



Universidad de Oviedo
Universidá d'Uviéu
University of Oviedo



Escuela de
Ingeniería
Informática
Universidad de Oviedo



Trabajo de
Desarrollo

APLICACIÓN DE GESTIÓN LABORAL PARA EMPLEADOS DE RENFE CERCANÍAS ASTURIAS Y VISUALIZACIÓN DE INFORMACIÓN PARA PASAJEROS

**GRADO EN INGENIERÍA INFORMÁTICA DEL
SOFTWARE**

TRABAJO DE FIN DE GRADO

AUTOR

Raquel Santos Fernández

TUTOR

Miguel Sánchez Santillán

Julio 2022

Versión 1.95 (15/07/2020)

Modelo Métrica 3



Este documento ha sido creado basándose en la plantilla elaborada por JOSÉ MANUEL REDONDO LÓPEZ. [1]



Agradecimientos

En primer lugar, quiero agradecer a mi tutor Miguel por estar siempre disponible, por ayudarme en todo lo necesario y por hacer que intente dar lo mejor de mí.

A nivel académico y profesional quiero agradecer a todos aquellos profesores que han hecho del aprendizaje algo divertido y ameno, que nos han inculcado su interés en la materia y que se han preocupado por nosotros y de que realmente aprendiésemos.

A nivel personal quiero agradecer a mis padres y a mi hermano por ser el pilar fundamental de mi vida ya que al final son ellos los que han estado a mi lado, han sufrido por mí y se han alegrado conmigo en todos los momentos por los que he pasado a lo largo de la realización de este trabajo y a lo largo de toda mi carrera universitaria. Gracias por aguantarme incluso cuando ni yo misma lo hago.

Gracias a todas los integrantes del grupo Hingüeniens, creo sin vosotros no podría haber llegado hasta donde estoy ahora mismo. Gracias por hacer más amenas las mañanas, las tardes, los desayunos, comidas en la uni, y podría decirse que incluso las noches. Gracias también a aquellos alumnos que se han molestado en explicarme cualquier duda, que me han dejado algún apunte y con los que he compartido mesa en clase.

Quiero hacer una mención especial a María, la persona que más me ha ayudado sin ninguna duda en este proceso final, por ayudarme a creer en mí misma, por estar disponible para escucharme las 24 horas del día, por animarme incluso cuando todo parecía estar perdido, en definitiva, gracias por ser el gran descubrimiento que me llevo de mi etapa en la Universidad.

En último lugar, me gustaría agradecer a mis amigas y amigos. Chicas del Equipo de investigación, gracias por confiar en mi mucho más de lo que lo hago yo y por darme siempre ese empujón para continuar. Carmen y María, gracias por las reuniones de los domingos desde que tengo uso de razón. Champ, gracias por entenderme sin siquiera hablar y por ser la mejor amiga que se puede tener. Chicos, en este último año me habéis ayudado mucho más de lo que os podéis imaginar, gracias por hacerme desconectar de todo siempre que lo necesitaba.

Me siento muy afortunada de teneros a todos en mi vida. Gracias.



RESUMEN

El objetivo de este trabajo es diseñar e implementar una aplicación web destinada principalmente a los empleados de Renfe Cercanías Asturias para consultar las rutas que se deben realizar en la jornada laboral, visualizar en qué andén se encuentra el tren que deben conducir y a qué hora sale.

Además, los empleados pueden señalar las averías, eventos anómalos del tren o información relevante para que los siguientes trabajadores tengan constancia.

Por otra parte, la aplicación permite a los empleados seleccionar un período de vacaciones entre los disponibles, que posteriormente será validado o no por la persona encargada. Visualizar el calendario mensual con las jornadas y sus tareas, solicitar días libres e interactuar en un portal de solicitud de cambios donde un empleado puede enviar una petición de cambio de jornada y cualquier compañero la puede aceptar si está interesado y su jornada puede ser intercambiada.

Por último, la aplicación contendrá un apartado destinado a los pasajeros de la empresa en el cual pueden visualizar diferente información.

La aplicación no ha sido creada por encargo de la empresa Renfe pero propone una solución alternativa al mecanismo usado actualmente el cuál se puede ver descrito avanzado el documento.

Palabras clave: Aplicación Web, gestión empleados, jornada laboral, consultar horarios, empleado Renfe, pasajero, tren.



Índice de contenido

Resumen.....	3
Capítulo 1 Planificación del Sistema de Información	15
1.1 PSI 1: Inicio del Plan de Sistemas de Información.....	16
1.1.1 PSI 1.1: Análisis de la Necesidad del PSI	16
1.1.2 PSI 1.2: Identificación del Alcance del PSI.....	17
1.1.3 PSI 1.3: Determinación de Responsables.....	18
1.2 PSI 2: Definición y Organización del PSI	19
1.2.1 PSI 2.1: Especificación del Ámbito y Alcance	19
1.2.2 PSI 2.2: Organización del PSI.....	21
1.3 PSI 3: Estudio de la Información Relevante	22
1.3.1 PSI 3.1: Selección y Análisis de Antecedentes	24
Capítulo 2 PSI 7: Definición de la Arquitectura Tecnológica	26
2.1.1 PSI 7.1: Identificación de las Necesidades de Infraestructura Tecnológica.....	27
2.1.2 PSI 7.2: Selección de la Arquitectura Tecnológica	28
Capítulo 3 Estudio de Viabilidad del Sistema.....	29
3.1 EVS 4, 5 y 6: Estudio y Valoración de Alternativas de Solución y Selección de Alternativa Final 30	
3.1.1 Alternativas jornada laboral	30
3.1.2 Alternativas visualización información por parte de los pasajeros.....	33
3.1.3 Alternativas gestión vacaciones y días libres.....	35
Capítulo 4 Planificación y Gestión del TFG	38
4.1 Planificación del proyecto	39
4.1.1 OBS y PBS	39
4.1.2 Planificación Inicial. WBS	41
4.1.3 Riesgos	48
4.1.4 Presupuesto Inicial.....	49
4.2 Ejecución del Proyecto	56
4.2.1 Bitácora de Incidencias del Proyecto	56
4.3 Cierre del Proyecto.....	57



4.3.1	Planificación Final	57
4.3.2	Presupuesto Final.....	59
4.3.3	Informe de Cierre de Proyecto	61
Capítulo 5	Análisis del Sistema de Información	63
5.1	ASI 1: Definición del Sistema	64
5.1.1	Determinación del Alcance del Sistema	64
5.2	ASI 2: Establecimiento de Requisitos	65
5.2.1	Obtención de los Requisitos del Sistema.....	65
5.2.2	Identificación de Actores del Sistema.....	70
5.2.3	Especificación de Casos de Uso	72
5.3	ASI 3: Identificación de Subsistemas de Análisis.....	78
5.3.1	Descripción de los Subsistemas	78
5.3.2	Descripción de los Interfaces entre Subsistemas	79
5.4	ASI 4: Análisis de los Casos de Uso.....	80
5.4.1	Caso de Uso 1: Iniciar de Sesión	80
5.4.2	Caso de Uso 2: Consultar rutas y horarios.....	81
5.4.3	Caso de uso 3: Añadir nuevo usuario	81
5.4.4	Caso de uso 4: Gestionar solicitudes de vacaciones y días libres (administrador)	82
5.4.5	Caso de uso 5: Gestionar jornada laboral.....	83
5.4.6	Caso de uso 6: Solicitar cambio de jornada a compañero.....	83
5.4.7	Caso de uso 7: Aceptar cambio de jornada	84
5.4.8	Caso de uso 8: Solicitar vacaciones.....	85
5.4.9	Caso de uso 9: Solicitar días libres	85
5.4.10	Caso de uso 10: Notificar y comprobar incidencias.....	86
5.4.11	Caso de uso 11: Consultar jornada laboral	87
5.5	ASI 5: Análisis de Clases.....	88
5.5.1	Diagrama de Clases	88
5.5.2	Descripción de las Clases	89
5.6	ASI 8: Definición de Interfaces de Usuario	94
5.6.1	Descripción de la Interfaz	94
5.6.2	Descripción del Comportamiento de la Interfaz.....	102



5.6.3	Diagrama de Navegabilidad	102
5.7	ASI 10: Especificación del Plan de Pruebas	103
5.7.1	Pruebas unitarias	103
5.7.2	Pruebas de integración	103
5.7.3	Pruebas de usabilidad	103
Capítulo 6	Diseño del Sistema de Información	104
6.1	DSI 3: Diseño de Casos de Uso Reales.....	105
6.1.1	Caso de Uso 2: Consultar rutas y horarios.....	105
6.1.2	Caso de uso 4: Gestionar solicitudes de vacaciones y días libres (administrador)	105
6.1.3	Caso de uso 5: Gestionar jornada laboral.....	106
6.1.4	Caso de uso 6: Solicitar cambio de jornada a compañero.....	107
6.1.5	Caso de uso 7: Aceptar cambio de jornada	107
6.1.6	Caso de uso 8: Solicitar vacaciones.....	108
6.1.7	Caso de uso 9: Solicitar días libres	108
6.2	DSI 4: Diseño de Clases.....	110
6.2.1	Diagrama de Clases	110
6.3	DSI 5: Diseño de la Arquitectura de Módulos del Sistema	126
6.3.1	DSI 5.1 Diseño de Módulos del Sistema	126
6.3.2	DSI 5.3 Revisión de la Interfaz de Usuario	129
6.4	DSI 6: Diseño Físico de Datos	139
6.4.1	Descripción del SGBD Usado.....	139
6.4.2	Integración del SGBD en Nuestro Sistema.....	139
6.4.3	Diagrama E-R.....	139
6.5	10: Especificación Técnica del Plan de Pruebas	141
6.5.1	Pruebas Unitarias	141
6.5.2	Pruebas de Integración y del Sistema.....	151
6.5.3	Pruebas de Usabilidad.....	154
Capítulo 7	Construcción del Sistema de Información.....	159
7.1	CSI 1: Preparación del Entorno de Generación y Construcción	160
7.1.1	Estándares y normas seguidos.....	160
7.1.2	Lenguajes de programación.....	160



7.1.3	Librerías y frameworks.....	162
7.1.4	Herramientas y programas usados	163
7.2	CSI 2: Generación del Código de los Componentes y Procedimientos	168
7.2.1	JornadaService	168
7.2.2	SolicitudService	169
7.3	CSI 3: Ejecución de las Pruebas Unitarias.....	170
7.3.1	EmpleadoService.....	170
7.3.2	EstacionService	171
7.3.3	IncidenciaService	172
7.3.4	JornadaService	172
7.3.5	RutaService	175
7.3.6	SolicitudService	176
7.3.7	TareaService.....	179
7.3.8	TareaEstacionService	180
7.3.9	TrenService	180
7.4	CSI 4: Ejecución de las Pruebas de Integración.....	182
7.4.1	Usuario no identificado.....	182
7.4.2	Usuario trabajador identificado.....	183
7.4.3	Usuario administrador identificado.....	184
7.5	CSI 5: Ejecución de las Pruebas del Sistema.....	186
7.5.1	Pruebas de Usabilidad.....	186
7.6	CSI 6: Elaboración de los Manuales.....	195
7.6.1	Manual de Instalación.....	195
7.6.2	Manual de Ejecución.....	197
7.6.3	Manual de Usuario.....	199
Capítulo 8	Apéndices.....	213
8.1	Plan de gestión de riesgos.....	214
8.1.1	Planificar la Gestión de Riesgos	214
8.1.2	Identificación de Riesgos	215
8.1.3	Registro de riesgos.....	215
8.1.4	Resolución de Riesgos.....	217



8.2	Conclusiones.....	219
8.3	Ampliaciones	220
8.3.1	Aplicación móvil	220
8.3.2	Centro de notificaciones	220
8.3.3	Ampliar funcionalidad de los usuarios pasajeros	220
8.4	Referencias Bibliográficas	221
8.5	Contenido entregado en los anexos	223
8.5.1	Contenidos	223
8.6	GNU Free Documentation License	225
8.6.2	ADDENDUM: How to use this License for your documents	232



