0.0	0.129	0.0	0.0	0.0	-1.0	0.5	9.0	5.0	0.0
HTTPS://WWW.EHU.EUS/ES/	2.66135	51.0	0.305483	1357.95	0.0	1696.3	0.539916	0.338647	9.96792	5.65977	-1.0
HTTPS://WWW.MONDRAGON.EDU/ES/INICIO	-1.0	-1.0	-1.0	-1.0	-1.0	-1.0	-1.0	-1.0	-1.0	-1.0	-1.0
HTTPS://WWW.NEBRIJA.COM/	1.73156	26.35	0.553612	835.4	1.0	40.2	0.460922	0.142452	9.25484	2.83137	-1.0

SITES	WAB20	TOTAL IMAGES	IVORY	TOTAL WORDS	TOTAL FONTS	TEXT CLUSTER	TE	UWEM	WAB *	WAB	EMPHASIZED TEXT
HTTPS://WWW.UAH.ES/ES/	0.69809	-1.0	-1.0	-1.0	-1.0	101.9	-1.0	-1.0	-1.0	-1.0	-1.0
HTTPS://WWW.UDC.ES/	0.40898	4.15	0.35837	281.65	0.0	10.3	0.391489	0.0329301	1.70152	1.52829	-1.0
HTTPS://WWW.UNIA.ES/	3.06973	14.15	0.527878	542.5	0.0	33.75	0.368182	0.145268	7.75522	4.62202	1000.43

Tabla 164. Resultados al analizar los enlaces de las universidades de España

El análisis ha durado unas dos horas aproximadamente y durante el transcurso del mismo, se han recibido un conjunto de notificaciones por correo electrónico, informando a cerca del porcentaje de progreso del mismo. Entre estos, cabe destacar un aviso inicial y un aviso al finalizar el análisis.

```

miw.atenea@gmail.com 26/6/18
Atenea - Finish analyze URL's file.
Hi, Carmen The project analysis has finished with information: Name: Universidades Depth level: 2
Maximum pages: 20 Crawler: Basic The pages analyzed were the following: https://www.uah.es/es/ http://...

miw.atenea@gmail.com 26/6/18
Atenea - Information on the execution of analysis
Percentage analyzed: 99.68% --- TRACE --- http://www.uoc.edu/portal/es/ - TOTAL
IMAGES=TERMINATED http://www.ceu.es/ - IVORY=TERMINATED http://www.uca.es/ - TOTAL WORDS=TER...

miw.atenea@gmail.com 26/6/18
Atenea - Information on the execution of analysis
Percentage analyzed: 99.52% --- TRACE --- http://www.uoc.edu/portal/es/ - TOTAL
IMAGES=TERMINATED http://www.ceu.es/ - IVORY=TERMINATED http://www.uca.es/ - TOTAL WORDS=TER...

miw.atenea@gmail.com 26/6/18
Atenea - Information on the execution of analysis
Percentage analyzed: 99.52% --- TRACE --- http://www.uoc.edu/portal/es/ - TOTAL
IMAGES=TERMINATED http://www.ceu.es/ - IVORY=TERMINATED http://www.uca.es/ - TOTAL WORDS=TER...

miw.atenea@gmail.com 26/6/18
Atenea - Information on the execution of analysis
Percentage analyzed: 74.0% --- TRACE --- http://www.ceu.es/ - IVORY=TERMINATED http://www.uca.es/ -
TOTAL WORDS=TERMINATED http://www.upf.edu/ - TOTAL FONTS=TERMINATED http://www.url.es/ - IVOR...

miw.atenea@gmail.com 26/6/18
Atenea - Information on the execution of analysis
Percentage analyzed: 5.26% --- TRACE --- http://www.ulpgc.es/ - EMPHASIZED TEXT=TERMINATED
http://www.udg.es/ - EMPHASIZED TEXT=RUNNING http://www.urv.cat/ca/ - EMPHASIZED TEXT=TERMINA...

miw.atenea@gmail.com 26/6/18
Atenea - Start analyze URL's file.
Hi, Carmen The project analysis has started with information: Name: Universidades Depth level: 2
Maximum pages: 20 Crawler: Basic The pages to analyze are the following: https://www.uah.es/es/ http://w...
    
```

Tabla 165. Sistema de notificaciones

Para poder estudiar los resultados y llegar a una conclusión, se decide recoger una muestra más pequeña de estos resultados. Para ello, se realiza una selección aleatoria de las métricas escogidas para el estudio y se eliminan aquellos enlaces que por algún error no se han podido analizar. A continuación, se muestra el nuevo conjunto de datos seleccionado.

SITES	WAB
http://universidadeuropea.es	3.0
http://www.deusto.es	5.0
http://www.uab.cat/	5.70468
http://www.ubu.es/	4.35332
http://www.uc3m.es/	5.0
http://www.uca.es/	6.23498
http://www.uclm.es/	5.0
http://www.ucm.es/	0.933713
http://www.udg.edu/	4.63915
http://www.udl.es/ca/	3.77016
http://www.ugr.es/	0.521421
http://www.uhu.es/	5.0

http://www.uib.cat/	1.39913
http://www.uimp.es/	3.5635
http://www.ujaen.es/	5.0
http://www.ull.es/	3.0
http://www.umh.es/	4.69521
http://www.unav.edu/	5.0114
http://www.unavarra.es/	5.75989
http://www.unican.es/	3.0
http://www.unileon.es	5.0
http://www.uniovi.es/	3.37761
http://www.unirioja.es/	7.51435
http://www.unizar.es	3.02825
http://www.uoc.edu/portal/es/	4.68236
http://www.upc.edu/ca	2.86858
http://www.upm.es/	2.46036
http://www.upv.es/	2.84649
http://www.url.edu	4.63548
http://www.urv.cat/ca/	3.47208
http://www.us.es/	4.21557
http://www.usal.es/	5.60927
http://www.usc.es/	1.37063
http://www.uva.es/export/sites/uva/	3.59896
http://www.uvigo.es/uvigo/universidad.html	5.0
https://www.ehu.eus/es/	5.65977
https://www.nebrija.com/	2.83137
https://www.udc.es/	1.52829
https://www.unia.es/	4.62202

Tabla 166. Resultados de la métrica WAB

Antes de comenzar es necesario recordar que la métrica seleccionada se encarga de medir las violaciones a las pautas de la WCAG 1.0 y que un valor $WAB=0$ significa que el sitio evaluado tiene un nivel de accesibilidad aceptable. Un valor $WAB \leq 5.5$ significa que el sitio presenta pocas barreras de accesibilidad. Por el contrario, un valor superior a 5.5 implica que el sitio presenta muchas barreras de accesibilidad.

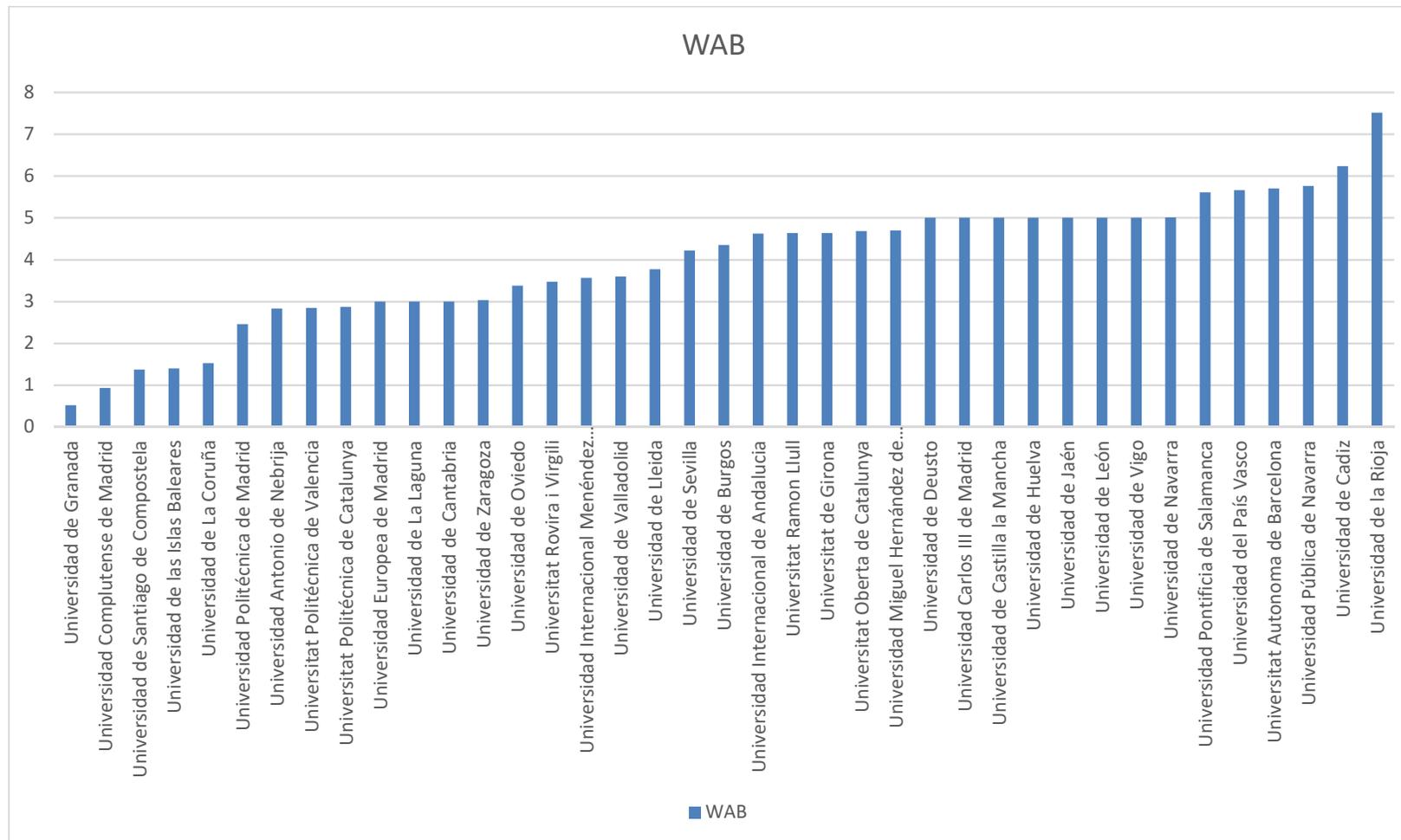


Imagen 117. Gráfico resultados métrica WAB

Tras este estudio y según la métrica WAB, se puede concluir que ninguna de las universidades españolas estudiadas presenta un nivel aceptable de accesibilidad. El 85% de las universidades presentan barreras de accesibilidad y el 15% presenta muchas barreras de este tipo. La universidad que presenta menos barreras, en cuanto a accesibilidad se refiere, es la universidad de Granada y la que más la universidad de la Rioja. La universidad de Oviedo ha obtenido una puntuación de 3,4 y, por lo tanto, sí presenta alguna barrera que podría impedir el acceso a un sector de la población, aunque no se ha situado en un mal puesto.

Finalmente, cabe destacar que la aplicación Atenea ha sido capaz de analizar alrededor del 72% de los enlaces introducidos.

Capítulo 10. Manuales del Sistema

10.1 Manual de Instalación

Para servir la aplicación de Atenea, se cuenta con una máquina física sobre la que se está ejecutando la plataforma de visualización, CitrixXenServer. En este caso la versión de XenServer es 6.0.2.

Dentro de esta máquina hay una máquina virtual con un sistema Debian Stretch y las siguientes características:

- Dos núcleos
- 2048 MB de RAM
- 40 GB de ROM
- Una IP fija y externa

Para instalar esta máquina es necesario realizar una serie de pasos, que se irán detallando a continuación:

En primer lugar, es necesario instalar la aplicación cliente que aporta XenServer para poder acceder a la máquina física y poder comenzar con toda la instalación.



Imagen 118. Instalar XenServer

Tras seguir los pasos de la instalación y añadir las credenciales oportunas se accederá a la página de administración de este sistema de virtualización.

Para obtener las credenciales es necesario ponerse en contacto con el dueño del servidor.