Índice de Tablas

TABLA 1. FASES PLANIFICACIÓN	41
TABLA 2. FASE DOCUMENTACIÓN PLANIFICACIÓN	42
TABLA 3. FASE ANÁLISIS PLANIFICACIÓN	42
TABLA 4. FASE DISEÑO PLANIFICACIÓN	43
TABLA 5. FASE DESARROLLO PLANIFICACIÓN	43
TABLA 6. FASE PRUEBAS PLANIFICACIÓN	44
TABLA 7. PRECIO/HORA RECURSOS DE PERSONAL	49
TABLA 8. PRESUPUESTO DE COSTES INICIAL RESUMEN	50
TABLA 9. PRESUPUESTO INICIAL DOCUMENTACIÓN	51
TABLA 10. PRESUPUESTO INICIAL ANÁLISIS	52
TABLA 11. PRESUPUESTO INICIAL DISEÑO	53
TABLA 12. PRESUPUESTO INICIAL DESARROLLO	53
TABLA 13. PRESUPUESTO INICIAL PRUEBAS	54
TABLA 14. PRESUPUESTO INICIAL OTROS COSTES	54
TABLA 15. PRESUPUESTO FINAL DE COSTES	59
TABLA 16. PRESUPUESTO FINAL DE CLIENTE	61
TABLA 17. INFORME DE CIERRE DE PROYECTO	62
TABLA 18. ESPECIFICACIÓN CASO DE USO. INICIAR SESIÓN	72
TABLA 19. ESPECIFICACIÓN CASO DE USO. CONSULTAR RUTAS Y HORARIOS	72
TABLA 20. ESPECIFICACIÓN CASO DE USO. AÑADIR NUEVO USUARIO	74
TABLA 21. ESPECIFICACIÓN CASO DE USO. GESTIONAR SOLICITUDES DE VACACIONES Y DÍAS LIBRES	74
TABLA 22. ESPECIFICACIÓN CASO DE USO. GESTIÓN JORNADA LABORAL	74
TABLA 23. ESPECIFICACIÓN CASO DE USO. SOLICITAR CAMBIO DE JORNADA A COMPAÑERO	76
TABLA 24. ESPECIFICACIÓN CASO DE USO. ACEPTAR CAMBIO DE JORNADA DE COMPAÑERO	76
TABLA 25. ESPECIFICACIÓN CASO DE USO. SOLICITAR VACACIONES	76
TABLA 26. ESPECIFICACIÓN CASO DE USO. SOLICITAR DÍAS LIBRES	77
TABLA 27. ESPECIFICACIÓN CASO DE USO. NOTIFICAR INCIDENCIA	77
TABLA 28. ESPECIFICACIÓN CASO DE USO. CONSULTAR JORNADA LABORAL	77
TABLA 29. ANÁLISIS CASOS DE USO. INICIAR SESIÓN	80
TABLA 30. ANÁLISIS CASOS DE USO. CONSULTAR RUTAS Y HORARIOS	81
TABLA 31. ANÁLISIS CASOS DE USO. AÑADIR NUEVO USUARIO	82
TABLA 32. ANÁLISIS CASOS DE USO. GESTIONAR SOLICITUD DE VACACIONES Y DÍAS LIBRES	82
TABLA 33. ANÁLISIS CASOS DE USO. GESTIONAR JORNADA LABORAL	83
TABLA 34. ANÁLISIS CASOS DE USO. SOLICITAR CAMBIO DE JORNADA	84
TABLA 35. ANÁLISIS CASOS DE USO. ACEPTAR CAMBIO DE JORNADA	84
TABLA 36. ANÁLISIS CASOS DE USO. SOLICITAR VACACIONES	85
TABLA 37. ANÁLISIS CASOS DE USO. SOLICITAR DÍAS LIBRES	86
TABLA 38. ANÁLISIS CASOS DE USO. NOTIFICAR Y COMPROBAR INCIDENCIAS	86
TABLA 39. ANÁLISIS CASOS DE USO. CONSULTAR JORNADA LABORAL	87
TABLA 40. DESCRIPCIÓN DE LA CLASE EMPLEADO	90
TABLA 41. DESCRIPCIÓN DE LA CLASE JORNADA	90
TABLA 42. DESCRIPCIÓN DE LA CLASE TAREA	90
TABLA 43. DESCRIPCIÓN DE LA CLASE TREN	90
TABLA 44. DESCRIPCIÓN DE LA CLASE INCIDENCIA	91
TABLA 45. DESCRIPCIÓN DE LA CLASE TAREA_ESTACION	91
TABLA 46. DESCRIPCIÓN DE LA CLASE SOLICITUD	91



TABLA 47. DESCRIPCIÓN DE LA CLASE SOLICITUDSIMPLE	91
TABLA 48. DESCRIPCIÓN DE LA CLASE SOLICITUDVACACIONES	91
TABLA 49. DESCRIPCIÓN DE LA CLASE SOLICITUDINTERCAMBIO	92
TABLA 50. DESCRIPCIÓN DE LA CLASE HORARIO	92
TABLA 51. DESCRIPCIÓN DE LA CLASE ESTACION	92
TABLA 52. DESCRIPCIÓN DE LA CLASE RUTA	92
TABLA 53. DESCRIPCIÓN DE LA CLASE RUTA_ESTACION	93
TABLA 54. DESCRIPCIÓN DE LA CLASE TRAYECTO	93
TABLA 55. DESCRIPCIÓN DE LA CLASE ROL	111
TABLA 56. DESCRIPCIÓN DE LA CLASE ESTADOINCIDENCIA	111
TABLA 57. DESCRIPCIÓN DE LA CLASE ESTADOSOLICITUD	112
TABLA 58. DESCRIPCIÓN DE LA CLASE SITUACIÓN	112
TABLA 59. DESCRIPCIÓN DE LA CLASE JORNADACONTROLLER	114
TABLA 60. DESCRIPCIÓN DE LA CLASE SOLICITUDCONTROLLER.....	115
TABLA 61. DESCRIPCIÓN DE LA CLASE RUTACONTROLLER	115
TABLA 62. DESCRIPCIÓN DE LA CLASE EMPLEADOCONTROLLER.....	116
TABLA 63. DESCRIPCIÓN DE LA CLASE ESTACIONCONTROLLER.....	116
TABLA 64. DESCRIPCIÓN DE LA CLASE TRENCONTROLLER	116
TABLA 65. DESCRIPCIÓN DE LA TABLA JORNADASERVICE	118
TABLA 66. DESCRIPCIÓN DE LA CLASE SOLICITUDSERVICE	119
TABLA 67. DESCRIPCIÓN DE LA CLASE RUTASERVICE	119
TABLA 68. DESCRIPCIÓN DE LA CLASE EMPLEADOCONTROLLER.....	120
TABLA 69. DESCRIPCIÓN DE LA CLASE ESTACIONSERVICE	120
TABLA 70. DESCRIPCIÓN DE LA CLASE TRENSERVICE.....	120
TABLA 71. DESCRIPCIÓN DE LA CLASE INCIDENCIA	121
TABLA 72. DESCRIPCIÓN DE LA CLASE TAREASERVICE.....	121
TABLA 73. DESCRIPCIÓN DE LA CLASE TAREAESTACIONSERVICE	121
TABLA 74. DESCRIPCIÓN DE LA TABLA JORNADAREPOSITORY	123
TABLA 75. DESCRIPCIÓN DE LA CLASE SOLICITUDSERVICE	123
TABLA 76. DESCRIPCIÓN DE LA CLASE SOLICITUDSIMPLEREPOSITORY	123
TABLA 77. DESCRIPCIÓN DE LA CLASE SOLICITUDINTERCAMBIOREPOSITORY	124
TABLA 78. DESCRIPCIÓN DE LA CLASE SOLICITUDVACACIONESREPOSITORIO	124
TABLA 79. DESCRIPCIÓN DE LA CLASE RUTASERVICE	125
TABLA 80. DESCRIPCIÓN DE LA CLASE EMPLEADOREPOSITORY	125
TABLA 81. DESCRIPCIÓN DE LA CLASE INCIDENCIAREPOSITORY	125
TABLA 82. DESCRIPCIÓN DE LA CLASE HORARIOREPOSITORY	125
TABLA 83. DISEÑO PRUEBAS UNITARIAS – EMPLEADOSERVICE	142
TABLA 84. DISEÑO PRUEBAS UNITARIAS – ESTACIONSERVICE	142
TABLA 85. DISEÑO DE PRUEBAS UNITARIAS – INCIDENCIASERVICE	143
TABLA 86. DISEÑO PRUEBAS UNITARIAS – JORNADASERVICE.....	145
TABLA 87. DISEÑO DE PRUEBAS UNITARIAS - RUTASERVICE.....	146
TABLA 88. DISEÑO DE PRUEBAS UNITARIAS - SOLICITUDSERVICE.....	150
TABLA 89. DISEÑO DE PRUEBAS UNITARIAS - TAREASERVICE	150
TABLA 90. DISEÑO DE PRUEBAS UNITARIAS - TAREAESTACIONSERVICE	150
TABLA 91. DISEÑO DE PRUEBAS UNITARIAS – TRENSERVICE	151
TABLA 92. DISEÑO DE PRUEBAS DE INTEGRACIÓN - USUARIO NO IDENTIFICADO	152
TABLA 93. DISEÑO DE PRUEBAS DE INTEGRACIÓN - USUARIO ADMINISTRADOR.....	153
TABLA 94. DISEÑO DE PRUEBAS DE INTEGRACIÓN - USUARIO ADMINISTRADOR.....	154
TABLA 95. CUESTIONARIO DE PREGUNTAS GENERALES DE USABILIDAD	156



TABLA 96. CUESTIONARIO DE PREGUNTAS CORTAS DE USABILIDAD.....	157
TABLA 97. CUESTIONARIO PARA EL RESPONSABLE DE LAS PRUEBAS DE USABILIDAD.....	158
TABLA 98. EJECUCIÓN DE PRUEBAS UNITARIAS – EMPLEADOSERVICE	171
TABLA 99. EJECUCIÓN DE PRUEBAS UNITARIAS – ESTACIONSERVICE	171
TABLA 100. EJECUCIÓN DE PRUEBAS UNITARIAS – INCIDENCIASERVICE	172
TABLA 101. EJECUCIÓN DE PRUEBAS UNITARIAS- JORNADASERVICE	175
TABLA 102. EJECUCIÓN DE PRUEBAS UNITARIAS - RUTASERVICE	176
TABLA 103. EJECUCIÓN DE PRUEBAS UNITARIAS – SOLICITUDSERVICE	179
TABLA 104. EJECUCIÓN DE PRUEBAS UNITARIAS – TAREASERVICE.....	180
TABLA 105. EJECUCIÓN DE PRUEBAS UNITARIAS - TAREAESTACIONSERVICE	180
TABLA 106. EJECUCIÓN DE PRUEBAS UNITARIAS - TRENSERVICE	180
TABLA 107. EJECUCIÓN DE PRUEBAS DE INTEGRACIÓN - USUARIO NO IDENTIFICADO.....	182
TABLA 108. EJECUCIÓN DE PRUEBAS DE INTEGRACIÓN - USUARIO TRABAJADOR IDENTIFICADO	184
TABLA 109. EJECUCIÓN DE PRUEBAS DE INTEGRACIÓN - USUARIO ADMINISTRADOR IDENTIFICADO.....	185
TABLA 110 VALOR DE LAS PROBABILIDADES	215
TABLA 111. ESCALA DE IMPACTO	216
TABLA 112. MATRIZ DE PROBABILIDAD E IMPACTO.....	216
TABLA 113. REGISTRO DE RIESGOS.....	217
TABLA 114. RESOLUCIÓN DE RIESGOS.....	218
TABLA 115. ESTRUCTURA DEL ARCHIVO ENTREGADO	223
TABLA 116. ESTRUCTURA DE DIRECTORIOS DE DESARROLLO - BACK-END	224
TABLA 117. ESTRUCTURA DE DIRECTORIOS DE DESARROLLO - FRONT-END.....	224
TABLA 118. CONTENIDOS DE LOS ANEXOS.....	224



Índice de Ilustraciones

ILUSTRACIÓN 1. DIAGRAMA DE CONCEPTOS	23
ILUSTRACIÓN 2. LOGO HORAS LABORALES 4B	30
ILUSTRACIÓN 3. CALENDARIO EVENTOS HORAS LABORALES 4B.....	31
ILUSTRACIÓN 4. REGISTRO LABORAL MENSUAL.....	31
ILUSTRACIÓN 5. LOGO JIRA.....	32
ILUSTRACIÓN 6. ESPECIFICACIÓN TAREAS JIRA	32
ILUSTRACIÓN 7. RESUMEN TAREAS JIRA	32
ILUSTRACIÓN 8. LOGO RENFE CERCANÍAS.....	33
ILUSTRACIÓN 9. LOGO ADIF.....	33
ILUSTRACIÓN 10. PANTALLA PRINCIPAL ADIF.....	34
ILUSTRACIÓN 11. HORARIOS TRAYECTO ADIF	34
ILUSTRACIÓN 12. LOGO WOFFU	35
ILUSTRACIÓN 13. SOLICITUDES PENDIENTES VACACIONES WOFFU	35
ILUSTRACIÓN 14. FORMULARIO SOLICITUD VACACIONES WOFFU	35
ILUSTRACIÓN 15. LOGO ABSENCE	36
ILUSTRACIÓN 16. FORMULARIO SOLICITUD VACACIONES ABSENCE.....	36
ILUSTRACIÓN 17. SOLICITUDES PENDIENTES ABSENCE	37
ILUSTRACIÓN 18. OBS.....	39
ILUSTRACIÓN 19. PBS.....	40
ILUSTRACIÓN 20. DIAGRAMA GANTT COMPLETO	45
ILUSTRACIÓN 21. DIAGRAMA GANTT DOCUMENTACIÓN	45
ILUSTRACIÓN 22. DIAGRAMA GANTT ANÁLISIS.....	46
ILUSTRACIÓN 23. DIAGRAMA GANTT DISEÑO.....	46
ILUSTRACIÓN 24. DIAGRAMA GANTT DESARROLLO.....	47
ILUSTRACIÓN 25. DIAGRAMA GANTT PRUEBAS	47
ILUSTRACIÓN 26. IMPORTES INTERMEDIOS PRESUPUESTO CLIENTE.....	54
ILUSTRACIÓN 27. PRESUPUESTO DE CLIENTE INICIAL	55
ILUSTRACIÓN 28. PLANIFICACIÓN FINAL	59
ILUSTRACIÓN 29. DIAGRAMA DE CASOS DE USO. USUARIO ANÓNIMO	72
ILUSTRACIÓN 30. DIAGRAMA DE CASOS DE USO. USUARIO ADMINISTRADOR IDENTIFICADO	73
ILUSTRACIÓN 31. DIAGRAMA DE CASOS DE USO. USUARIO EMPLEADO IDENTIFICADO	75
ILUSTRACIÓN 32. DESCRIPCIÓN DE LOS INTERFACES ENTRE SUBSISTEMAS	79
ILUSTRACIÓN 33. ANÁLISIS. DIAGRAMA DE CLASES	89
ILUSTRACIÓN 34. INTERFAZ ESQUEMA GENERAL.....	94
ILUSTRACIÓN 35. PROTOTIPO PANTALLA PRINCIPAL	95
ILUSTRACIÓN 36. PROTOTIPO CONSULTAR HORARIOS 1.....	95
ILUSTRACIÓN 37. PROTOTIPO CONSULTAR HORARIO 2.....	96
ILUSTRACIÓN 38. PROTOTIPO INICIAR SESIÓN	97
ILUSTRACIÓN 39. PROTOTIPO CONSULTAR JORNADA LABORAL.....	97
ILUSTRACIÓN 40. PROTOTIPO DETALLES DE UNA TAREA.....	98
ILUSTRACIÓN 41. PROTOTIPO SOLICITAR VACACIONES	98
ILUSTRACIÓN 42. PROTOTIPO PORTAL DE SOLICITUDES.....	99
ILUSTRACIÓN 43. PROTOTIPO GESTIONAR PERFIL	100
ILUSTRACIÓN 44. PROTOTIPO GESTIONAR USUARIOS Y JORNADA.....	101
ILUSTRACIÓN 45. PROTOTIPO GESTIONAR SOLICITUDES.....	101
ILUSTRACIÓN 46. DIAGRAMA DE NAVEGABILIDAD	102



ILUSTRACIÓN 47. DIAGRAMA DE INTERACCIÓN CASO DE USO 2	105
ILUSTRACIÓN 48. DIAGRAMA DE INTERACCIÓN CASO DE USO 4	106
ILUSTRACIÓN 49. DIAGRAMA DE INTERACCIÓN CASO DE USO 5	106
ILUSTRACIÓN 50. DIAGRAMA DE INTERACCIÓN CASO DE USO 6	107
ILUSTRACIÓN 51. DIAGRAMA DE INTERACCIÓN CASO DE USO 7	108
ILUSTRACIÓN 52. DIAGRAMA DE INTERACCIÓN CASO DE USO 8	108
ILUSTRACIÓN 53. DIAGRAMA DE INTERACCIÓN CASO DE USO 9	109
ILUSTRACIÓN 54. DIAGRAMA DE CLASES GENERAL	110
ILUSTRACIÓN 55. DIAGRAMA DE CLASES – ENTIDADES	111
ILUSTRACIÓN 56. DIAGRAMA DE CLASES – CONTROLADORES.....	113
ILUSTRACIÓN 57. DIAGRAMA DE CLASES – SERVICIOS	117
ILUSTRACIÓN 58. DIAGRAMA DE CLASES – REPOSITARIOS	122
ILUSTRACIÓN 59. DIAGRAMA DE PAQUETES	127
ILUSTRACIÓN 60. DIAGRAMA DE COMPONENTES	128
ILUSTRACIÓN 61. DIAGRAMA DE DESPLIEGUE	129
ILUSTRACIÓN 62. PANTALLA DE INICIO	130
ILUSTRACIÓN 63. PANTALLA CONSULTA DE HORARIOS	130
ILUSTRACIÓN 64. PANTALLA INICIO DE SESIÓN.....	131
ILUSTRACIÓN 65. PANTALLA DE INICIO - TRABAJADOR IDENTIFICADO.....	131
ILUSTRACIÓN 66. PANTALLA CONSULTAR JORNADA.....	132
ILUSTRACIÓN 67. PANTALLA SOLICITUD DE VACACIONES CASO 1	133
ILUSTRACIÓN 68. PANTALLA SOLICITUD DE VACACIONES CASO 2	133
ILUSTRACIÓN 69. PANTALLA PORTAL DE SOLICITUDES	134
ILUSTRACIÓN 70. PANTALLA PORTAL DE SOLICITUDES - NUEVA SOLICITUD	134
ILUSTRACIÓN 71. PANTALLA VER PERFIL	135
ILUSTRACIÓN 72. PANTALLA VER SOLICITUDES.....	136
ILUSTRACIÓN 73. PANTALLA GESTIONAR USUARIOS Y JORNADAS	137
ILUSTRACIÓN 74. PANTALLA GESTIONAR USUARIOS Y JORNADAS - AÑADIR USUARIO.....	137
ILUSTRACIÓN 75. PANTALLA GESTIONAR USUARIOS Y JORNADAS - AÑADIR TAREA	138
ILUSTRACIÓN 76. DIAGRAMA E -R	140
ILUSTRACIÓN 77. CÓDIGO QR PARA CUESTIONARIO DE USABILIDAD.....	158
ILUSTRACIÓN 78. LOGO JAVA	160
ILUSTRACIÓN 79. LOGO TYPESCRIPT	161
ILUSTRACIÓN 80. LOGO HTML5.....	161
ILUSTRACIÓN 81. LOGO CSS.....	161
ILUSTRACIÓN 82. LOGO ANGULAR	162
ILUSTRACIÓN 83. LOGO SPRING FRAMEWORK	162
ILUSTRACIÓN 84. LOGO JUNIT	163
ILUSTRACIÓN 85. INTELLIJ IDEA LOGO.....	163
ILUSTRACIÓN 86. LOGO MYSQL.....	163
ILUSTRACIÓN 87. LOGO MYSQL WORKBENCH	164
ILUSTRACIÓN 88. LOGO GITHUB.....	164
ILUSTRACIÓN 89. LOGO MOZILLA FIREFOX	164
ILUSTRACIÓN 90. LOGO SELENIUM	165
ILUSTRACIÓN 91. LOGO WORD.....	165
ILUSTRACIÓN 92. LOGO EXCEL.....	165
ILUSTRACIÓN 93. LOGO PROJECT	166
ILUSTRACIÓN 94. LOGO POWERPOINT	166
ILUSTRACIÓN 95. LOGO FORMS	166



ILUSTRACIÓN 96. LOGO DRAW.IO	167
ILUSTRACIÓN 97. LOGO BALSAMIQ.....	167
ILUSTRACIÓN 98. JAVADOC JORNADASERVICE.....	168
ILUSTRACIÓN 99. JAVADOC SOLICITUDSERVICE	169
ILUSTRACIÓN 100. EJECUCIÓN PRUEBAS DE USABILIDAD - PREGUNTAS DE CARÁCTER PERSONAL.....	187
ILUSTRACIÓN 101. EJECUCIÓN DE PRUEBAS DE USABILIDAD - ACTIVIDADES GUIADAS	190
ILUSTRACIÓN 102. EJECUCIÓN DE PRUEBAS DE USABILIDAD - PREGUNTAS CORTAS	192
ILUSTRACIÓN 103. EJECUCIÓN PRUEBAS DE USABILIDAD - CUESTIONARIO RESPONSABLE DE PRUEBAS	194
ILUSTRACIÓN 104. CONFIGURACIÓN DE JAVA	195
ILUSTRACIÓN 105. DESCARGAR MYSQL	196
ILUSTRACIÓN 106. CONFIRMACIÓN DESCARGA MYSQL	196
ILUSTRACIÓN 107. CREACIÓN JAR -MAVEN INSTALL.....	197
ILUSTRACIÓN 108. CARGA SCRIPT – MENÚ IMPORT	198
ILUSTRACIÓN 109. CARGA SCRIPT - CONFIGURACIÓN IMPORT	199
ILUSTRACIÓN 110. PANTALLA PRINCIPAL	200
ILUSTRACIÓN 111. BOTÓN MAPA Y HORARIOS.....	200
ILUSTRACIÓN 112. MAPA Y HORARIOS.....	201
ILUSTRACIÓN 113. BOTÓN IDENTIFICARSE	201
ILUSTRACIÓN 114. INICIO DE SESIÓN	201
ILUSTRACIÓN 115. MENÚ CONSULTAR JORNADA	202
ILUSTRACIÓN 116. CONSULTAR JORNADA	202
ILUSTRACIÓN 117. CONSULTAR JORNADA - SOLICITAR DÍA LIBRE	203
ILUSTRACIÓN 118. DETALLES DE TAREA	203
ILUSTRACIÓN 119. MENÚ SOLICITAR VACACIONES.....	204
ILUSTRACIÓN 120. SOLICITAR VACACIONES	204
ILUSTRACIÓN 121. SOLICITUD DE VACACIONES - EXISTENTE	205
ILUSTRACIÓN 122. MENÚ PORTAL DE SOLICITUDES	205
ILUSTRACIÓN 123. PORTAL DE SOLICITUDES.....	206
ILUSTRACIÓN 124. SOLICITAR INTERCAMBIO	206
ILUSTRACIÓN 125. MENÚ VER PERFIL	207
ILUSTRACIÓN 126. VER Y MODIFICAR CONTRASEÑA	207
ILUSTRACIÓN 127. NUEVA CONTRASEÑA.....	208
ILUSTRACIÓN 128. MENÚ VER SOLICITUDES	208
ILUSTRACIÓN 129. VER SOLICITUDES	208
ILUSTRACIÓN 130. MENSAJE CONFIRMACIÓN SOLICITUD	209
ILUSTRACIÓN 131. MENÚ GESTIONAR USUARIOS Y JORNADAS	209
ILUSTRACIÓN 132. GESTIÓN USUARIOS	210
ILUSTRACIÓN 133. AÑADIR NUEVO USUARIO	210
ILUSTRACIÓN 134. GESTIÓN DE USUARIOS Y JORNADAS	211
ILUSTRACIÓN 135. NUEVA TAREA	212
ILUSTRACIÓN 136. DIAGRAMA GESTIÓN DE RIESGOS	214
ILUSTRACIÓN 137. ESTRUCTURA DE DESGLOSE DE RIESGOS	215



Capítulo 1 PLANIFICACIÓN DEL SISTEMA DE INFORMACIÓN

FASE DE PLANIFICACIÓN

PSI



1.1 PSI 1: INICIO DEL PLAN DE SISTEMAS DE INFORMACIÓN

Renfe Cercanías[2] es una división de la compañía Renfe dedicada a conectar los núcleos urbanos cercanos a las grandes ciudades mediante líneas de alta capacidad y frecuencia. Esta división opera actualmente en 12 provincias españolas, pero este proyecto únicamente estará enfocado al área del Principado de Asturias; de todas formas, esta aplicación podría ser extrapolable a los demás núcleos.

Es por lo anteriormente mencionado, que cuando a lo largo del documento se mencione Renfe o Renfe Cercanías referirá siempre a Renfe Cercanías Asturias.

Con la informatización de Renfe Cercanías en todos los aspectos, ha surgido la necesidad de que con ella se actualicen las comunicaciones y la gestión de las labores y jornadas del personal de la empresa.