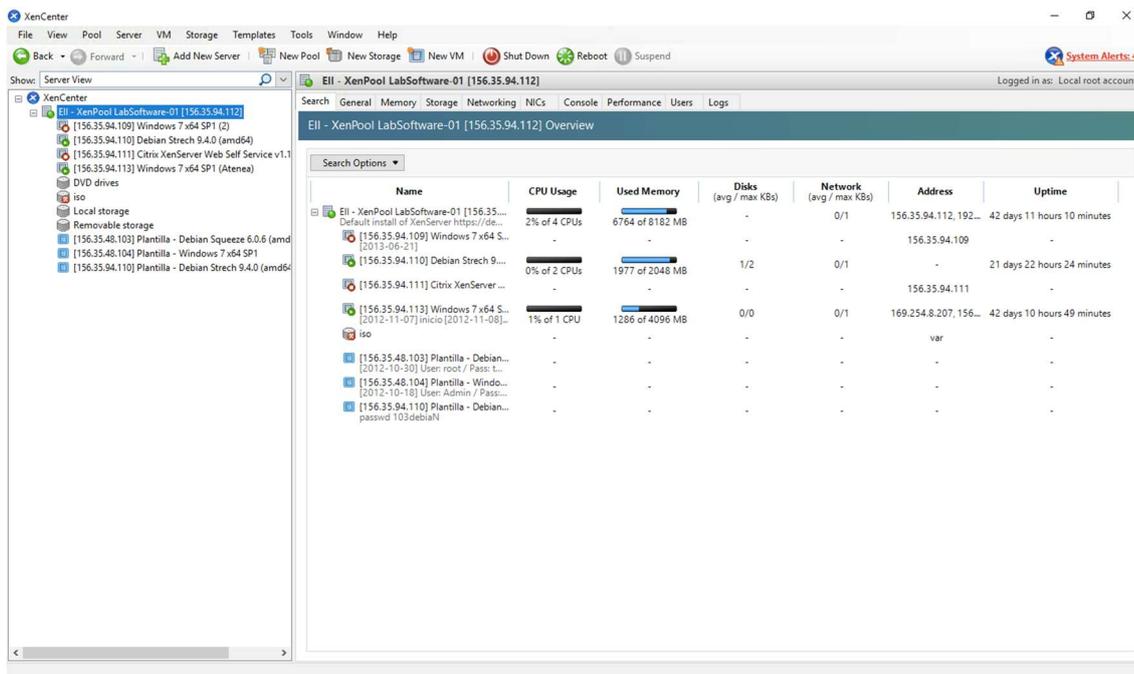


Imagen 119. Administración de XenServer

XenServer proporciona una serie de plantillas predeterminadas para poder instalar nuevas máquinas virtuales con unos determinados sistemas operativos. En este caso se desea instalar una distribución de Debian muy actual e incompatible con esta versión de XenServer, por lo que, es necesario realizar unos pasos previos para poder instalar la distribución deseada.

Paso 1 – Acceso

A través del servicio SSH accedemos remotamente al servidor XenServer e introducimos las credenciales oportunas.

Paso 2 – Configurar el nuevo repositorio

A través del comando **df -h** encontraremos un directorio con suficiente espacio como para añadir la nueva ISO a instalar.

Dentro del directorio con espacio suficiente se crea una nueva carpeta llamada iso, para poder encontrarla con mayor facilidad. Para esto, se utiliza el comando **mkdir**.

Finalmente, para que el sistema de virtualización XenServer reconozca este nuevo directorio como un repositorio de ISOs, es necesario ejecutar el siguiente comando:

```
xe sr-create name-label="iso" type=iso device-config:location=rutadirectorioconespacio/iso device-config:legacy_mode=true content-type=iso
```

Ahora aparecerá en XenServer un nuevo directorio donde se podrá añadir la nueva iso que se desea instalar.

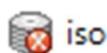


Imagen 120. Nuevo repositorio de ISOs

Paso 3 – Importar ISO Debian

Mediante un sistema SCP, como FileZilla, se accede al sistema de directorios de XenServer para poder importar en el directorio, anteriormente, creado el instalador de Debian Stretch.

Después de realizar estos pasos, mediante la opción de instalar otros medios, aparecerá la nueva ISO para poder ser instalada.

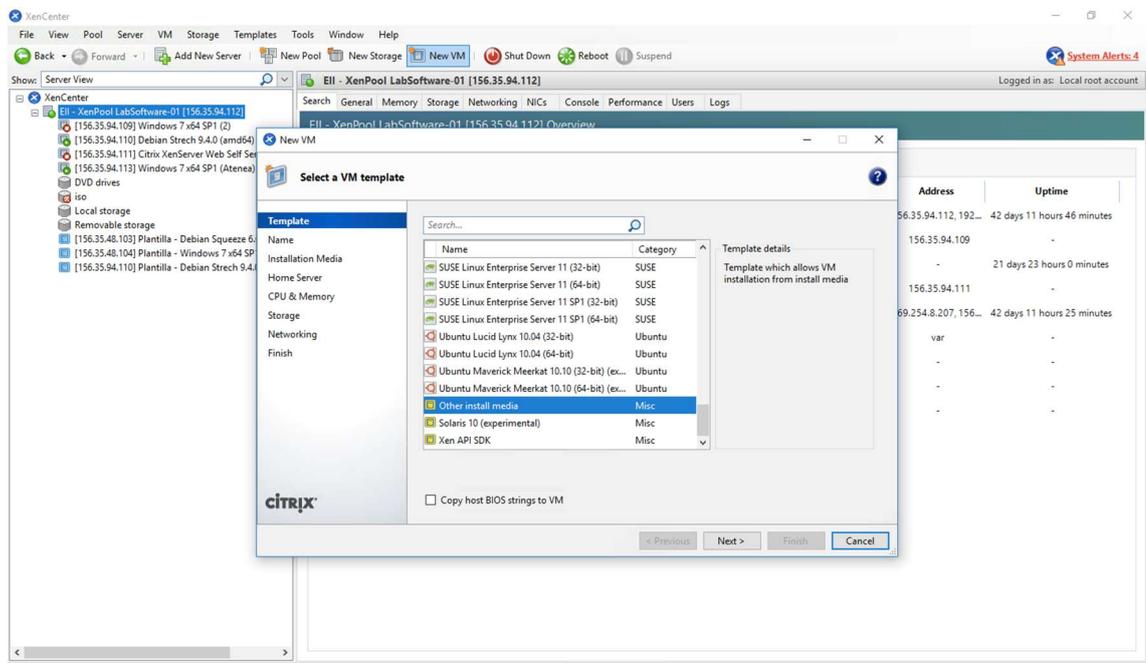


Imagen 121. Nueva máquina virtual

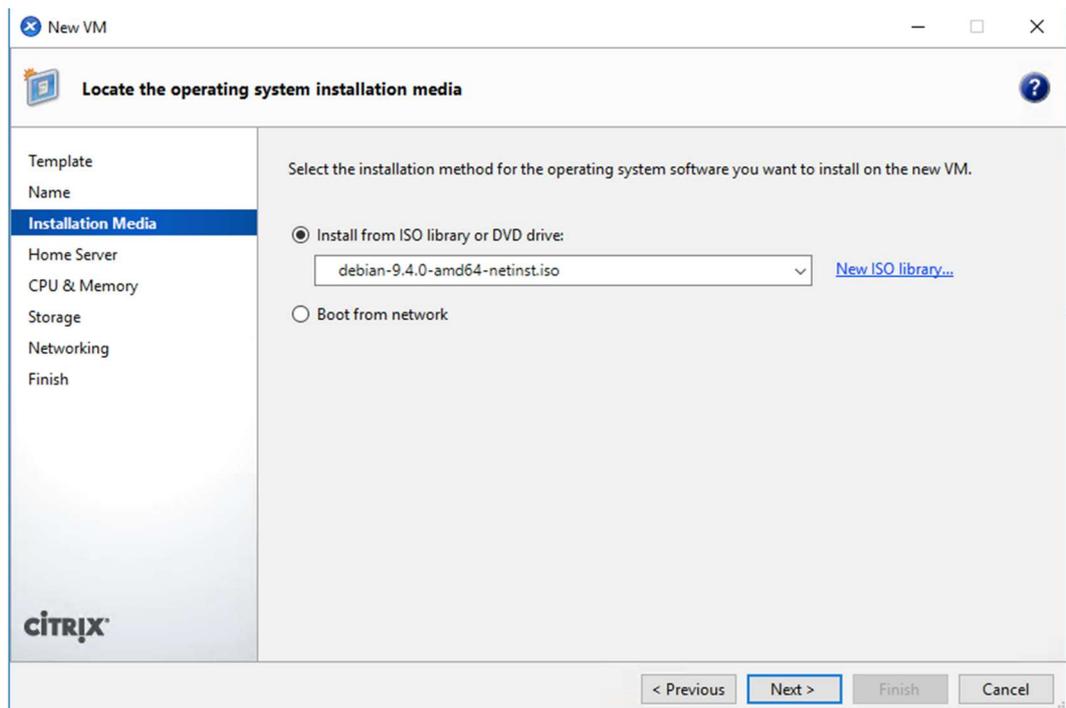


Imagen 122. Instalar Debian Stretch

Para completar el instalador, será necesario establecer las características de la nueva máquina virtual, anteriormente, citadas.

Finalmente, se siguen los pasos de instalación del sistema operativo Debian Stretch y ya contamos con una máquina virtual que hará como servidor de nuestra aplicación Atenea.

```
Debian GNU/Linux 9 debian-9-4-0-amd64 tty1
debian-9-4-0-amd64 login:
```

Imagen 123. Debian Stretch

Antes de comenzar a instalar el servidor de aplicaciones y el sistema de gestión de bases de datos, entre otros programas, se realizan los siguientes comandos, con el fin de mantener el sistema correctamente actualizado. **apt-get update apt-get upgrade**

10.1.1 Instalar Tomcat

Para instalar el servidor de aplicaciones Tomcat (Apache Tomcat s.f.) (MySQL s.f.), es necesario realizar una serie de pasos que se detallan a continuación.

Antes de instalar esta herramienta, es necesario conocer si se encuentra instalado el kit de desarrolladores Java con siguiente comando: **java -version**

Si no está instalado será necesario instalarlo, en nuestro sistema, de la siguiente manera:

```
apt-get update
apt-get install default-jdk
```

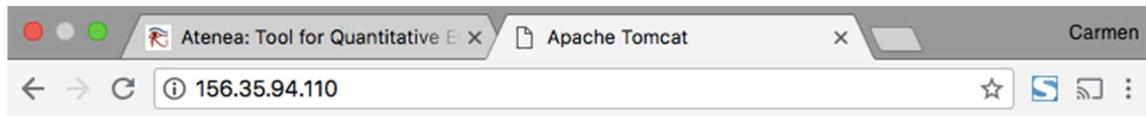
Imagen 124. Instalando JDK

A continuación, se procede a la instalación de Tomcat ejecutando lo siguiente:

```
apt-get update
apt-get install tomcat8
apt-get install tomcat8-admin tomcat8-examples tomcat8-docs
```

Imagen 125. Instalando Tomcat

Ahora, aunque la instalación no está todavía completa, ya se puede echar un vistazo de la página principal de Apache Tomcat.



It works !

If you're seeing this page via a web browser, it means you've setup Tomcat successfully. Congratulations!

This is the default Tomcat home page. It can be found on the local filesystem at:
`/var/lib/tomcat8/webapps/ROOT/index.html`

Tomcat8 veterans might be pleased to learn that this system instance of Tomcat is installed with `CATALINA_HOME` in `/usr/share/tomcat8` and `CATALINA_BASE` in `/var/lib/tomcat8`, following the rules from `/usr/share/doc/tomcat8-common/RUNNING.txt.gz`.

You might consider installing the following packages, if you haven't already done so:

tomcat8-docs: This package installs a web application that allows to browse the Tomcat 8 documentation locally. Once installed, you can access it by clicking [here](#).

tomcat8-examples: This package installs a web application that allows to access the Tomcat 8 Servlet and JSP examples. Once installed, you can access it by clicking [here](#).

tomcat8-admin: This package installs two web applications that can help managing this Tomcat instance. Once installed, you can access the [manager webapp](#) and the [host-manager webapp](#).

NOTE: For security reasons, using the manager webapp is restricted to users with role "manager-gui". The host-manager webapp is restricted to users with role "admin-gui". Users are defined in `/etc/tomcat8/tomcat-users.xml`.