Al igual que en cualquier empresa, ya sea una de grandes dimensiones o una PIME, cuando se introduce un nuevo sistema informático el principal objetivo es su correcta utilización y su rápida implantación en la medida de lo posible; para ello es fundamental que en primer lugar el tiempo de formación necesario sea el menor posible abarcando una completa comprensión por parte del personal.

En este caso, dado que la aplicación está dirigida y enfocada principalmente en los empleados de Renfe, el argumento anterior cobra aún más sentido; por lo que se busca realizar una aplicación fácil de usar y que unifique las existentes actualmente.

En este apartado del documento se realizará una primera definición del trabajo a realizar, indicando los objetivos generales y a raíz de qué necesidades surge la creación de esta aplicación web y cuáles son las ventajas que aporta.

Adicionalmente, se indican los usuarios a quienes va dirigida la aplicación y los cuales interactuarán directamente con ella.

1.1.1 PSI 1.1: Análisis de la Necesidad del PSI

A medida que han pasado los años, con la necesidad de informatización para gestionar tanto al personal, como para visualización de información necesaria para empleados y usuarios, han surgido nuevas aplicaciones móvil y web para realizar dichas funciones.

Actualmente, la compañía Renfe Cercanías cuenta con un sistema informatizado, pero aún con mucho campo de mejora. Es por ello que se ha decidido realizar una aplicación cuyo objetivo es principalmente unificar en un solo sitio parte de la información necesaria para el personal, como es, su jornada laboral; y, adicionalmente aportar cierta información concreta como los horarios a los pasajeros de la compañía.



Relativo a lo anteriormente expuesto sobre la fusión de la información y de las aplicaciones existentes en un único lugar, se debe destacar como principal móvil de la tarea facilitar el trabajo de los empleados.

Las funcionalidades que se pretende aunar en esta nueva aplicación son la visualización de información necesaria para las jornadas laborales de los trabajadores, la gestión de las vacaciones, cambios de jornadas, solicitudes de días libres y pequeñas funcionalidades para los usuarios de Renfe.

La aplicación es principalmente gestionada por los administradores quienes son los encargados de introducir toda la información necesaria en la web. Por otra parte, tenemos los perfiles de los trabajadores que se desarrollan detalladamente más adelante en la documentación y el perfil de usuario que hace consumo de Renfe.

1.1.2 PSI 1.2: Identificación del Alcance del PSI

La aplicación web propuesta pretende facilitar el trabajo de los empleados de la compañía de manera que el tiempo empleado en la consulta y la introducción de información en las diferentes plataformas se vea considerablemente reducido. Por otra parte, además de los propios maquinistas, este sistema facilita la tarea de los administradores y de los responsables.

Actualmente, para la funcionalidad de solicitar cambios de jornadas entre compañeros los empleados se comunican entre ellos vía plataforma, como puede ser WhatsApp o Email, y de manera extraoficial llegan a un acuerdo, una vez ambas partes estén conformes, se lo comunican al responsable de Renfe encargado de organizar las jornadas para que oficialmente se aplique este cambio. Esto tiene la ventaja de que no requiere una formación adicional para realizarse, pero es un proceso informal y tedioso. Con la propuesta de este proyecto lo que se permite es automatizar la solicitud de estos cambios de jornada, de forma que no se necesite la interacción de un administrador y ambas jornadas se actualicen automáticamente tras el intercambio.

Por otra parte, los empleados de Renfe pueden consultar información sobre su jornada laboral a través de una aplicación muy básica creada por un trabajador de la propia compañía. Esta no es una aplicación oficial, solamente es usada entre compañeros. La ventaja que tiene esta app es que es una aplicación móvil para Android, y existe una muy similar para iOS. Sin embargo, es una aplicación compleja, con mala usabilidad y no automatizada en cuanto a cualquier cambio en la planificación mensual. Esta información ha podido ser obtenida gracias a la colaboración de empleados de Renfe que me han permitido navegar detenidamente por dichas aplicaciones; como estas son exclusivas del personal y no están disponibles para descarga externa, por lo que no se dispone de ninguna imagen representativa.

Mi propuesta plantea la idea de que sea la propia empresa la que gestione la aplicación, en este caso un administrador; de esta forma, se exime a los trabajadores de introducir los datos de las jornadas a mano.



1.1.3 PSI 1.3: Determinación de Responsables

A pesar de que la temática del proyecto sea una aplicación destinada a la compañía Cercanías Renfe Asturias, se debe destacar que la empresa no está involucrada de ninguna manera con el trabajo. Dado que el proyecto no es en pareja y no está asociado directamente a una empresa. Yo, propietaria de esta aplicación, cubriré los siguientes roles:

- Diseñadora
- Desarrolladora Software
- Arquitecta Software
- Analista
- Administrador de Base de Datos
- Tester

1.2 PSI 2: DEFINICIÓN Y ORGANIZACIÓN DEL PSI

1.2.1 PSI 2.1: Especificación del Ámbito y Alcance

El sistema consiste en una aplicación web para la planificación y gestión de los propios empleados de Renfe Cercanías Asturias y la visualización de información sencilla para pasajeros.

La aplicación podrá ser usada por empleados de Renfe (principalmente maquinistas), por los administradores de la empresa y por los pasajeros de la compañía. Los empleados registrados podrán comprobar su jornada laboral, recorridos, horarios y solicitar vacaciones y cambios de jornada; por otra parte, los usuarios registrados bajo el perfil de “pasajero” tendrán únicamente acceso a su perfil y a los horarios de los trenes.

Fase 1: Realización de la documentación inicial

En esta primera fase, junto con el tutor, se examina la plantilla y se adecúa la misma al proyecto actual. Se estudia en detalle el sistema actual y las posibles alternativas existentes al problema que plantea solucionar este proyecto; además, se habla con algunos de los usuarios relacionados con el proyecto, como empleados de la compañía, para saber cuáles son las necesidades a las que nos enfrentamos.

Se comienza con la planificación del Sistema de Información, la definición de la arquitectura tecnológica y el estudio de la viabilidad del sistema, la planificación, el análisis y el diseño del sistema de información.

Objetivos:

- Realizar un documento de planificación inicial que contenga las que serán las bases del proyecto.
- Conocer en detalle el sistema actual y las ofertas de mercado.
- Conocer las necesidades de los empleados de la empresa.

Fase 2: Inicio de sesión y visualización del perfil

La aplicación web estará diseñada para ser usada por diferentes perfiles de usuarios. Por una parte, tendremos los usuarios “pasajeros” cuya única funcionalidad disponible en la aplicación será la visualización de información. Varios tipos de usuarios empleados que se especifican más adelante en el documento (maquinistas, revisores...) a quienes estará dirigida principalmente la aplicación y que podrán hacer uso de las funcionalidades más importantes como gestión de la jornada laboral y solicitud e intercambio de vacaciones; y por último, el perfil del administrador.

Todos ellos podrán iniciar sesión en la aplicación y visualizar la información del perfil de usuario, pero únicamente el administrador será el que tenga el permiso de modificación de esta información de los perfiles.



Objetivos:

- Permitir el registro en la aplicación a todos los tipos de usuarios.
- Permitir la visualización del perfil a todos los tipos de usuarios.
- Permitir la modificación del perfil únicamente al usuario administrador.

Fase 3: Gestión de la jornada laboral

En esta fase comienza la etapa de desarrollo, en primer lugar, se desea que los empleados dispongan de un calendario mensual en el cuál puedan ver los días laborables. En cada uno de estos días vendrá especificado la ruta que el empleado debe realizar, en qué andén se encuentran situados los trenes en los que debe realizar cada tarea y a qué hora sale cada uno de ellos.

Cuando el empleado esté en una jornada de trabajo tendrá la posibilidad de indicar en una sección concreta si el tren en el que ha estado tenía algún avería o incidencia reseñable que se deba indicar para que el próximo trabajador tenga constancia de ello.

Objetivos:

- Visualizar calendario de la jornada laboral.
- Visualizar detalles relativos a una jornada específica.

Fase 4: Solicitud vacaciones y días libres

La empresa impone unos periodos de vacaciones concretos divididos por bloques a los cuales los empleados deben adecuarse. Entre estos periodos, los empleados pueden seleccionar el que mejor se adecúe a sus necesidades. Existen dos grandes periodos; el periodo de invierno y el de verano, ambos suman un total de 34 días no laborables.

Para poder pedir un día libre el empleado debe indicar el motivo de la ausencia. En la compañía Renfe existen varios estados en los que puede estar una jornada. Los días libres como tal, encajarían como lo que la empresa denomina días de licencias (ausencias por asuntos propios); además, existen otras posibles razones como reconocimiento médico u otras más concretas que el trabajador quiera especificar.

En cuanto al intercambio de días con otros compañeros de la compañía, se establece un portal dónde los empleados pueden subir las peticiones de cambio de jornada, y los demás empleados podrán aceptar o no el intercambio. Además, se actualizará de manera automática la planificación mensual de ambos.

Objetivos:

- Los empleados deben de poder coger vacaciones.
- Los empleados deben poder escoger días libres si tienen alguno disponible.
- Solicitar y aprobar peticiones de cambios de jornada.
- Actualización automática de la planificación mensual.

1.2.2 PSI 2.2: Organización del PSI

En la realización de este proyecto solamente están involucradas dos personas que cubren todos los perfiles necesarios: el tutor del proyecto y el alumno.

No se hará uso de ningún material auxiliar para el desarrollo del trabajo.

USUARIO	PERFIL	FUNCIÓN
EQUIPO DE SUPERVISIÓN		
Tutor del proyecto	Jefe de proyecto	Supervisión de la realización del proyecto y comprobación de que se cumplen las dimensiones esperadas.
EQUIPO DE DESARROLLO		
Alumno	Diseñadora	Diseño de la estructura del proyecto, tanto de la parte de documentación como del diseño de la propia aplicación.
Alumno	Desarrolladora Software	Desarrollo del proyecto, incluyendo la parte back-end y la parte front-end del mismo.
Alumno	Arquitecta Software	Encargada de la obtención de requisitos, involucrada en la parte de diseño y encargada de la planificación del proyecto. Además es la encargada de la propia arquitectura del sistema
Alumno	Analista	Encargada de realizar el análisis del proyecto.
Alumno	Administrador de Base de Datos	Encargado de la integración de la base de datos en el sistema y de la gestión de la misma.
EQUIPO DE PRUEBAS Y CARGA DE DATOS		
Alumno	Tester	Encargado de realizar las pruebas unitarias, pruebas de integración y pruebas de usabilidad.



1.3 PSI 3: ESTUDIO DE LA INFORMACIÓN RELEVANTE

En la empresa Renfe Cercanías Asturias existen diferentes tipos de días laborables y días libres; para poder realizar una mejor comprensión de este proyecto se va a proceder a realizar una aclaración de alguno de los conceptos principales.

Relativos a la jornada laboral de los empleados

- **Detraibles:** Días de descanso que la empresa puede quitar al empleado en determinadas condiciones.
- **Licencias:** Días que los empleados pueden seleccionar para ausentarse por asuntos propios.
- **Otras ausencias.**
- **Guardia:** Días que el empleado tiene descanso, pero este puede ser eliminado en caso de que ocurra una urgencia o se deba cubrir otra plaza.
- **Fechas de formación:** Fechas señaladas en las que el empleado tiene cursos formativos específicos de la empresa.

Relativos a los conceptos ferroviarios:

- **Vía:** Estructura compuesta por dos carriles por los cuales circula el tren.
- **Andén:** Plataforma situada a lo largo de la vía en estaciones que permite un fácil acceso al tren.
- **Revisor:** Persona empleada que revisa la validez de los billetes a los pasajeros.

A lo largo del documento se tratan conceptos como “jornada”, “tarea” o “solicitud”; para entender el significado de cada uno de estos términos y así comprender adecuadamente cómo funciona la aplicación se expone el diagrama de la Ilustración 1.

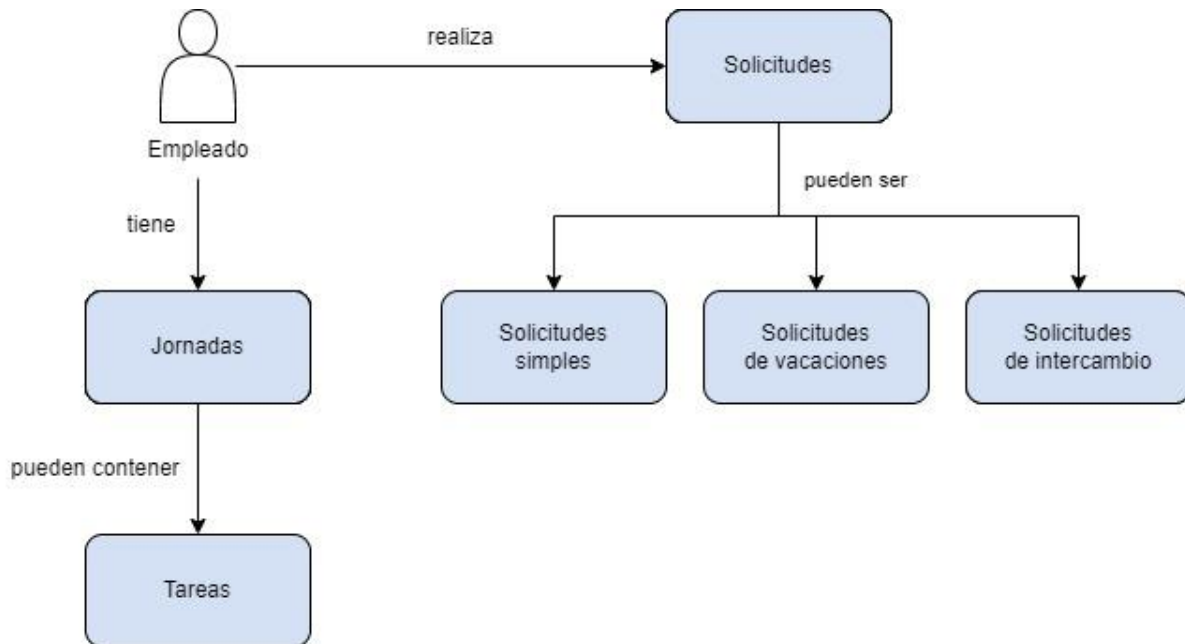


Ilustración 1. Diagrama de conceptos

- **Jornada:** Un empleado tiene asignadas jornadas; estas representan cada uno de los días en los que, o bien el empleado trabaja, o bien el empleado tiene asignado un día libre.
- **Tarea:** Dentro de una jornada el empleado puede realizar varias tareas; es decir, varias actividades o recorridos entre dos estaciones.
- **Solicitudes:** El empleado puede realizar solicitudes de varios tipos
 - **Solicitudes simples:** Las solicitudes simples representan las solicitudes de días libres, es decir, aquellos días que el empleado desea no tener asignada ninguna tarea en su jornada.
 - **Solicitudes de vacaciones:** Las solicitudes de vacaciones representan periodos de tiempo en los que el empleado tendrá asignadas jornadas que sean días libres, es decir no trabajará y por lo tanto, no podrá realizar ninguna tarea.
 - **Solicitudes de intercambio:** Las solicitudes de intercambio representan los intercambios de jornada que se puedan realizar entre diferentes empleados. En caso de producirse un intercambio de jornada entre dos empleados, todas las tareas asignadas a sus jornadas se intercambiarán.

En cuanto a los términos relativos a los recorridos de los trenes se debe aclarar lo siguiente:

- **Estación:** Representa una parada de tren.
- **Ruta:** Recorrido predeterminado compuesto por varias estaciones.
- **Trayecto:** Cada ruta tiene dos trayectos, uno de ida, y otro de vuelta; donde el origen y el destino se intercambian.
- **Horario:** Los horarios permiten saber a qué hora pasará el tren por cada una de las estaciones que forman un trayecto.

1.3.1 PSI 3.1: Selección y Análisis de Antecedentes

En este apartado se trata la situación actual en la que se encuentra la empresa y los principales problemas con los que se encuentra. Además, se presentan los principales documentos a tener en cuenta para la correcta realización del mismo.

Para el estudio de los sistemas actuales se ha utilizado la metodología de entrevistas con los empleados de la empresa, los mismos me han permitido, a través de sus teléfonos, acceder a las aplicaciones de la compañía para poder estudiarlas con detenimiento; además, han tratado de solventar todas las dudas que les he planteado.

Las entrevistas realizadas eran de tipo estructuradas, con una previa preparación de investigación y documentación; a continuación, se procede a la realización de la misma siguiendo la estructura (apertura-desarrollo-terminación) y, por último, se realiza una fase de análisis resumiendo todo lo tratado en un acta. Las principales cuestiones planteadas en estas entrevistas han sido:

- ¿Cómo se seleccionan actualmente las vacaciones?

Respuesta: “Actualmente las vacaciones se solicitan por periodos, al entrar en la compañía se selecciona un periodo, este va cambiando automáticamente cada año. En caso de querer solicitar un día libre, se debe llamar al responsable por teléfono o bien acudir presencialmente a la oficina para notificarlo”

- ¿Cómo se realizan los cambios de jornadas entre compañeros actualmente?

Respuesta: “Actualmente los cambios de jornadas entre compañeros se acuerdan de manera extraoficial por WhatsApp, una vez hemos llegado a un acuerdo dos compañeros, se lo comunicamos al responsable o administrador para que tenga constancia del cambio”

- ¿Cuál es el sistema que se utiliza para la visualización de la jornada laboral?

Respuesta: “Las jornadas laborales actualmente se pueden consultar de manera mensual en una aplicación muy básica que ha creado uno de los empleados, en ella, cargamos el documento inicial que nos proporciona el administrador (un archivo Excel) y nos permite ver la jornada más adecuadamente. El principal problema que presenta esto, es que es una aplicación muy simple al haber sido realizada por una persona sin apenas conocimientos de informática y que funciona de manera automatizada. Cualquier cambio que se produzca, el administrador no es notificado y se debe de realizar “a mano” en la aplicación para así poder ver la jornada actualizada”.

- ¿Qué necesidades tenéis los trabajadores que no cubren las actuales aplicaciones?

Respuesta: “Nos gustaría que todas las acciones que debemos hacer para la gestión de nuestra jornada laboral, tanto la solicitud de las vacaciones, como el intercambio de turnos, como la consulta de la jornada sea más sencilla, se pueda realizar de manera automática y esté unificada en una única aplicación”



- ¿Qué mejoras plantearías en las aplicaciones actuales?

Respuesta: “Básicamente lo mismo que en la pregunta anterior, realización automática de las tareas y unificación de las mismas en una sola aplicación”.

Los empleados de la empresa cuentan con dos periodos de vacaciones, el periodo de invierno compuesto de 20 días, y el periodo de verano compuesto de 14 días. Ambos períodos vienen establecidos por la empresa, y separados en grupos, los empleados podrán seleccionar uno de esos grupos de fechas del periodo. En total, uniendo los días de vacaciones de ambos ciclos, con días de descanso y días de licencia resultan un total de 115 días al año.



Capítulo 2 PSI 7: DEFINICIÓN DE LA ARQUITECTURA TECNOLÓGICA

FASE DE PLANIFICACIÓN

PSI

En este apartado se exponen las alternativas tecnológicas, analizando sus características principales y finalmente haciendo una comparativa y seleccionando aquellas con las que se ha realizado el proyecto.

2.1.1 PSI 7.1: Identificación de las Necesidades de Infraestructura Tecnológica

Para el desarrollo de este proyecto se han estudiado varias alternativas tecnológicas las cuales cito y explico brevemente a continuación. Para ello, se han comparado qué ventajas e inconvenientes aportaban cada una de ellas al enfoque que se tenía sobre el proyecto.

2.1.1.1 Tecnologías de desarrollo

En cuanto a las tecnologías de desarrollo, la decisión estaba entre tres posibles opciones; en primer lugar Node.js, por otra parte Spring Boot y en último lugar Angular.

- Node.js [3] → JavaScript[4]

Node.js es un entorno en tiempo de ejecución basado en una arquitectura orientada a eventos. Entre alguna de sus propiedades destacan la escalabilidad (las aplicaciones que emplean este entorno son capaces de escalar con el negocio con un esfuerzo bastante reducido en comparación con otros), el alto rendimiento que aporta y el uso de microservicios. Además, dispone de un gran número de bibliotecas a través de NPM para cualquier necesidad de desarrollo concreta.

- Spring Framework[5] → Spring Boot → Java

En primer lugar, debemos hacer una breve diferenciación entre Spring y Spring Boot[6].

Spring es un framework basado en Java que nos permite crear controladores Web, de vistas MVC y de aplicaciones REST. Mientras que Spring Boot podría definirse como una extensión de Spring que permite crear proyectos de una manera más ágil.

Sus características principales son la facilidad de configuración; ya que proporciona cantidad de dependencias que podemos añadir a nuestro proyecto de Spring y de terceros. Y, por otra parte, la facilidad de despliegue y facilidad en el acceso a datos.

- Angular → TypeScript[7]

Angular es un framework de código abierto para desarrollar aplicaciones web de una sola página. El lenguaje de programación que utiliza por defecto es TypeScript. Este framework está basado en componentes, estos son los que harán de páginas de la web, permitiendo cambiar la pantalla en la que se encuentra el usuario.

2.1.1.2 Bases de datos

Para la decisión de la base de datos a emplear he comparado dos de las principales bases de datos tanto NoSQL como SQL.

- MongoDB[8]

MongoDB es una base de datos no relacional que como su nombre indica sus consultas no requieren que se entrelacen tablas puesto que no poseen relaciones. Además, no tienen un esquema definido, es decir, cada documento puede tener distintos datos. La ventaja principal es que son fáciles de modificar, es decir si necesitas añadir nuevos campos no es necesario remodelar la tabla completa; asimismo son muy útiles en consultas de lectura y escritura.

- MySQL[9]

Por otra parte, MySQL es una base de datos relacional, cuyo esquema está claramente definido, lo que implica una buena organización y proporciona robustez, pero a su vez rigidez; añadir nuevos campos implica modificar la tabla y todos los registros existentes de la misma derivando en un esquema estático.

2.1.2 PSI 7.2: Selección de la Arquitectura Tecnológica

Dado que ambos aportan unos pros y unos contras a su utilización he decidido que lo mejor es realizar el proyecto en base a una combinación de Java y TypeScript, separando de esta manera el back-end y el front-end en dos aplicaciones diferentes.

Se ha desacoplado el back-end de la aplicación que estará hecho empleando Java (Spring) del front-end que estará realizado con Angular (TypeScript); de esta forma, parte del código podrá aprovecharse en el caso de transformar la aplicación web en una aplicación móvil.

Adicionalmente, esta decisión la he tomado teniendo en cuenta que el lenguaje con el que más cómoda me siento es con Java ya que es el que más en profundidad se estudia en el grado y el que más domino. Para aportar un punto más de aprendizaje se ha seleccionado Angular, tecnología sobre la que no se tienen muchos conocimientos pero muy destacada en la actualidad.

En cuanto a la base de datos a emplear me he decantado por MySQL debido a su rápida recuperación ante fallos y su rendimiento, por otra parte, también ha influido en la decisión que mi experiencia trabajando con bases de datos relacionales es superior a la experiencia con las no relacionales. Concretamente se utilizará MariaDB, un sistema de gestión de bases de datos derivado de MySQL.



Capítulo 3 ESTUDIO DE VIABILIDAD DEL SISTEMA

FASE DE DESARROLLO

EVS

3.1 EVS 4, 5 Y 6: ESTUDIO Y VALORACIÓN DE ALTERNATIVAS DE SOLUCIÓN Y SELECCIÓN DE ALTERNATIVA FINAL

Existen varias aplicaciones que cumplen algunos de los objetivos del proyecto, sin embargo, otros de ellos quedan descubiertos por todas las alternativas estudiadas.

En los siguientes apartados de la sección se muestran las alternativas al proyecto planteado junto con una descripción, indicando sus ventajas y sus inconvenientes. Finalmente se explica la alternativa final junto con el por qué y los objetivos que se consiguen con ello.

Puesto que no existe ningún sistema que cumpla todos los objetivos propuestos de manera simultánea se han estudiado alternativas para cada uno de los objetivos de manera individual.

3.1.1 Alternativas jornada laboral

3.1.1.1 Horas Laborales 4b



Ilustración 2. Logo Horas Laborales 4b

La funcionalidad principal de esta aplicación es llevar un registro horario de los empleados. Estos, con esta aplicación pueden indicar la hora de entrada y de salida laboral. Además, permite ver el calendario mensual con todos los turnos que se tienen asignados.