Imagen 126. Página inicial Tomcat

Para poder acceder a la página de administración de Tomcat, es necesario crear un usuario con sus propias credenciales y, poder así, administrar el servidor. Para ello se debe modificar el fichero de configuración de usuarios de Tomcat.

```
GNU nano 2.7.4 Fichero: tomcat-users.xml

NOTE: The sample user and role entries below are intended for use with the
examples web application. They are wrapped in a comment and thus are ignored
when reading this file. If you wish to configure these users for use with the
examples web application, do not forget to remove the <!-- ..> that surrounds
them. You will also need to set the passwords to something appropriate.
-->
<!--
  <role rolename="tomcat"/>
-->
  <role rolename="manager-gui"/>
  <user username="tomcat" password="tomcat" roles="manager-gui"/>
<!-- <user username="both" password="<must-be-changed>" roles="tomcat,role1"/>
  <user username="role1" password="<must-be-changed>" roles="role1"/>
-->
</tomcat-users>
```

Imagen 127. Configurando usuarios

Siempre que se modifica algún fichero de configuración de esta herramienta, es necesario reiniciarla, mediante el siguiente comando:

```
/etc/init.d/tomcat8 restart
```

Imagen 128. Reiniciando Tomcat

Ahora para acceder a la zona de administración de nuestro servidor de aplicaciones, efectivamente, nos pide introducir el usuario y contraseña correcto. Se introduce **tomcat** **tomcat**, respectivamente, y ya podemos acceder.

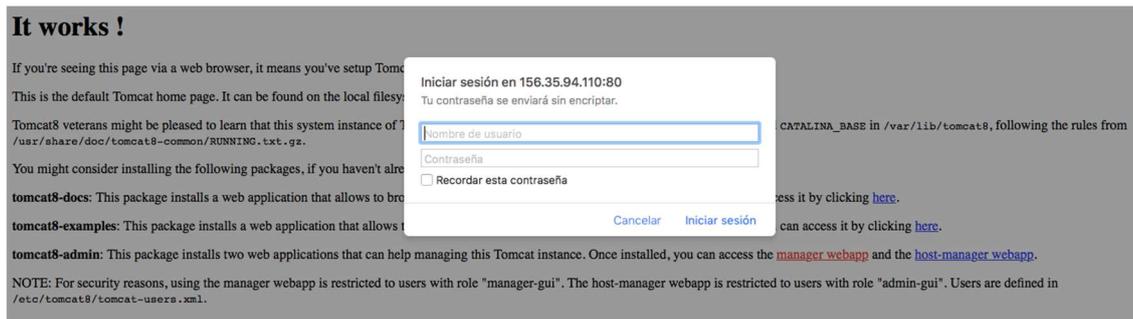


Imagen 129. Accediendo a Tomcat

Finalmente, ya tenemos correctamente instalado el servidor donde se ejecutará Atenea.



Gestor de Aplicaciones Web de Tomcat

Mensaje: OK

Gestor

[Listar Aplicaciones](#) | [Ayuda HTML de Gestor](#) | [Ayuda de Gestor](#) | [Estado de Servidor](#)

Aplicaciones					
Trayectoria	Versión	Nombre a Mostrar	Ejecutándose	Sesiones	Comandos
/	Ninguno especificado		true	0	Arrancar Parar Recargar Replegar Expirar sesiones sin trabajar ≥ 30 minutos
/CrawlerConfiguration	Ninguno especificado		true	0	Arrancar Parar Recargar Replegar Expirar sesiones sin trabajar ≥ 30 minutos
/atenea	Ninguno especificado		true	1	Arrancar Parar Recargar Replegar Expirar sesiones sin trabajar ≥ 30 minutos

Imagen 130. Tomcat instalado

Desde aquí podemos subir nuevas aplicaciones, elimina existentes, parar y reiniciarlas, entre otras cosas.

Por defecto, el puerto del servidor Tomcat es el 8080, en este caso, se desea modificar y, para ello, es necesario modificar un parámetro en el fichero de configuración server.xml.

```
GNU nano 2.7.4 Fichero: server.xml

<!-- A "Connector" represents an endpoint by which requests are received
and responses are returned. Documentation at :
Java HTTP Connector: /docs/config/http.html
Java AJP Connector: /docs/config/ajp.html
APR (HTTP/AJP) Connector: /docs/apr.html
Define a non-SSL/TLS HTTP/1.1 Connector on port 8080
-->
<Connector port="80" protocol="HTTP/1.1"
           connectionTimeout="20000"
           redirectPort="443"
           maxPostSize="78643200" />
<!-- A "Connector" using the shared thread pool-->
<!--
<Connector executor="tomcatThreadPool"
           port="8080" protocol="HTTP/1.1"
           connectionTimeout="20000"
           redirectPort="8443" />
```

Imagen 131. Modificando el puerto

Recordar que es necesario reiniciar el servicio, para que los cambios surjan efecto.

10.1.2 Instalar sistema de gestión de bases de datos

La instalación del servidor de MySQL (MySQL s.f.) en Linux es muy simple, ya que con lo siguientes comandos y algún paso más estaría.

```
sudo apt-get update
sudo apt-get -y install mariadb-server mariadb-client
```

Imagen 132. Instalando MySQL

Después de estos comandos, el sistema de gestión de bases de datos estará instalado y, solamente, será necesaria configurarlo. Para ello, se entra en el modo de configuración seguro, mediante la ejecución de **sudo mysql_secure_installation**.

```
mysql -u root -p
```

Imagen 133. Accediendo a MySQL

En este punto, se puede acceder a MySQL sin la necesidad de introducir una contraseña, lo que es algo muy inseguro. Por esto, se va a modificar la contraseña del usuario principal **root**.

```
mysql> update user set password=PASSWORD("root") where user="root";
mysql> flush privileges;
```

10.1.3 Instalar Atenea

Ahora que ya tenemos tanto el servidor de aplicaciones como el sistema de gestión de bases de datos instalados, se va a instalar Atenea para poder utilizarlo.

Desde la administración de Tomcat se suben los dos .war que forman la aplicación Atenea.

Imagen 134. Subiendo WAR

Puede ser que, debido al gran tamaño de estos archivos, el servidor dé un error y no deje desplegar la aplicación. Para solventar esto, es necesario modificar dos ficheros de configuración de Tomcat y reiniciar.

In `conf\server.xml`

```
<Connector port="80" protocol="HTTP/1.1"
  connectionTimeout="20000"
  redirectPort="8443"
  maxPostSize="67589953" />
```

In `webapps\manager\WEB-INF\web.xml`

```
<multipart-config>
  <!-- 52MB max -->
  <max-file-size>52428800</max-file-size>
  <max-request-size>52428800</max-request-size>
  <file-size-threshold>0</file-size-threshold>
</multipart-config>
```

Imagen 135. Aumentando tamaño de subida

Ahora ya se puede subir la aplicación Atenea y su complementario. Como el usuario puede interactuar con Atenea, subiendo y descargando archivos, es necesario dar permisos al usuario Tomcat para que pueda escribir en la carpeta donde se aloja la aplicación.

Finalmente, es necesario configurar la base de datos con la que trabaja Atenea para dar un correcto funcionamiento. Para ello, se ejecuta la orden: **mysql – u root -p < atenea.sql**

Además, será necesario crear dos nuevos usuarios: **Atenea** y **user**, que puedan gestionar esta nueva base de datos **Atenea**. Para ello, se ejecutan las siguiente ordenes dos veces, modificando el usuario.

```
CREATE USER 'golden'@'localhost' IDENTIFIED BY 'password';
GRANT ALL PRIVILEGES ON * . * TO 'golden'@'localhost';
FLUSH PRIVILEGES;
```

Imagen 136. Creando usuarios para Atenea

10.2 Manual de Ejecución

Para ejecutar la aplicación web de Atenea, se introducirá en un navegador web la URL correspondiente al servidor donde se encuentre alojada la aplicación web. En este caso, dicha URL es <http://156.35.94.110/atenea/>.

Son necesarias las siguientes credenciales: **admin** y **password**.

10.3 Manual de Usuario

En este apartado, se describirá detalladamente el funcionamiento de la aplicación Atenea, con el fin de mostrar a los usuarios como deben utilizarla.

10.3.1 Zona pública

10.3.1.1 Análisis público

La zona pública de Atenea permite realizar análisis de la página principal de un enlace dado, para ello, es necesario introducir el enlace y seleccionar tanto una métrica como un crawler.

← → ↻ local:8080/atenea/ ☆ 📄 ⋮

Home | Help | Private | Spanish | English

atenea Atenea: Tool for Quantitative Evaluation of Web Sites

URL:

Metric: -- Please Select -- Crawler: -- Please Select --

Evaluate!

What is Atenea?

Atenea is a web tool for quantitative evaluation of websites. Atenea assigns a numerical value to the level of accessibility/usability of one or more Web sites, through the application of different metrics. Atenea also uses different crawlers to get the links to pages. The metrics used conform with [Web Content Accessibility Guidelines 1.0 \(WCAG 1.0\)](#). An experimental metric conforms with [WCAG 2.0](#) is also available.

In this way, the tool can assist in developing accessible websites, in studies on the status of usability and the behavior of different metrics.

Visit [help](#) section for more information.

©Atenea: Tool for Quantitative Evaluation of Web Sites
Contact | Accessibility | About

Imagen 137. Manual de usuario: Pantalla de inicio

Cuando se haya introducido y seleccionado toda la información se debe presionar el botón proporcionado para evaluar. Mientras el proceso de evaluación se va ejecutando, el sistema muestra un mensaje de espera.

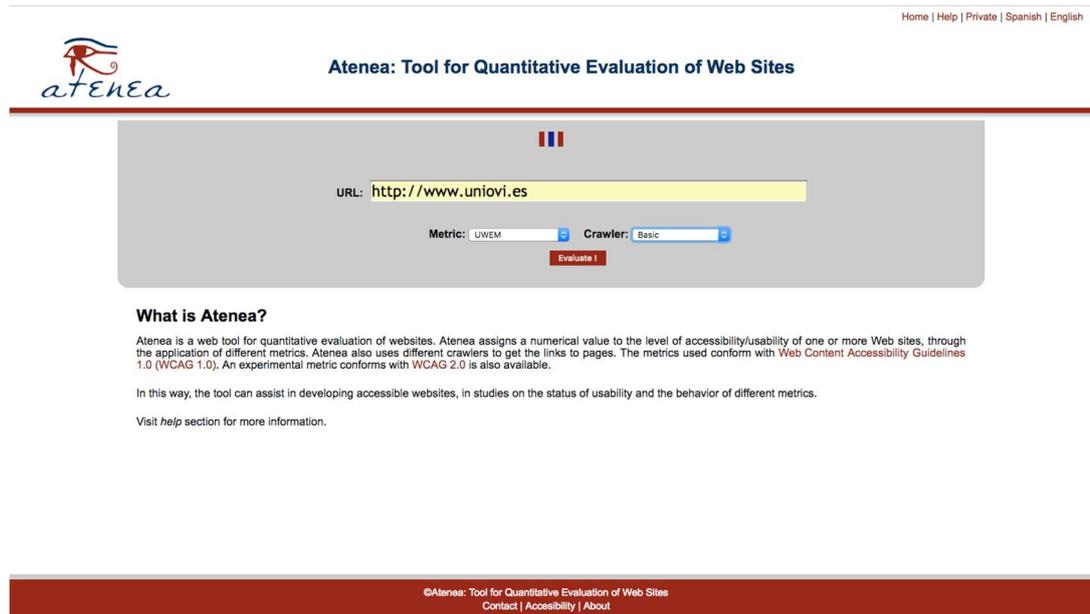


Imagen 138. Manual de usuario: Cargando análisis público

Finalmente, cuando el proceso de análisis termina, se podrá visualizar el resultado obtenido para el enlace introducido.