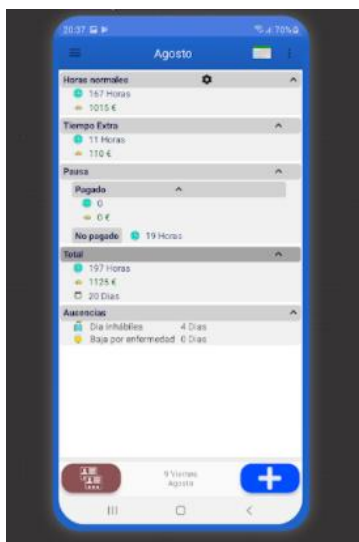


Ilustración 3. Calendario eventos Horas Laborales 4b



Ilustración 4. Registro laboral mensual

Las principales ventajas de esta aplicación es que es una app móvil que puedes llevar siempre contigo y que, además, permite hacer un control horario en tiempo real, cosa que la aplicación propuesta no llega a contemplar.

Como inconveniente cabe resaltar el hecho de que no se plantea un perfil de administrador a través del cual esas tareas son asignadas al calendario y se les incluye una descripción detallada; las tareas que contiene el mismo son añadidas por los propios empleados para organizar su jornada laboral.

3.1.1.2 Jira



Ilustración 5. Logo Jira

Aplicación web que permite a un administrador asignar tareas a diferentes personas, en este caso empleados.

Esta aplicación ofrece la posibilidad de incluir una descripción en las tareas, esto permitiría indicar al trabajador por ejemplo el andén, el tren que debe coger o conducir y los horarios entre otras cosas.

Esta app permite indicar varios estados para una tarea, por lo que, se puede saber en tiempo real si el trabajador ha iniciado el turno o lo ha finalizado. Además, permite introducir las horas empleadas en cada una de las tareas asignadas y marcarlas como “Done” una vez se finalice la misma, por lo que se puede llevar un control de horas trabajadas.

Además, permite tener varios perfiles como maquinistas, revisores, mecánicos y unificar sus tareas mediante el uso de etiquetas comunes en cada uno de los grupos.

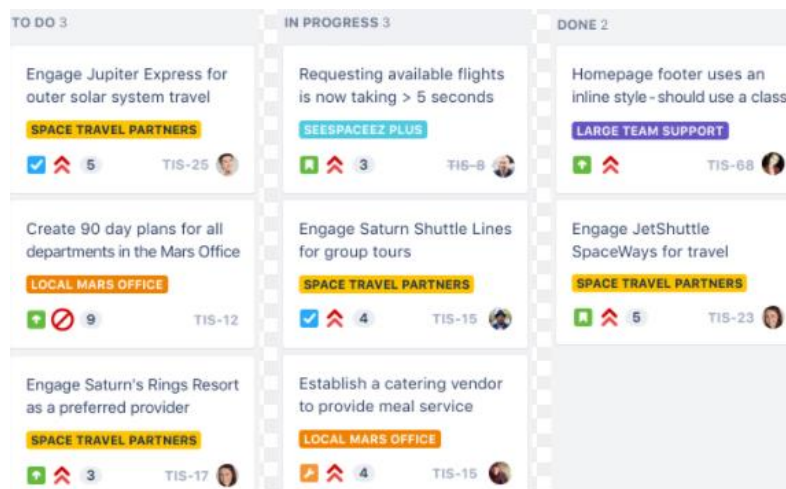


Ilustración 6. Especificación tareas Jira

Type	Key	Summary	Status	Assignee	Due date	Priority
Task	TK00-1	Create pitch deck	TO DO	Alana Song	2021-11-16	Highest
Task	TK00-2	Final up review and feedback gathering	TO DO	Alana Song	2021-07-05	Medium
Task	TK00-3	Competitive analysis	IN PROGRESS	Alana Song	2021-06-26	Low
Task	TK00-7	Gather information for website	IN PROGRESS	Alana Song	2021-07-28	Low
Task	TK00-10	Source and create images	IN PROGRESS	Alana Song	2021-03-03	Medium
Task	TK00-16	Submit creatives brief	DONE	Alana Song	2021-09-13	Low
Task	TK00-17	Audit current experience	DONE	Alana Song	2021-06-01	Low

Ilustración 7. Resumen tareas Jira

La principal ventaja es que es una herramienta muy potente que nos permite cubrir gran número de funcionalidades, pero a su vez, puede que esto se convierta en un inconveniente ya que abarca más de lo necesario y puede acabar complicando el trabajo de los empleados al tener que aprender a manejarla. Otra de sus ventajas es que existe la posibilidad de usarse tanto como aplicación web, como aplicación móvil en Android y en iOS; con lo que el objetivo futuro de la aplicación propuesta en este proyecto de que sea una aplicación móvil queda cubierto.

3.1.2 Alternativas visualización información por parte de los pasajeros

3.1.2.1 Renfe Cercanías

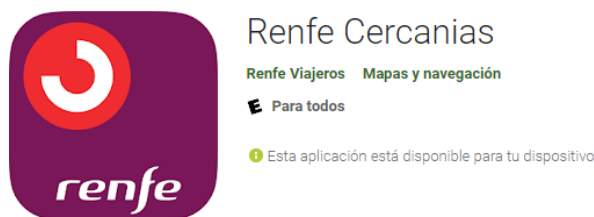


Ilustración 8. Logo Renfe Cercanías

Aplicación móvil que permite la visualización de horarios y mapas de rutas de la compañía. Una de las ventajas que presenta es que unifica en una sola aplicación todas las provincias españolas en las que Renfe Cercanías opera.

Como inconveniente se debe resaltar la falta de una información actualizada de manera constante; es decir, en el caso de que se produzca un retraso en un tren la aplicación no nos informará del mismo.

3.1.2.2 Adif

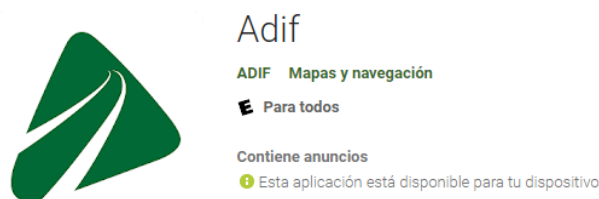


Ilustración 9. Logo Adif

Esta aplicación móvil es gestionada por la Administración de Infraestructura Ferroviaria (Adif) que es otra empresa independiente a Renfe. Permite ver en tiempo real dónde se encuentra el tren, los servicios disponibles en cada una de las estaciones, los mapas de líneas y los horarios para una fecha determinada



Ilustración 10. Pantalla principal Adif



Llegada	Salida	Estación	Vía
	10:10	POLA DE LENA	4
10:13	10:14	VILLALLANA	1
10:16	10:17	UJO	1
10:18	10:20	SANTULLANO	1
10:22	10:22	MIERES-PUENTE	1
10:26	10:26	ABLAÑA	1
10:28	10:28	LA PEREDA-RIOSIA	2
10:31	10:31	OLLONIEGO	2
10:35	10:35	SOTO DE REY	1
10:37	10:37	LAS SEGADAS	1
10:43	10:43	LLAMAQUIQUE	1
10:45	10:46	OVIEDO	1
11:08	11:08	CALZADA ASTURIAS	1
11:11		GIJON-SANZ CRESPO	2

Ilustración 11. Horarios trayecto Adif

La característica más destacable en comparación con la aplicación anterior es, sin duda, la posibilidad de ver en tiempo real la situación y el horario de los trenes. Sin embargo, esta aplicación no permite buscar trenes para una fecha concreta, lo cuál resulta bastante útil ya que el horario entre semana varía frente a los fines de semana.

3.1.3 Alternativas gestión vacaciones y días libres

3.1.3.1 Woffu



Ilustración 12. Logo Woffu

Permite a los empleados solicitar vacaciones y notificar ausencias de manera sencilla. Además, los trabajadores pueden adjuntar todos los justificantes directamente a la aplicación.

Por otra parte, los empleados disponen de un apartado con toda la información necesaria relativa a sus vacaciones y a los días que les quedan libres.

Permite al administrador aceptar o rechazar las peticiones de los empleados de manera sencilla, centralizando en una sola ventana todas estas solicitudes.



Ilustración 13. Solicitudes pendientes vacaciones Woffu



Solicitar

3 días
Disponibles

19 días Solicitados 22 días Asignados

0 días Del próximo ejercicio

Motivo: Vacaciones (3 días)

Fecha inicio: 14/10/2020 Fecha fin: 14/10/2020

Observaciones: Añadir comentario (opcional)

Añadir documentos

Solicitar 1 día

Ilustración 14. Formulario solicitud vacaciones Woffu

3.1.3.2 Absence



Ilustración 15. Logo absence

Aplicación de características similares a la anterior, permite de manera sencilla tanto a los empleados como al personal de RRHH o administradores gestionar las vacaciones de los trabajadores.

Los empleados solicitan las vacaciones o justifican la ausencia seleccionando uno de los tipos disponibles en el desplegable. Además, también pueden ver las vacaciones que les quedan restantes y los días libres al igual que la anterior. Por parte de los administradores pueden aceptar o denegar de manera sencilla las peticiones y el usuario instantáneamente será notificado de la resolución.

La ventaja de esta aplicación frente a la anterior es la facilidad de asignación de sustitutos a los turnos que quedan pendientes por estas ausencias. Sin embargo, el inconveniente es que no permite adjuntar justificantes en las ausencias.

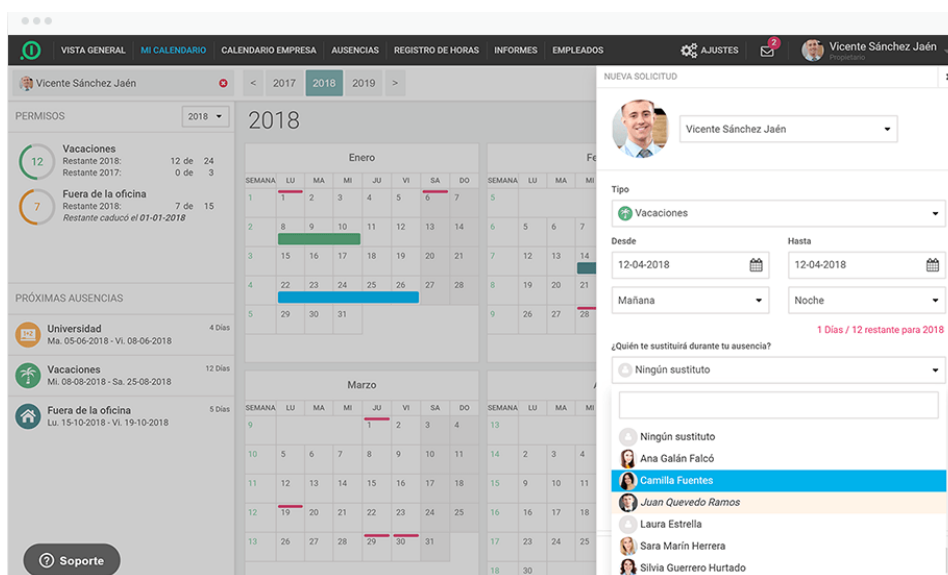


Ilustración 16. Formulario solicitud vacaciones absence











SOLICITUDES DE AUSENCIA		10 fuera de 12	Mostrar en calendario	Mostrar todos
Responsable				
	Daniel Ferrán Fuentes Vicente Sánchez Jaén • Vacaciones • 5 Días	Lu: 19-06-2017 Do: 25-06-2017	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
	Ana Galán Falcó Vicente Sánchez Jaén • Vacaciones • 4 Días	Lu: 26-06-2017 Do: 02-07-2017	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
	María Rodríguez García Vicente Sánchez Jaén • Vacaciones • 5 Días	Lu: 31-07-2017 Do: 06-08-2017	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
	Juan Quevedo Ramos Vicente Sánchez Jaén • Vacaciones • 4 Días	Lu: 04-09-2017 Do: 10-09-2017	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
	Ana Galán Falcó Vicente Sánchez Jaén • Universidad • 2 Días	Ju: 07-09-2017 Vi: 08-09-2017	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
	Juan Quevedo Ramos Vicente Sánchez Jaén • Vacaciones • 1 Día	Vi: 24-11-2017 Do: 25-11-2017	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
	Valentina Lima Margarita García González • Vacaciones • 10 Días	Lu: 02-04-2018 Vi: 13-04-2018	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
	Ana Galán Falcó Vicente Sánchez Jaén • Universidad • 1 Día	Vi: 03-08-2018 Vi: 03-08-2018	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
	Valentina Lima Margarita García González • Fuera de la oficina • 1 Día	Vi: 03-08-2018 Vi: 03-08-2018	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
	Ana Galán Falcó Vicente Sánchez Jaén • Universidad • 2 Días	Lu: 20-08-2018 Mi: 22-08-2018	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

Ilustración 17. Solicitudes pendientes absence

Sin embargo, ninguna de las dos últimas aplicaciones propuestas en este apartado cumple uno de los principales objetivos de este proyecto que es, la existencia de un portal de intercambio de jornada entre compañeros en el cuál no media el administrador hasta una vez llegado a un acuerdo entre los trabajadores.