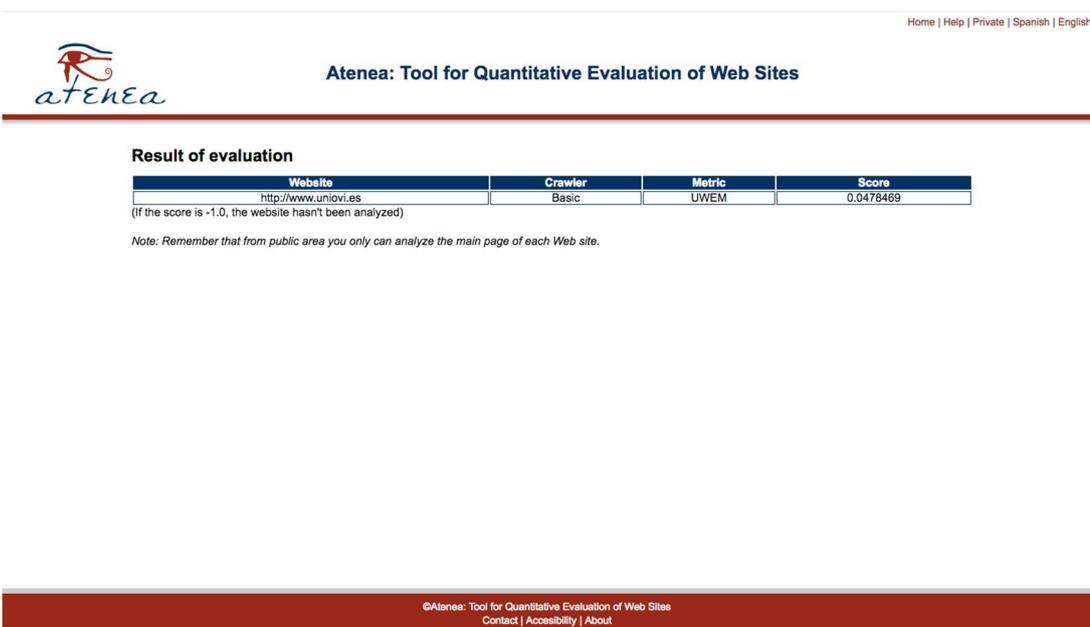


Imagen 139. Manual de usuario: Pantalla de resultados públicos

10.3.1.2 Sección de ayuda

La zona pública de la aplicación cuenta con una sección que ayuda al usuario a utilizar esta misma zona con una serie de instrucciones. Es decir, indica cómo se debe usar la zona pública al igual que este manual. Además, aparecen todos los datos y su descripción tanto de las métricas como de los crawlers para poder utilizarlos correctamente.

**Help**Initial Considerations

All analysis executed from the public area will be conducted with a depth level 0 (main page).

Note: The private area allows analysis up to 3 depth levels and with different metrics simultaneously.

Params of the analysis

- **Uri**
Uri of the web page to evaluate. If you choose evaluate a website, this uri is the main uri of the website. The *http://* can be omitted.
- **Metric**
You can choose a metric from the list of available metrics, below is a list of the available metrics with a short description.
- **Crawler**
At the moment, there are two crawlers to get the links of analyzed sites:
 1. *Basic Crawler*. Own simple web crawler.
 2. *Jobo Crawler*. It's an open source crawler with better performance. Visit [website](#)

Available metrics**Accessibility**

- **WAB**
(*Web Accessibility Barrier*) This metric considers the potential errors presented on the page and the priorities or the impact of each error on navigation of disabled users.
- **WAB ***
This metric is created as a fusion of the most important characteristics of the WAB and UWEM, trying to apply its advantages and avoid disadvantages.

Imagen 140. Manual de usuario: Pantalla de ayuda pública

10.3.1.3 Modificar idioma

En la cabecera de la aplicación aparece un menú, permitiendo que desde cualquier punto de la zona pública se pueda modificar el idioma de la misma.



Imagen 141. Manual de usuario: Modificar idioma

10.3.1.4 Más información

En el pie de la aplicación se pueden observar una serie de opciones que permiten saber más acerca de la propia web de Atenea.

Imagen 142. Manual de usuario: Más información

- Datos necesarios para establecer contacto con el administrador de Atenea.
- Datos del desarrollador de la aplicación.
- Sección donde se indican todas las normas de accesibilidad seguidas para este sitio.

10.3.1.5 Inicio de sesión

Si se cuenta con las credenciales adecuadas, desde esta zona, se puede acceder a la zona privada, mediante un formulario.

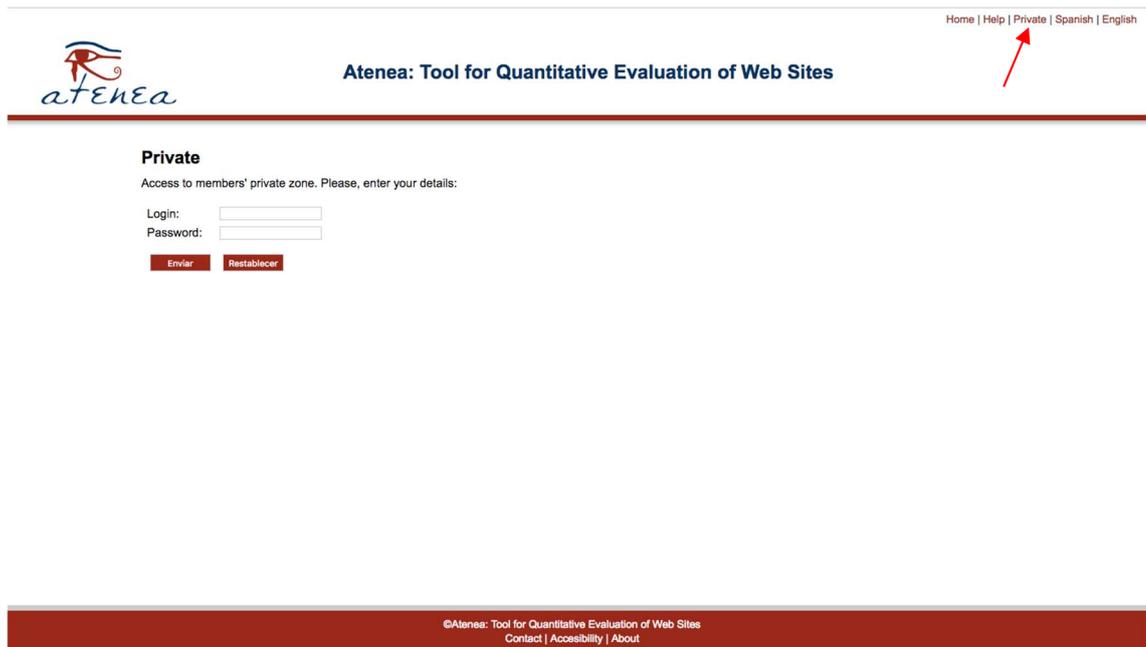


Imagen 143. Manual de usuario: Inicio de sesión

Es necesario introducir el usuario y contraseña correctos y presionar el botón proporcionado para validar esta información. Si las credenciales son incorrectas, el sistema no permitirá acceder a la zona privada.

10.3.2 Zona privada

Desde la página inicial de la zona privada se puede acceder a todas y cada una de las funciones que esta sección proporciona.

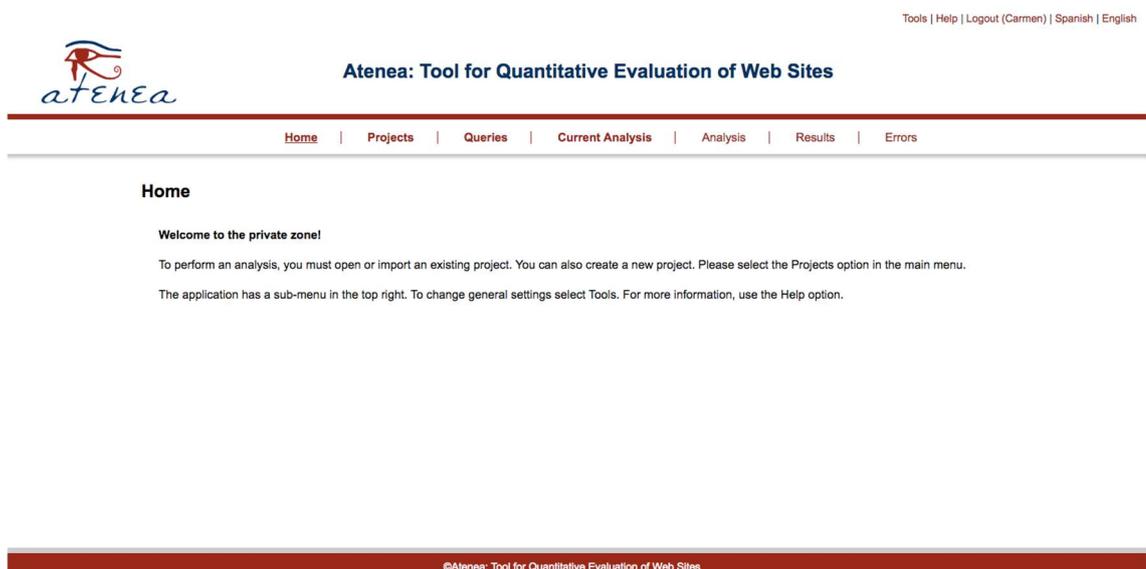


Imagen 144. Manual de usuario: Página de inicio privada

10.3.2.1 Sección proyectos

Desde esta sección se pueden gestionar los proyectos, es decir, proporciona la opción de añadir, abrir, eliminar, importar y exportar proyectos.

10.3.2.1.1 Añadir un proyecto

Para añadir un nuevo proyecto, que permita realizar evaluaciones, es necesario rellenar el siguiente formulario adecuadamente.

Tools | Help | Logout (Carmen) | Spanish | English

Atenea: Tool for Quantitative Evaluation of Web Sites

Home | **Projects** | Queries | Current Analysis | Analysis | Results | Errors

Projects

New Project | Open Project | Delete Project | Import Project | Export Project

Complete the following to create a new project:

Name of project:

Short description:

Metrics to use: EMPHASIZED TEXT IVORY TEXT CLUSTER TOTAL FONTS TOTAL IMAGES TOTAL WORDS TE UWEM WAB WAB20

Depth level:

Maximum pages:

Crawler:

©Atenea: Tool for Quantitative Evaluation of Web Sites

Imagen 145. Manual de usuario: Añadir proyecto

- **Nombre del proyecto:** Nombre identificativo del proyecto.
- **Descripción corta:** Se aconseja introducir una descripción que explique claramente el contenido del proyecto para evitar confusiones.
- **Métricas:** Se escogerán todas las métricas que se quieren utilizar en los análisis asociados al proyecto.
- **Nivel de profundidad:** Nivel de profundidad a utilizar para la evaluación de los sitios web.
- **Número máximo de páginas:** Máximo número de páginas a evaluar dentro de un sitio web.
- **Crawler:** Elemento que obtiene los enlaces para cada sitio web.

En ocasiones algunos crawlers necesitan una configuración especial, por lo que, se muestra una nueva pantalla para poder configurar estas características.



Atenea: Tool for Quantitative Evaluation of Web Sites

Home | **Projects** | Queries | Current Analysis | Analysis | Results | Errors

Projects

New Project | Open Project | Delete Project | Import Project | Export Project

New Project - Step 2
CrawlerConfiguration

Please select the following link to configure the crawler or continue without configuring the crawler:

Jobo Crawler Configuration

Continue without configuration

Imagen 146. Manual de usuario: Pantalla de configuración de crawler

Finalmente, si todo es correcto el proyecto se abrirá automáticamente en el sistema, para poder comenzar a realizar nuevos análisis.



Atenea: Tool for Quantitative Evaluation of Web Sites

Home | **Projects** | Queries | Current Analysis | **Analysis** | Results | Errors

Analysis

Opened Project: a created at 2018-06-10T13:56:49 (Close)



Imagen 147. Manual de usuario: Pantalla proyecto abierto

En todo momento se podrá cerrar el proyecto abierto.

10.3.2.1.2 Abrir, eliminar o exportar proyecto

El funcionamiento de estas tres opciones es muy similar, por lo que, se explicará a la vez y utilizando una única imagen.



Atenea: Tool for Quantitative Evaluation of Web Sites

Home | **Projects** | Queries | Current Analysis | Analysis | Results | Errors

Projects

Add project | **Open Project** | Delete Project | Import Project | Export Project

Show 10 entries

Search:

List projects to open

	Name of project	Short description	Date	Depth level	Maximum pages	Metrics	Crawler
<input type="radio"/>	Universidades	Universidades españolas a analizar	2018-06-18T14:49:25	1	10	EMPHASIZED TEXT,TOTAL IMAGES,TOTAL WORDS,WAB *	Basic
<input type="radio"/>	uno	uno	2018-06-15T12:27:14	0	5	WAB20,	Basic
<input type="radio"/>	a	a	2018-06-10T13:56:49	0	5	EMPHASIZED TEXT,IVORY,TEXT CLUSTER,TOTAL FONTS,TOTAL IMAGES,TOTAL WORDS,TE,UWEM,WAB *,WAB,WAB20,	Basic

Showing 1 to 3 of 3 entries

Previous 1 Next

Open

Imagen 148. Manual de usuario: Pantalla acciones proyectos

El sistema muestra una tabla con la información más relevante de cada uno de los proyectos existentes en el sistema. La primera columna cuenta con unos que permiten seleccionar qué proyecto se desea abrir, eliminar y/o exportar.

En el caso de querer abrir un proyecto nuevo, el sistema solamente permitirá esta acción si no existe un proyecto ya abierto.

Para exportar un proyecto, sí se ha seleccionado el correcto y no ha habido ningún problema, el sistema exportará un fichero XML al ordenador del usuario. El sistema exporta únicamente los parámetros de creación del proyecto y no todos sus análisis.

En cualquiera de las acciones, mostrará un mensaje de error o de verificación, atendiendo a la respuesta producida por el sistema.

Finalmente, cabe destacar que todas las tablas con información sobre proyectos cuentan con un mecanismo, que permite realizar búsquedas, con el fin de mejorar la experiencia de usuario.

10.3.2.1.3 Importar un proyecto

Para importar un proyecto, el sistema pondrá a disposición del usuario un formulario muy sencillo, que abrirá el explorador de archivos del ordenador del cliente para poder seleccionar, exclusivamente, ficheros con formato XML.



Imagen 149. Manual de usuario: Pantalla importar proyecto

10.3.2.2 Sección consultas

Desde esta sección se pueden realizar consultas, directamente, a la base de datos, con el fin de conocer el estado de toda la información persistida en la misma.

Se pueden ejecutar consultas con sintaxis SQL directamente o se pueden guardar algunas consultas predeterminada para utilizarlas posteriormente. Toda consulta guardada puede ser eliminada y se puede volver a crear en cualquier momento.

El mecanismo de ejecutar y guardar una consulta es muy similar, al igual que los mecanismos para abrir o eliminar una consulta. Por esto, se explicará el funcionamiento de una de las acciones de cada pareja comentada.

10.3.2.2.1 Ejecutar o guardar una consulta

El sistema presenta un formulario muy sencillo para poder realizar alguna de estas dos acciones.

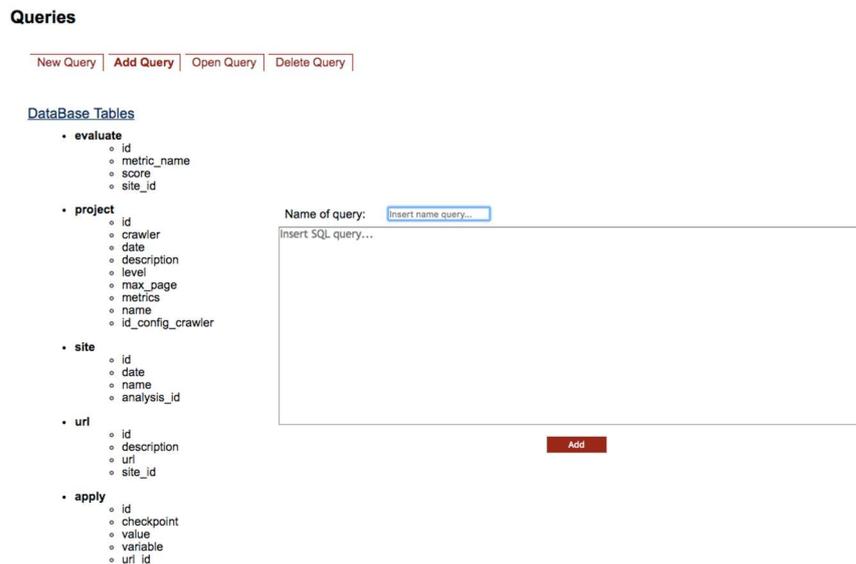


Imagen 150. Manual de usuario: Pantalla guardar consulta

Toda sentencia introducida en el sistema debe mantener una sintaxis SQL correcta. En el caso de almacenar una consulta, el sistema no permitirá añadir consultas repetidas.

10.3.2.2.2 Abrir o eliminar una consulta

Al igual que en la sección de proyecto, el sistema mostrará una tabla con toda la información asociada a cada consulta, así como un botón para seleccionar la consulta a abrir o eliminar.

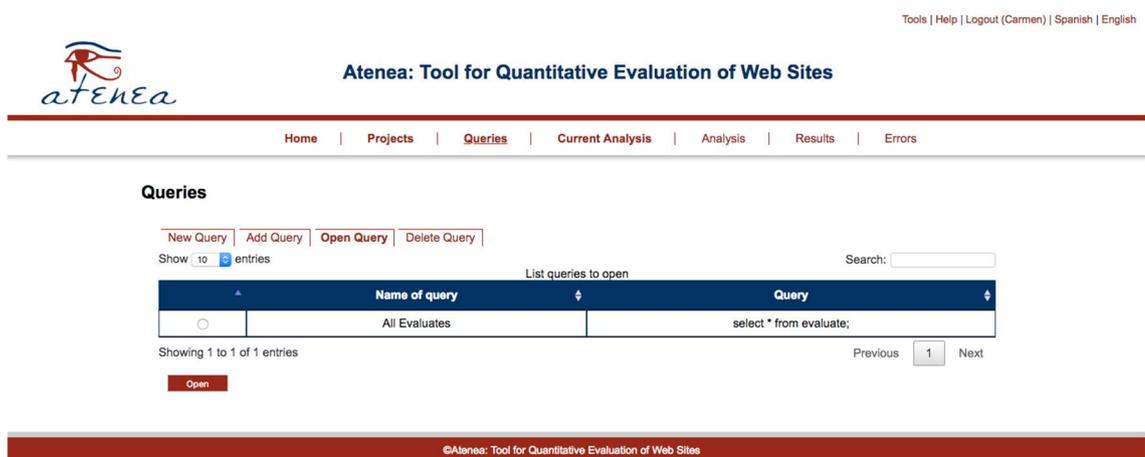


Imagen 151. Manual de usuario: Pantalla abrir consulta

Para accionar el botón es obligatorio seleccionar una consulta, previamente.

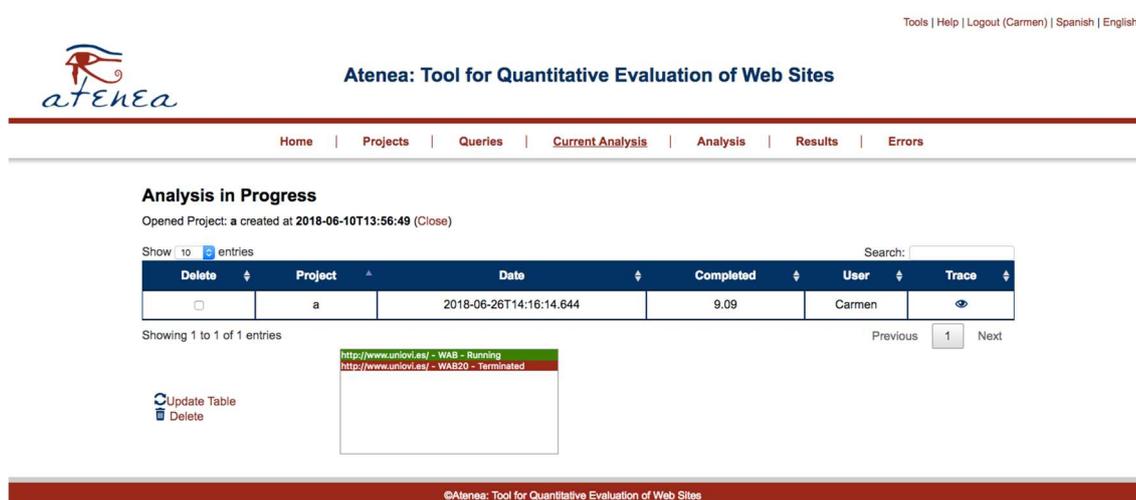
10.3.2.3 Sección análisis actual

Esta sección solamente mostrará información siempre que haya algún análisis en ejecución. Entre esta información, se puede visualizar el nombre del proyecto, la fecha, el porcentaje de progreso, así como el nombre del propietario.

Además, cada análisis tendrá dos botones asociados, uno para poder ver su traza de ejecución y otro para poder seleccionarlo y ser, posteriormente, eliminado.

En la traza de ejecución se puede observar un conjunto de todos los enlaces y métricas que se están ejecutando y a su lado un estado. Este estado puede ser en ejecución o terminado, según su situación.

La pantalla también cuenta con un botón para actualizar toda la tabla y así poder ver los cambios producidos en el porcentaje de progreso.



Tools | Help | Logout (Carmen) | Spanish | English

Atenea: Tool for Quantitative Evaluation of Web Sites

Home | Projects | Queries | **Current Analysis** | Analysis | Results | Errors

Analysis in Progress
Opened Project: a created at 2018-06-10T13:56:49 (Close)

Show 10 entries Search:

Delete	Project	Date	Completed	User	Trace
<input type="checkbox"/>	a	2018-06-26T14:16:14.644	9.09	Carmen	

Showing 1 to 1 of 1 entries Previous 1 Next

Update Table
Delete

http://www.uniovi.es/ - WAB - Running
http://www.uniovi.es/ - WAB20 - Terminated

©Atenea: Tool for Quantitative Evaluation of Web Sites

Imagen 152. Manual de usuario: Pantalla de análisis actual

Los análisis mostrados en esta sección son los análisis de la zona privada que han comenzado desde un fichero, es decir, análisis de un conjunto de URLs que, en principio, son más largos y costosos.

10.3.2.4 Sección de análisis

Esta sección permite realizar dos tipos de análisis: uno dirigido a una única URL y el otro para analizar un conjunto de URLs. Además, se da la opción de visualizar y exportar los resultados para el análisis de un fichero de URLs.

10.3.2.4.1 Analizar una URL

Este formulario es muy similar al de la parte pública, aunque con la nueva opción de escoger el número de hilos utilizado en el análisis.

La métrica y el crawler, así como los demás parámetros necesarios, están asociados al proyecto que debe estar abierto para poder realizar este tipo de análisis.

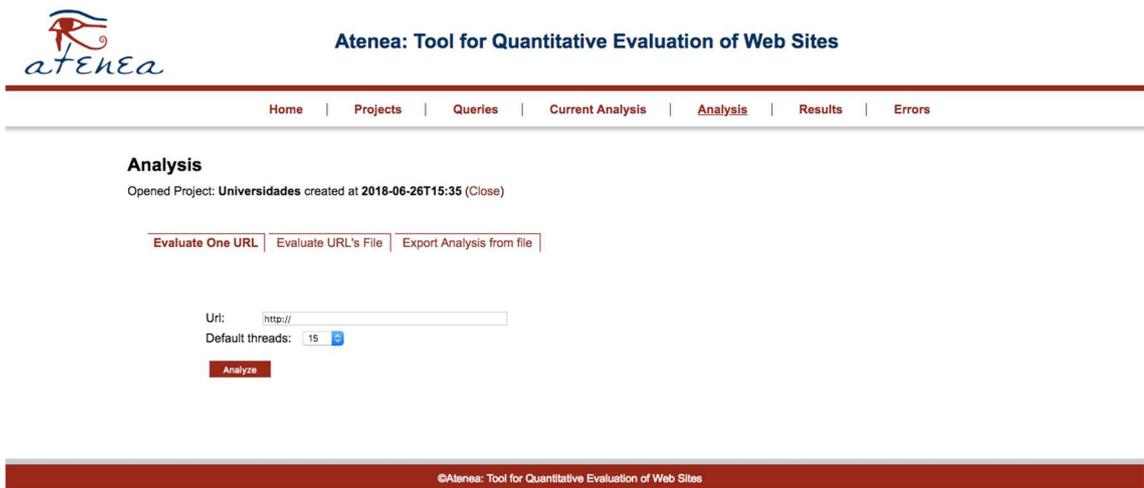


Imagen 153. Manual de usuario: Pantalla analiza una URL

10.3.2.4.2 Analizar un conjunto de URLs

Para analizar un conjunto de URLs, es necesario subir al sistema un fichero de texto plano con un listado de enlaces separados por un retorno de carro.