Capítulo 4 PLANIFICACIÓN Y GESTIÓN DEL TFG

FASE DE DESARROLLO

4.1 PLANIFICACIÓN DEL PROYECTO

En este apartado se desarrolla la planificación del proyecto, incluyendo un OBS y PBS del mismo. Además, se detalla una planificación inicial con un presupuesto inicial y se identifican los riesgos que pueden surgir a lo largo de la vida del proyecto.

4.1.1 OBS y PBS

El OBS (Organization Breakdown Structure) es un organigrama del proyecto, una representación en forma de árbol de las responsabilidades a la hora de realizar el proyecto, permite identificar quién va a hacer cada tarea.



Ilustración 18. OBS

El PBS (Product Breakdown Structure) es, al igual que el anterior, una representación en forma de árbol donde se describen los objetivos del proyecto, es decir, los productos que hay que crear para completar el proyecto.

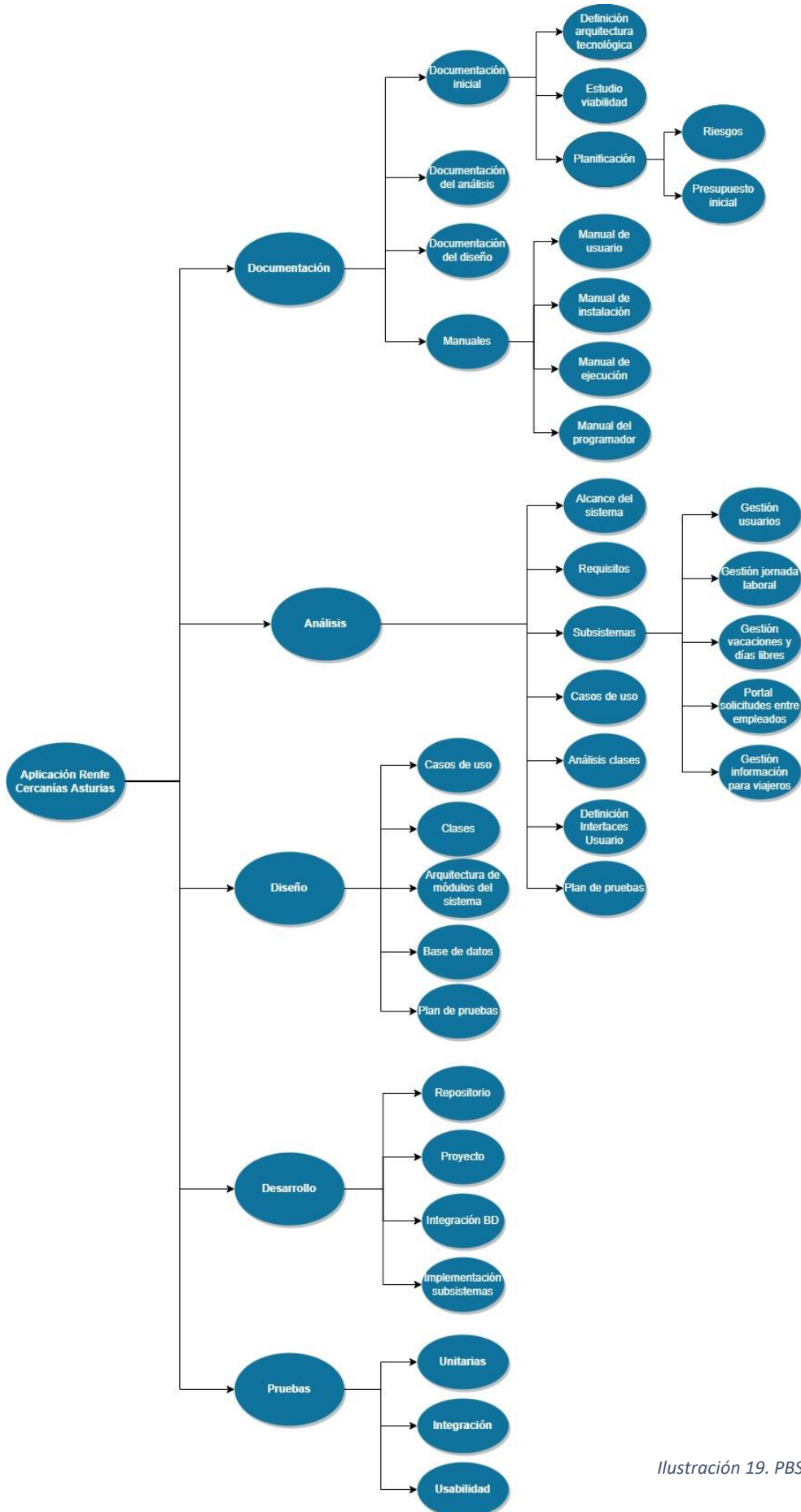


Ilustración 19. PBS

4.1.2 Planificación Inicial. WBS

El WBS (Work Breakdown Structure) es una representación de la estructura de descomposición del trabajo, en él se incluye una lista estructurada de los trabajos necesarios para completar el proyecto.

Para la realización de esta planificación se ha hecho uso de la herramienta Microsoft Project[10]. En este apartado se incluye un cronograma explicativo de cada una de las partes en las que está dividido el proyecto, las subtarefas que las conforman y la duración de las mismas.

El proyecto se divide en cinco fases correspondientes con el WBS:

- Documentación
- Análisis
- Diseño
- Desarrollo
- Pruebas

Se ha creado un nuevo calendario para representar la jornada laboral de los empleados, este consta de semanas de 5 días laborales, con 4 horas diarias de trabajo.

Número de esquema	Nombre de tarea	Duración	Comienzo	Fin
1	Aplicación Web Renfe Cercanías	372 horas	jue 17/02/22	lun 27/06/22
1.1	Documentación	372 horas	jue 17/02/22	lun 27/06/22
1.2	Análisis	52 horas	mié 09/03/22	vie 25/03/22
1.3	Diseño	44 horas	lun 28/03/22	lun 11/04/22
1.4	Desarrollo	196 horas	mar 12/04/22	vie 17/06/22
1.5	Pruebas	200 horas	mié 13/04/22	mar 21/06/22

Tabla 1. Fases planificación

4.1.2.1 Documentación

Esta fase se desarrolla a lo largo de todo el proyecto, engloba tanto la gestión inicial del mismo, como los manuales para la utilización de la aplicación, además de toda la documentación correspondiente al proyecto.

Es por lo anteriormente descrito y por los posibles cambios en ciertas partes de la documentación, como pueden ser los requisitos, a lo largo de la vida del proyecto que esta fase termina una vez finaliza la creación de la aplicación web.

Número de esquema	Nombre de tarea	Duración	Comienzo	Fin
1.1	Documentación	372 horas	jue 17/02/22	lun 27/06/22
1.1.1	Documentación inicial	24 horas	jue 17/02/22	jue 24/02/22
1.1.1.1	Definición arquitectura tecnológica	6 horas	jue 17/02/22	vie 18/02/22
1.1.1.2	Estudio viabilidad	6 horas	vie 18/02/22	lun 21/02/22
1.1.1.3	Planificación	12 horas	mar 22/02/22	jue 24/02/22
1.1.1.3.1	Riesgos	6 horas	mar 22/02/22	mié 23/02/22
1.1.1.3.2	Presupuesto inicial	6 horas	mié 23/02/22	jue 24/02/22
1.1.2	Documentación del análisis	16 horas	vie 25/02/22	mié 02/03/22
1.1.3	Documentación del diseño	16 horas	jue 03/03/22	mar 08/03/22
1.1.4	Manuales	16 horas	mié 22/06/22	lun 27/06/22
1.1.4.1	Manual de usuario	4 horas	mié 22/06/22	mié 22/06/22
1.1.4.2	Manual de instalación	4 horas	jue 23/06/22	jue 23/06/22
1.1.4.3	Manual de ejecución	4 horas	vie 24/06/22	vie 24/06/22
1.1.4.4	Manual del programador	4 horas	lun 27/06/22	lun 27/06/22

Tabla 2. Fase Documentación planificación

4.1.2.2 Análisis

En la fase de análisis se define el alcance del sistema, se definen los requisitos, se realiza un análisis de los subsistemas (incluyendo descripción de los mismos y de los interfaces entre ellos). Por otra parte, se realiza el análisis de los casos de uso y de las clases.

En último lugar se definen las Interfaces de usuario, junto con su aspecto, comportamiento, navegabilidad, etc. y se especifica un plan de pruebas.

Número de esquema	Nombre de tarea	Duración	Comienzo	Fin
1.2	Análisis	52 horas	mié 09/03/22	vie 25/03/22
1.2.1	Alcance del sistema	2 horas	mié 09/03/22	mié 09/03/22
1.2.2	Requisitos	8 horas	mié 09/03/22	vie 11/03/22
1.2.3	Análisis de subsistemas	20 horas	vie 11/03/22	vie 18/03/22
1.2.3.1	Gestión de usuarios	4 horas	vie 11/03/22	lun 14/03/22
1.2.3.2	Gestión jornada laboral	4 horas	lun 14/03/22	mar 15/03/22
1.2.3.3	Gestión vacaciones y días libres	4 horas	mar 15/03/22	mié 16/03/22
1.2.3.4	Portal solicitudes entre empleados	4 horas	mié 16/03/22	jue 17/03/22
1.2.3.5	Gestión información para viajeros	4 horas	jue 17/03/22	vie 18/03/22
1.2.4	Casos de uso	4 horas	vie 18/03/22	lun 21/03/22
1.2.5	Análisis de clases	8 horas	lun 21/03/22	mié 23/03/22
1.2.6	Definición Interfaces de Usuario	4 horas	mié 23/03/22	jue 24/03/22
1.2.7	Especificación plan de pruebas	6 horas	jue 24/03/22	vie 25/03/22

Tabla 3. Fase Análisis planificación

4.1.2.3 Diseño

La fase de diseño consta de cinco subfases, el diseño de los casos de uso reales, el diseño de clases, incluyendo el diagrama de clases; el diseño de la arquitectura de Módulos del sistema y como se comunican entre ellos, el diseño físico de datos, describiendo todo lo relativo a la base de datos, y por último, la especificación técnica del plan de pruebas

Número de esquema	Nombre de tarea	Duración	Comienzo	Fin
1.3	Diseño	44 horas	lun 28/03/22	lun 11/04/22
1.3.1	Casos de uso	6 horas	lun 28/03/22	mar 29/03/22
1.3.2	Clases	6 horas	mar 29/03/22	mié 30/03/22
1.3.3	Arquitectura de módulos del sistema	12 horas	jue 31/03/22	lun 04/04/22
1.3.4	Base de datos	12 horas	mar 05/04/22	jue 07/04/22
1.3.5	Especificación técnica plan de pruebas	8 horas	vie 08/04/22	lun 11/04/22

Tabla 4. Fase Diseño planificación

4.1.2.4 Desarrollo

Esta fase contiene toda la parte del desarrollo de la aplicación, desde la creación de un repositorio en Github, la creación del proyecto inicial, la integración con la BD y la implementación de los cinco subsistemas que conforman la aplicación.

Número de esquema	Nombre de tarea	Duración	Comienzo	Fin
1.4	Desarrollo	196 horas	mar 12/04/22	vie 17/06/22
1.4.1	Repositorio	1 hora	mar 12/04/22	mar 12/04/22
1.4.2	Proyecto	2 horas	mar 12/04/22	mar 12/04/22
1.4.3	Integración BD	8 horas	mar 12/04/22	jue 14/04/22
1.4.4	Implementación subsistemas	185 horas	jue 14/04/22	vie 17/06/22
1.4.4.1	Gestión de usuarios	40 horas	jue 14/04/22	jue 28/04/22
1.4.4.2	Gestión jornada laboral	40 horas	jue 28/04/22	jue 12/05/22
1.4.4.3	Gestión vacaciones y días libres	45 horas	jue 12/05/22	vie 27/05/22
1.4.4.4	Portal solicitudes entre empleados	45 horas	lun 30/05/22	mar 14/06/22
1.4.4.5	Gestión información para viajeros	15 horas	mar 14/06/22	vie 17/06/22

Tabla 5. Fase Desarrollo planificación

4.1.2.5 Pruebas

La fase de la realización de las pruebas engloba a la realización de los tres tipos de pruebas (unitarias, de integración y de usabilidad); las pruebas unitarias y pruebas funcionales se realizan durante la fase de desarrollo y las pruebas de usabilidad una vez esta termine.