Hay que recordar que como estos análisis tienen a ser largos y tediosos, por lo que, se pone a disposición del usuario tanto una sección para poder comprobar su estado, como un sistema de notificaciones vía correo electrónico.



Imagen 154. Manual de usuario: Pantalla análisis de URLs

10.3.2.4.3 Exportar análisis

Desde esta sección se puede seleccionar un análisis realizado desde un fichero, con el fin de poder estudiar sus resultados.



Atenea: Tool for Quantitative Evaluation of Web Sites

Home | Projects | Queries | Current Analysis | **Analysis** | Results | Errors

Analysis

Opened Project: **Universidades** created at 2018-06-26T15:35 (Close)

Evaluate One URL | Evaluate URL's File | **Export Analysis from file**

Select the analysis to see the results (one or more):

Show 10 entries

List of analysis URL's file to export

Search:

	Date	URL File (.txt)
<input type="checkbox"/>	2018-06-26T15:35:34	universidades.txt

Showing 1 to 1 of 1 entries

Previous 1 Next

Execute

©Atenea: Tool for Quantitative Evaluation of Web Sites

Imagen 155. Manual de usuario: Pantalla de exportar ficheros

Una vez seleccionado el análisis, se mostrarán los resultados en forma de tabla y aparecerá la siguiente información: Enlace al sitio web, Métrica y Puntuación.

Estos resultados se podrán exportar en formato XML, PDF y XLS.

10.3.2.5 Sección resultados

Cuando un proyecto está abierto, se puede ir a la sección de resultados, para visualizar y estudiar todos los resultados de los análisis tanto desde una única URL como desde un conjunto de URLs. Cabe recordar que si la puntuación de un análisis es -1, quiere decir que este enlace con una métrica determinada no ha podido ser evaluado.

El sistema pone a disposición del usuario un mecanismo para realizar búsquedas, así como realizar filtrados por métrica o enlace. Además, estos resultados se pueden exportar en formato XML, PDF y XLS.

Results

Opened Project: **a** created at 2018-06-10T13:56:49 (Close)

Export XML | Export PDF | Export XLS

Show 10 entries

Search:

WebSite	Metric	score
http://www.uniovi.es/	IVORY	0.643722
http://www.uniovi.es/	EMPHASIZED TEXT	4.47761
http://www.uniovi.es/	TOTAL FONTS	0.0
http://www.uniovi.es/	TEXT CLUSTER	24.0
http://www.uniovi.es/	UWEM	0.0478469
http://www.uniovi.es/	TOTAL IMAGES	370.0
http://www.uniovi.es/	TE	0.428571
http://www.uniovi.es/	WAB *	9.04217
http://www.uniovi.es/	TOTAL WORDS	657.0
http://www.uniovi.es/	WAB	4.04217

Showing 1 to 10 of 11 entries
(If the score is -1.0, the website hasn't been analyzed)

Previous 1 2 Next

©Atenea: Tool for Quantitative Evaluation of Web Sites

Imagen 156. Manual de usuario: Pantalla de resultados

10.3.2.5.1 Sección de errores

Durante los análisis, el sistema además de almacenar los resultados almacena todos los errores asociados a las métricas y sus checkpoints.

Para visualizar estos errores, el sistema obliga a seleccionar una métrica y un conjunto de enlaces asociados a un análisis.

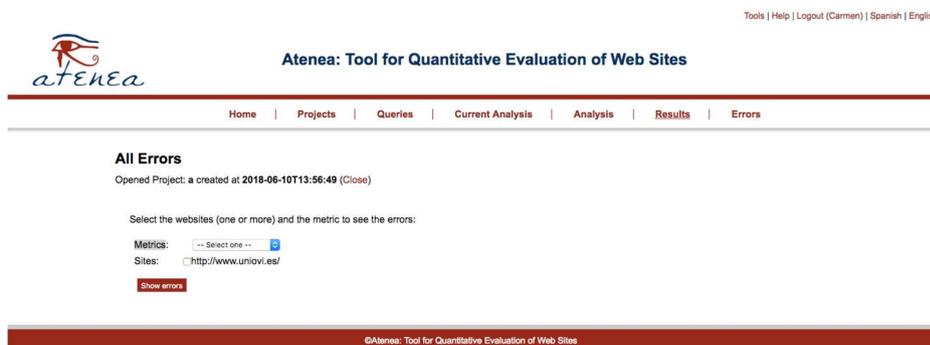


Imagen 157. Manual de usuario: Pantalla de errores

Una vez seleccionados estos parámetros, el sistema mostrará sus correspondientes errores. La forma de mostrar estos errores depende de si la métrica implementa un informe personalizado.

10.3.2.6 Sección de herramientas

En la parte superior derecha de todas las pantallas de la zona privada aparece un menú y una de las opciones sirve para la configuración del sistema.

Desde aquí se pueden realizar diferentes funciones, como son:

- Añadir métricas y/o crawlers
- Añadir o eliminar usuarios del sistema.
- Actualizar la configuración básica de la aplicación.
- Configurar la frecuencia con la que se desea recibir notificaciones o no.

10.3.2.6.1 Configuración básica

Los parámetros aquí modificados serán los de por defecto a la hora de crear proyectos y realizar evaluaciones.



Imagen 158. Manual de usuario: Configuración básica

10.3.2.6.2 Añadir métricas o crawlers

El sistema permite añadir, en tiempo de ejecución, nuevos ficheros JAR externos para medir aspectos de accesibilidad y usabilidad o para añadir nuevas formas de obtener los enlaces asociados a una página dada.

Imagen 159. Manual de usuario: Pantalla de adición de métricas

El sistema muestra un formulario con un campo que abre el sistema de archivos del usuario, para que este, pueda escoger un fichero con un formato JAR.

10.3.2.6.3 Añadir o eliminar un usuario

Característica que permite añadir o eliminar un nuevo usuario en el sistema.

Para añadir un nuevo usuario será necesario introducir el nombre de usuario, su correo electrónico y la contraseña del mismo.

Imagen 160. Manual de usuario: Pantalla de añadir un usuario

El propio sistema indicará que campos son estrictamente obligatorios.

Para eliminar un usuario, debe seleccionarse el usuario en cuestión de entre una lista con todos los usuarios del sistema, a excepción del usuario en sesión y el propio administrador.

10.3.2.6.4 Configuración de notificaciones

Como ya se ha mencionado, cuando el usuario ejecuta un análisis de un conjunto de URLs desde un fichero de texto plano, esta evaluación tiende a ser larga y tediosa. Por esto, se ha desarrollado un sistema que permite notificar el estado de dichos análisis.

Se ha creado una nueva pantalla para seleccionar si se desea o no recibir correos electrónicos y, en caso afirmativo, se puede actualizar la frecuencia o periodicidad.

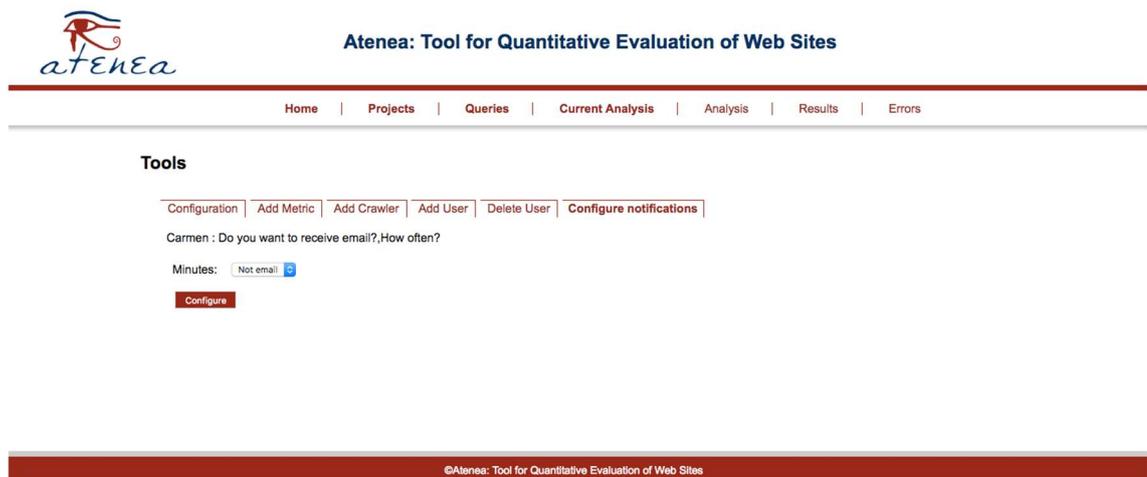


Imagen 161. Manual de usuario: Pantalla de configuración de notificaciones

Además de recibir mensaje cada X tiempo con el estado y porcentaje de los análisis, se recibirá un mensaje inicial y otro final.

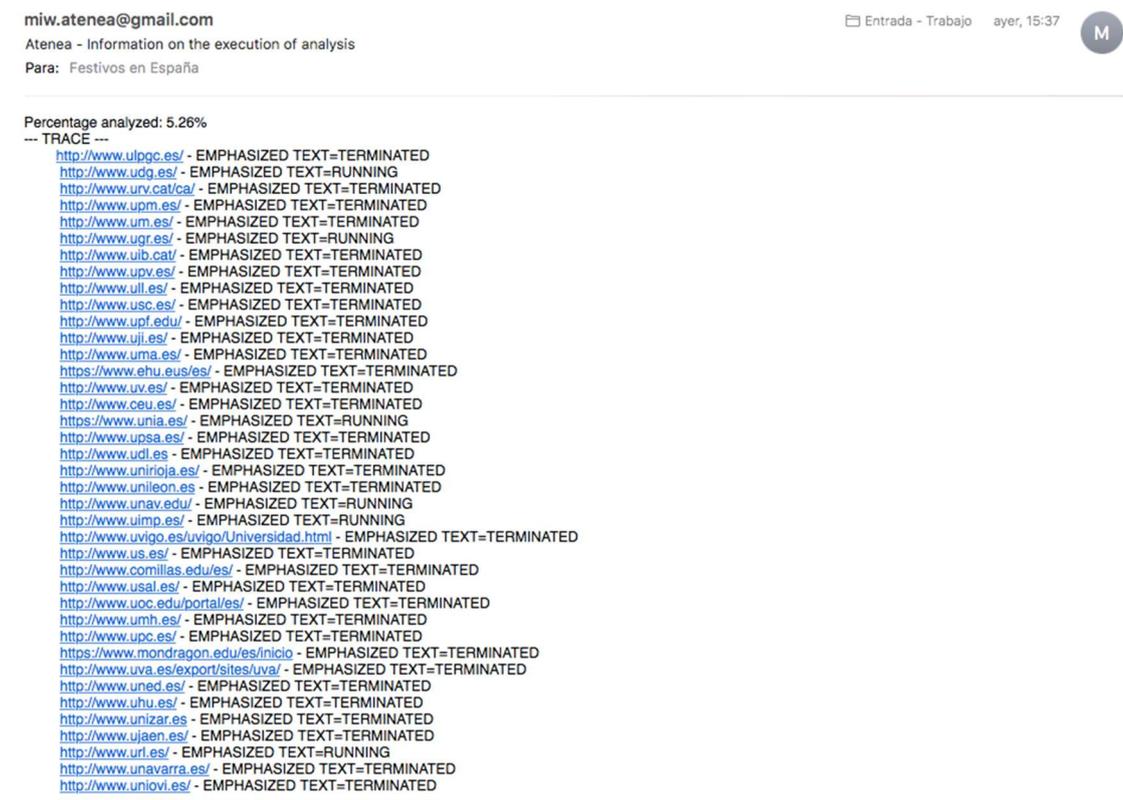


Imagen 162. Manual de usuario: Aspecto de un correo

10.3.2.7 Otras funciones

En el menú superior derecho de la zona privada de la aplicación, además de la sección de herramienta, se pueden encontrar varias funciones más, como son: modificar el idioma, cerrar sesión y consultar la ayuda de esta misma zona.



Atenea: Tool for Quantitative Evaluation of Web Sites

Tools | Help | Logout (Carmen) | Spanish | English

Imagen 163. Manual de usuario: Menú superior derecho

Para salir de la aplicación, el usuario deberá presión el botón dispuoto para esta acción y el sistema volver a la zona pública de la aplicación. El usuario puede volver a la zona privada siempre que lo desee introduciendo sus credenciales.

El sistema permite modificar el idioma de español a inglés y viceversa.

Finamente, comentar que existe una sección de ayuda donde aparece información similar a la de sete manual, es decir, enseña a utilizar todas las funciones de la parte privada.

10.4 Manual del Programador

10.4.1 Manual para añadir nuevas métricas

Una de las principales características de Atenea es que tiene la capacidad de añadir fácilmente nuevas métricas. Es precisamente esta característica la que permite la realización de múltiples ampliaciones en forma de métricas. Los pasos para implementar una métrica son los siguientes:

- Implementar los checkpoints.
- Implementar la métrica.
- Implementar el informe de la métrica.
- Crear el archivo .jar que los incluya.

10.4.1.1 Implementar los checkpoints

Un checkpoint mide una característica de una página web. Para implementar checkpoints para esta herramienta se deben de seguir una serie de normas:

- Tiene que pertenecer al paquete (NOMBRE_NORMATIVA).checkpoints
- Cada checkpoint debe heredar de la clase *CheckpointModel* que implemente la interfaz *ICheckpoint*. El método *toMeasure* es en el que realiza la medición así que tiene que ser sobrescrito por cada checkpoint único.
- Por motivos de almacenamiento se recomienda que cada checkpoint se llame *Checkpointx* donde “x” es un número. Existen 108 checkpoints implementados actualmente por lo que tienen que ser números mayores que 108 o podría haber problemas a la hora de almacenarlos la aplicación.
- Se debe inicializar el atributo “description” con la descripción de la función que realiza el checkpoint.
- Se le debe asignar el nombre al atributo “name”. Este nombre tiene que coincidir con el nombre de la clase.
- Se debe llamar al constructor de *CheckpointModel* para pasarle el atributo “normative” con el nombre de la normativa a la que pertenece el checkpoint.

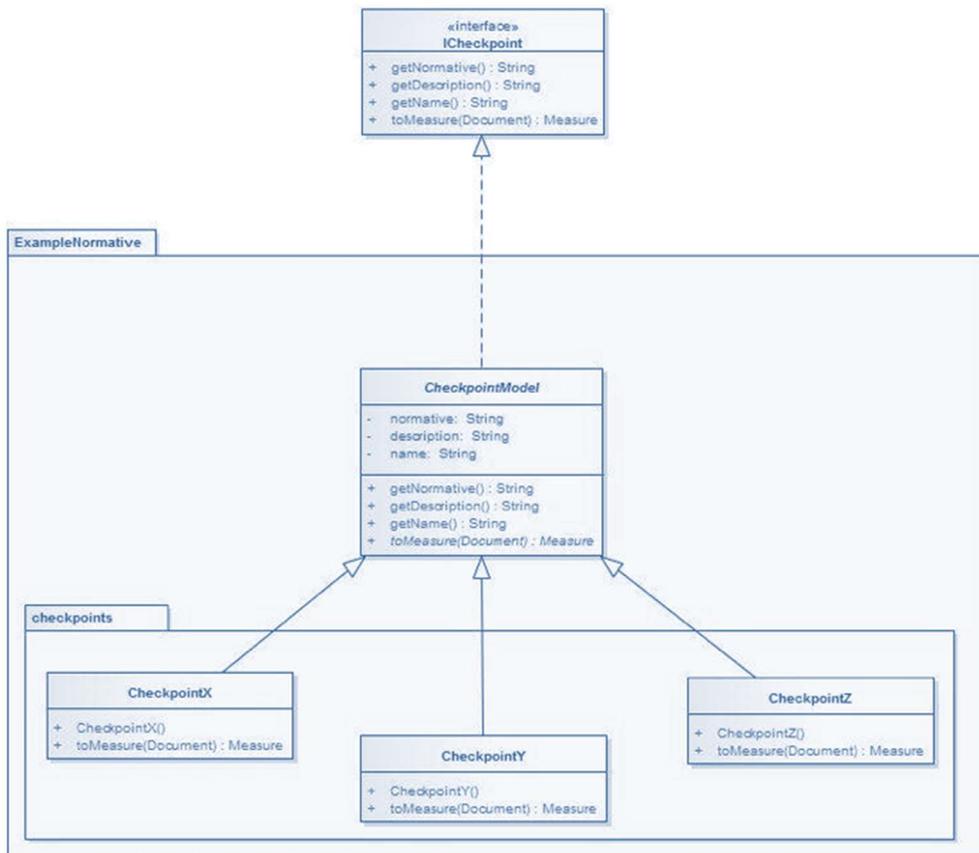


Imagen 164. Diagrama de clases Checkpoints

10.4.1.2 Implementar la métrica

Cada métrica se encarga de calcular el resultado de la medición de una página web. Al implementar una métrica para esta herramienta se deben de seguir las siguientes normas:

- Debe pertenecer al paquete (NOMBRE_NORMATIVA).metrics
- Heredar de la clase *MetricModel*, que implemente la interfaz *IMetric* y rescribir el método *calculate*.
- Su nombre debe de ser *MetricX*, donde X es el nombre concreto de la métrica. Por ejemplo: *MetricWAB*
- Cada métrica deberá tener un *Collection* que contendrá los nombres de los checkpoints pertenecientes a ella.
- Además, se deberá crear un array de *String* que va a contener los puntos a medir una vez que se inicie un análisis.
- Se debe asignar un nombre al atributo *name*.
- Se debe llamar al constructor de *MetricModel* para pasarle el atributo “*normative*” con el nombre de la normativa a la que pertenece esta métrica.

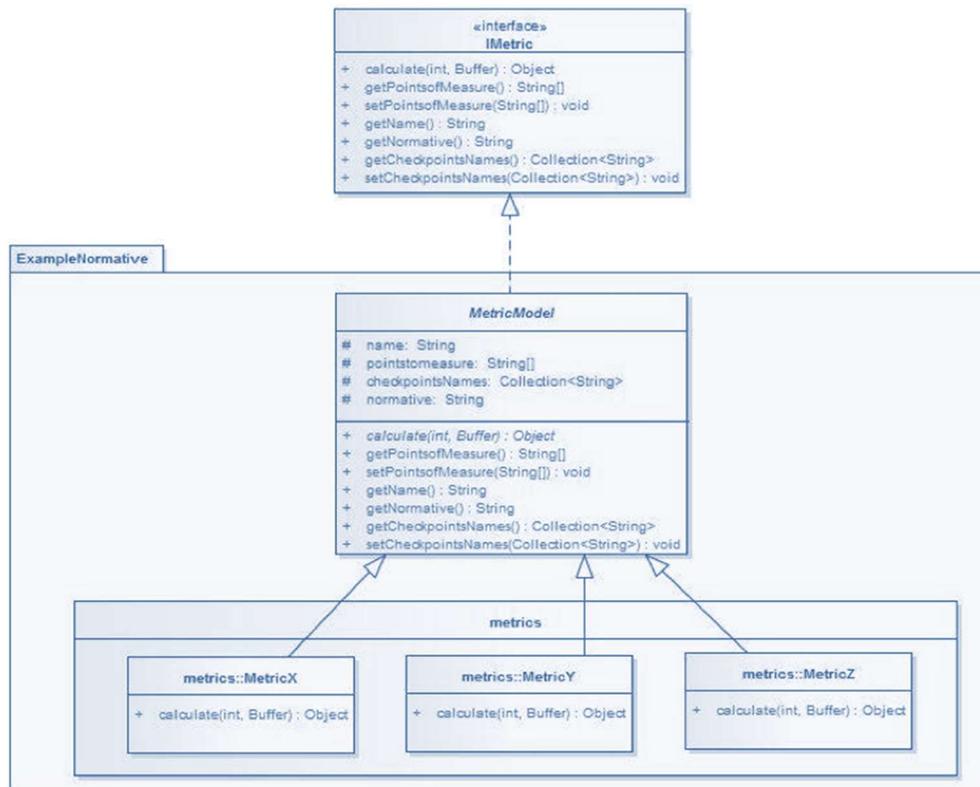


Imagen 165. Diagrama de clases Métrica

10.4.1.3 Implementar el informe de errores

Si no se implementa ningún informe, la aplicación creará un informe genérico que mostrará todos los errores que han ocurrido durante el análisis de una métrica.

Si se quiere implementar, un informe consta de dos partes, una será la interfaz y la otra la clase que ejecuta la lógica necesaria para mostrar los informes por pantalla, que llamaremos report.

10.4.1.3.1 Interfaz

Cada interfaz deberá heredar de la clase WebInterfaceModel y deberá seguir las siguientes normas:

- Pertenecer al paquete (NOMBRE_NORMATIVA).reports
- Implementar el método activateErrors(). Va a ser el método que la ventana principal de la aplicación llamará cuando inicialice la interfaz y a la que le pasará la clase que implementará la lógica, por lo que es recomendable que en este método se inicialicen las variables necesarias para que funcione el informe completo.
- Implementar el método getMetric(), que devolverá la normativa a la que pertenece la interfaz.

10.4.1.3.2 Report

El Report implementará la lógica de la interfaz (con lo cual se trabajará con la misma versión para la herramienta de escritorio y la aplicación web), hará uso de una clase que será una versión capada de *MainService* y ofrecerá al report unos métodos para acceder a los errores, resultados, etc... necesarios para mostrar en pantalla la información necesaria. Cada report deberá:

- Pertenecer al paquete (NOMBRE_NORMATIVA).reports
- Implementar la interfaz *IReport*, así como sus métodos `getNormative()` y `setReportService(ReportService rs)`.
- Poseer un atributo *ReportService* para poder acceder a la información necesaria.

10.4.1.4 Crear el archivo .jar

Una vez implementados los checkpoints y la métrica hay que introducirlos en un archivo .jar para que el sistema pueda utilizarlos. Para ello se ha de hacer lo siguiente:

- Se compilan todas las clases para obtener los .class.
- Todos los .class deberán estar distribuidos en las siguientes carpetas:
 - Checkpoints: (NOMBRE_METRICA).checkpoints
 - Métricas: (NOMBRE_METRICA).metrics
 - Informes e interfaces: (NOMBRE_METRICA).reports
- Se crea el .jar introduciendo la siguiente orden en línea de comandos:

```
jar -cf nombre.jar (NOMBRE_METRICA)/
```

- Es necesario comprobar que el archivo .jar contenga, además de la carpeta con la meta-información, todos los .class dentro de los directorios correspondientes directorio (NOMBRE_METRICA) /
- Añadir la métrica al sistema como se indica en los diferentes manuales de las aplicaciones.

10.4.2 Manual para añadir nuevos crawlers

Con el fin de mejorar la incorporación de nuevos crawlers a la aplicación Atenea implementa un método sencillo, pero eficaz para añadir nuevos crawlers fácilmente. Para ello es preciso:

- Implementar un adaptador
- Crear e incorporar un archivo .jar

10.4.2.1 Implementar un adaptador

Para poder incorporar el nuevo crawler a la aplicación es necesario implementar una clase o módulo Java que adapte el crawler a la aplicación. A continuación, se describirán los pasos necesarios para crearlo, así como las condiciones que debe cumplir:

- El adaptador debe estar contenido en un paquete con el nombre syntactic_analysis.crawlers.(NOMBRE_CRAWLER)Adapter
- La estructura que debe cumplir el adaptador que se implemente es la siguiente:

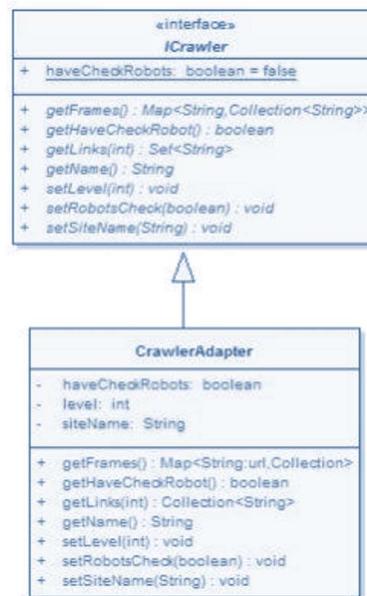


Imagen 166. Diagrama de clases Adaptador

- A continuación, se describirá la función de cada uno de los métodos y atributos que deben implementarse:

CrawlerAdapter	
Atributos	
haveChecksRobots: boolean	Booleano que determine si se implementan o no las comprobaciones al archivo “robots.txt”
level: int	Entero para representar el nivel de profundidad al que debe realizarse la búsqueda.
siteName: String	String que representa la URL base a partir de la cual debe iniciarse la búsqueda.
Métodos	
getFrames(): Map<String:url, Collection>	Método que debe retornar una colección con los frames obtenidos
getHaveCheckRobots(): Boolean	Método que debe retornar un booleano indicando si el crawler implementado hace comprobaciones o no al archivo “robots.txt”
getLinks(int): Collection<String>	Método que debe retornar una colección con los links obtenidos
getName(): String	Retorna el nombre asignado al crawler que se implementa

setLevel(int): void	Método que sirve para modificar el valor del atributo level
setRobotsCheck(boolean): void	Método que sirve para indicar al crawler implementado si debe hacer comprobaciones o no del archivo "robots.txt"
setSiteName(String): void	Método que sirve para modificar el valor del atributo siteName

Imagen 167. Métodos y atributos de crawleradapter

- El nombre de la clase que implemente la interfaz debe ir seguido de "Adapter". Por ejemplo: "JoboAdapter", "JSpiderAdapter".
- El adaptador no debe disponer de constructor o si lo tiene no debería tener parámetros.

10.4.2.2 Creación e incorporación del archivo .jar

Una vez creado el adaptador, el resto de las clases a utilizar deben ser empaquetadas en un fichero .jar e incorporarlo al sistema para que éste pueda hacer uso de él. Para ello se siguen los siguientes pasos:

- Se compilan todos los archivos .java para obtener los .class necesarios.
- Se introducen los archivos .class en un directorio syntactic_analysis/crawlers/(NOMBRE_CRAWLER)Adapter y se crea el .jar a través de línea de comandos:

```
jar -cf nombre.jar syntactic_analysis/crawlers/(NOMBRE_CRAWLER)Adapter
```

- Finalmente, se introduce el fichero en la herramienta de alguna de las formas especificadas en los diferentes manuales.

10.4.3 Configuración de un crawler

No ha sido posible integrar la interfaz de configuración del crawler para la herramienta web dentro de un fichero jar. No se conoce una solución para incluir elementos de presentación (JSP) dentro del plugin (jar) que contiene el crawler.

De este modo, la solución adoptada ha sido utilizar otra aplicación web diferente a la principal. Cuando la aplicación sea web no se cargará la configuración del crawler como un plugin, sino como otra aplicación web que hace esos trabajos con anterioridad.

Lo que se hace finalmente, es introducir en la aplicación principal un enlace a la nueva aplicación, para que pueda llamarse a la configuración y todo parezca interconectado.

Capítulo 11. Conclusiones y Ampliaciones

11.1 Conclusiones

Este proyecto estaba enfocado a la realización de una reingeniería y migración de un proyecto existente, dedicado al estudio cuantitativo tanto de la accesibilidad como de la usabilidad de sitios web. Cuando se inició el desarrollo del presente proyecto, se planteaban una serie de objetivos, atendiendo a la funcionalidad, como son:

- Dar independencia a la aplicación Atenea del sistema operativo que la sirve.
- Estudiar y analizar el sistema, con el fin de detectar posibles deficiencias y solventarlas.
- Actualizar el sistema, ayudándose de las nuevas tecnologías desarrolladas durante estos años.
- Reducir los tiempos de respuesta de la aplicación.
- Añadir nuevas funcionalidades, entre las que destaca la creación de un nuevo sistema de notificaciones, con el fin de adaptarse a las nuevas exigencias de la sociedad.

Además, era necesario conseguir estos objetivos sin modificar la esencia de la aplicación, es decir, la aplicación siempre debe ir a mejor y nunca perder de vista su funcionalidad básica, su correcto diseño o la línea base de arquitectura sobre la que está cimentada.

Otro objetivo desembocaba en la necesidad de aunar, mejorar y actualizar la documentación de ambos proyectos en un único documento, con el fin de facilitar sus posteriores modificaciones o adiciones, así como su estudio.

Tras finalizar el proceso de implementación, se puede concluir que se han cumplido todos estos objetivos.

Cabe mencionar que, a nivel personal, también puedo concluir, que esta experiencia me ha ayudado a poner en práctica muchos de los conocimientos adquiridos durante mi carrera profesional, así como a mejorarlos.

Finalmente, hay que destacar que era la primera vez que acometía un proceso de reingeniería y trabajaba con un código ajeno completamente a mí, además, nunca me había enfrentado a un proyecto que, finalmente, se implantará para su utilización. Por esto, al principio, la presión de la responsabilidad era enorme, pero a medida que trabajaba en dicho proyecto esta sensación cambió. He podido concluir que un buen desarrollador no solamente trabaja con su código o en solitario, además de la importancia de contar con una buena documentación base.

11.2 Ampliaciones

A continuación, se detallan un conjunto de sugerencias que, en un futuro, sería interesante implementar para que Atenea cuente con nuevas o mejoradas funciones.

11.2.1 Crawlers con interfaz de usuario

Uno de los grandes problemas que ha surgido dentro de este proyecto, ha sido la imposibilidad de introducir elementos JSP dentro de los nuevos crawlers, para su correcta configuración.

Actualmente, en la aplicación web de Atenea, es otra aplicación secundaria la que se encarga de realizar la configuración de los crawlers.

Este sería un punto interesante de investigación. Estudiar la posibilidad de introducir elementos JSP en un plugin, de manera que más tarde pudieran ser cargados en tiempo de ejecución.

11.2.2 Reutilización de hilos

Una ampliación interesante sería la posibilidad de reutilizar hilos, ya que la tecnología Java y los Thread que se implementan no permiten que estos hilos se vuelvan a utilizar. Esto es, cuando un proceso termina, no puede ser usado de nuevo.

11.2.3 Prioridades entre análisis en curso

Es posible que en algún momento existan varios análisis en ejecución y el usuario puede necesitar los resultados de uno de ellos con mayor urgencia, por lo que, se sugiere añadir la funcionalidad de poder dar prioridad a los análisis en ejecución.

11.2.4 Opción para analizar un sitio web completo

Hasta el momento, no existe una opción para analizar todas las páginas de un sitio web. Por lo tanto, sería interesante poder añadir una opción para analizar, automáticamente y de forma sencilla, todas las páginas de un sitio web.

11.2.5 Creación de una app de apoyo

Sería también muy interesante tener una *app* que pudiera interactuar con la aplicación web para mantener informado al usuario de la marcha de los análisis. Esta *app* podría, a su vez, tener funcionalidad para lanzar nuevos análisis, pararlos, ponerles prioridad, etc.

11.2.6 Gestión de usuarios avanzada

Otra ampliación de gran interés consistiría en la mejora de gestión de usuarios para permitir a terceras partes el uso de la zona privada, pero con menos privilegios que los usuarios privados actuales, de manera que se pudiera permitir a otros investigadores comprobar los resultados ofertados por la herramienta.

Capítulo 12. Bibliografía

12.1 Libros y Artículos

Association, IEEE-Standards. 2000. *1471-2000 - IEEE Recommended Practice for Architectural Description for Software-Intensive Systems*. IEE.

Gamma, Erich, Richard Helm, Ralph Johnson, y John Vlissides. 2009. *Patrones de Diseño*. Addison Wesley.

Ivory, Melody Y., Rashmi R. Sinha, y Marti A. Hearst. 2001. *Empirically Validated Web Page Design Metrics*.

Slatin, John, y Sharron Rush. 2003. *Maximum accesibility: making your web site more usable for everyone*.

Sommerville, Ian. 2011. *Ingeniería de Software*. Novena edición. Estado de México: Addison Wesley.

12.2 Referencias en internet

s.f. *AChecker*. <https://achecker.ca/checker/index.php>.

s.f. *Apache Tomcat*. <http://tomcat.apache.org/>.

s.f. *Colour Contrast Analyzer*.
<https://developer.paciellogroup.com/resources/contrastanalyser/>.

s.f. *MySQL*. <https://www.mysql.com/>.

s.f. *Spring*. <https://spring.io/>.

s.f. *TAW*. <https://www.tawdis.net/index>.

s.f. *WAVE*. <https://wave.webaim.org/>.

s.f. *Web Accessibility Initiative*. <https://www.w3.org/WAI/>.

s.f. *Web Content Accessibility Guidelines*. <https://www.w3.org/WAI/standards-guidelines/wcag/>.

s.f. *World Wide Web Consortium*. <https://www.w3c.es/>.

12.3 Otros trabajos fin de máster

- *Atenea 3.0: Reingeniería del Sistema para la evaluación cuantitativa de Sitios Web*
Proyecto Fin de Máster de la Escuela de Ingeniería Informática en la Universidad de Oviedo (2012)
Javier Oyón Fernández

Capítulo 13. Apéndices

13.1 Glosario y Diccionario de Datos

A continuación, se explica de forma breve y concisa el significado de los principales términos usados en el presente proyecto:

- **Accesibilidad:** Grado en el que todas las personas pueden acceder a un servicio independientemente de sus capacidades.
- **Atenea:** Sistema para la evaluación cuantitativa de la accesibilidad de sitios Web.
- **Base de datos:** Modelos de persistencia de datos. Conjunto de datos ordenados y estructurados para permitir su consulta.
- **Componente:** Módulo integrante de una arquitectura software que tiene gran cohesión interna y ofrece servicios a otros componentes través de conectores.
- **Crawler (Araña web):** programa que inspecciona las páginas Word Wide Web de forma metódica y automatizada.
- **Despliegue:** Representación de la disposición de los elementos de software en los de hardware durante la ejecución.
- **Educción:** Sacar una cosa de otra, deducir.
- **Framework:** Entorno o ambiente de trabajo para desarrollo, dependiendo del lenguaje, normalmente integra componentes que facilitan el desarrollo de aplicaciones como el soporte de programa, bibliotecas, plantillas y más.
- **Interfaz:** Abstracción que define el modelo de interacción de un elemento de la arquitectura con el resto de los elementos.
- **Métrica cuantitativa:** Medición que da como resultado un valor numérico.
- **Migración:** proceso por el cual los datos, ya sean de un servidor o un sistema de gestión de base de datos, son movidos a otro servidor o sistema. Los motivos pueden ser muy diversos.
- **Paquete:** Elemento contenedor para el empaquetado de una parte del código de un sistema.
- **Patrón arquitectónico:** Identificación de abstracciones que se repiten en las arquitecturas ante determinados requisitos del dominio. Son elementos de buenas prácticas. Solución general, reutilizable a un problema que aparece de forma recurrente en un determinado contexto dentro de la arquitectura del software.
- **Reingeniería:** Modificación de un producto software, o de ciertos componentes, usando para el análisis del sistema existente técnicas de Ingeniería Inversa y, para la

etapa de reconstrucción, herramientas de Ingeniería Directa, de tal manera que se oriente este cambio hacia mayores niveles de facilidad en cuanto a mantenimiento, reutilización, comprensión o evaluación.

- **Requisito:** Circunstancia o condición necesaria para un sistema.
- **Sistema:** Elemento construido en base a unos requisitos sobre un dominio y que automatiza los procesos relacionados con dichos requisitos.
- **STS:** Programa complementario al entorno de desarrollo, con el fin de facilitar la creación de aplicación web, utilizando el framework Spring.
- **Transacción:** Una transacción es una interacción con una estructura de datos compleja, compuesta por varios procesos que se han de aplicar uno después del otro.
- **URL (Uniform Resource Locator):** Dirección única que identifica una página web en Internet.
- **Usabilidad:** Facilidad con la que las personas pueden utilizar una herramienta.
- **WCAG:** Pautas de Accesibilidad del Contenido en la Web. Explican cómo hacer que el contenido en la Web sea accesible para personas con discapacidad.