Número de esquema	Nombre de tarea	Duración	Comienzo	Fin
1.5	Pruebas	200 horas	mié 13/04/22	mar 21/06/22
1.5.1	Pruebas unitarias	187 horas	mié 13/04/22	jue 16/06/22
1.5.2	Pruebas funcionales	187 horas	mié 13/04/22	jue 16/06/22
1.5.3	Pruebas usabilidad	8 horas	lun 20/06/22	mar 21/06/22

Tabla 6. Fase Pruebas planificación

4.1.2.6 Diagrama de Gantt

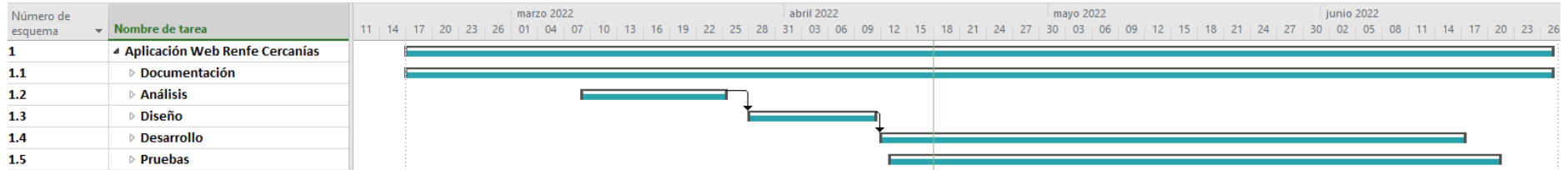


Ilustración 20. Diagrama Gantt completo

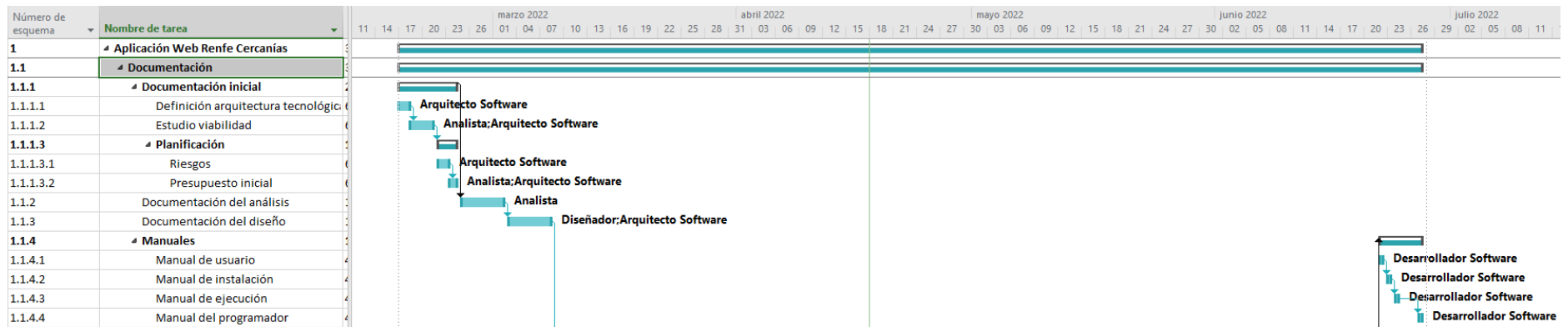


Ilustración 21. Diagrama Gantt documentación

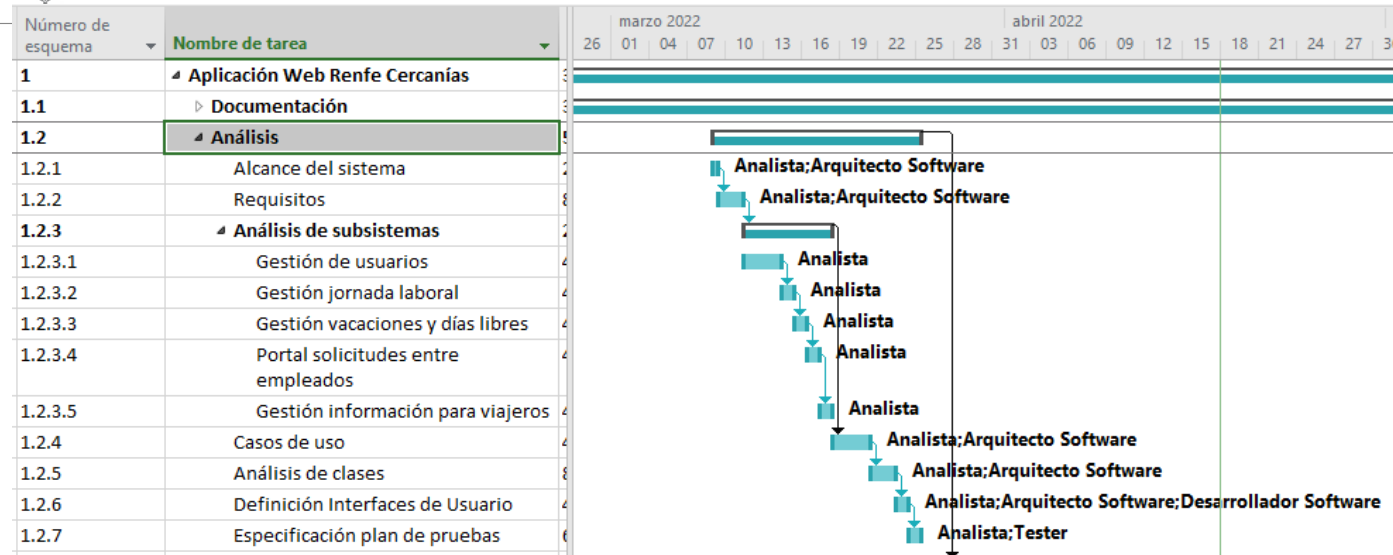


Ilustración 22. Diagrama Gantt análisis

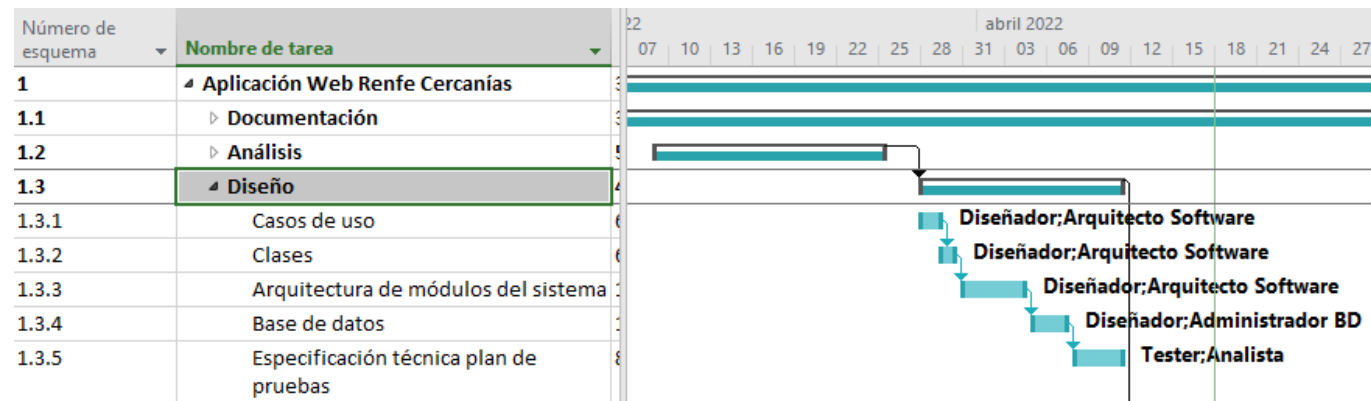


Ilustración 23. Diagrama Gantt diseño

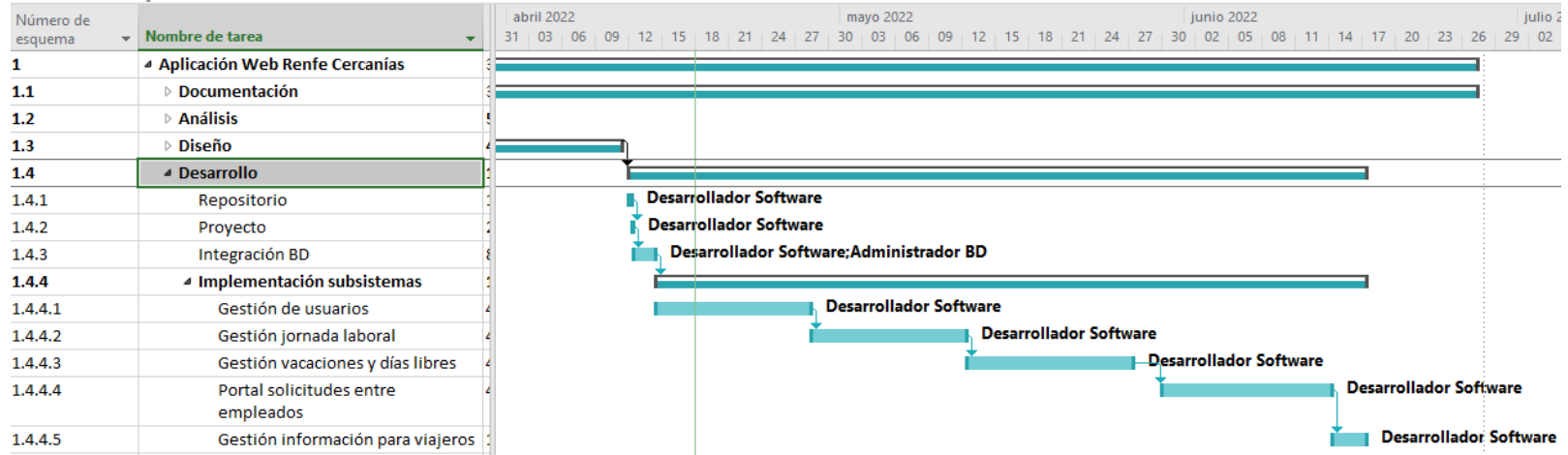


Ilustración 24. Diagrama Gantt desarrollo

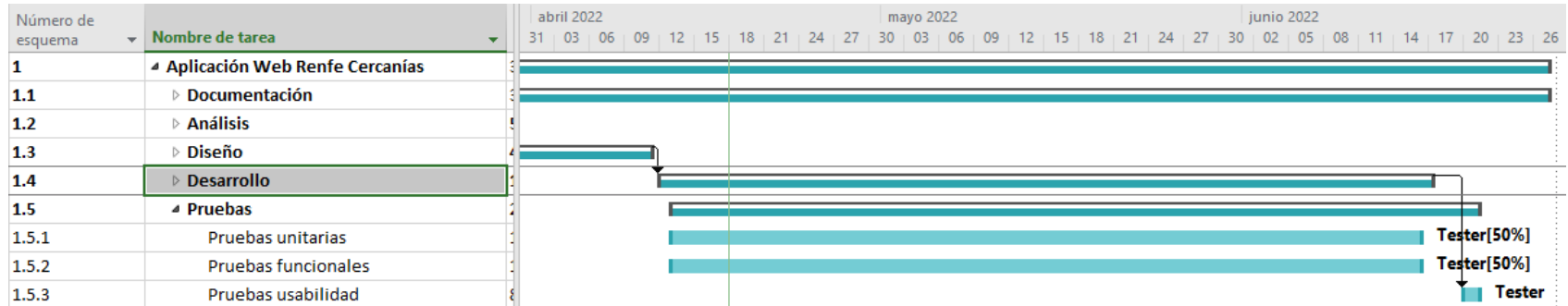


Ilustración 25. Diagrama Gantt pruebas

4.1.3 Riesgos

Denominamos riesgos de un proyecto a aquellos impedimentos que pueden surgir en la realización del mismo y que pueden alterar su trayectoria o afectar a su planificación.

Además de riesgos negativos o amenazas que pueden crear problemas en el proyecto, también existen riesgos positivos u oportunidades; estos eventos favorecen el desarrollo del proyecto.

4.1.3.1 Plan de Gestión de Riesgos

El plan de Gestión de Riesgos está detallado en el anexo denominado *Plan de gestión de riesgos*

4.1.3.2 Identificación de Riesgos

A continuación, se enumeran e identifican los riesgos encontrados para el proyecto de manera priorizada en función de su impacto.

La identificación de Riesgos tiene lugar tras emplear la técnica de brainstorming de manera que surge una lista de riesgos sobre la que se va iterando repetidamente hasta obtener la lista definitiva.

1. Fallos en la estimación temporal: Fallos en las estimaciones temporales debido a alguna situación concreta o enfermedad y se produzcan retrasos de semanas.
2. Fallos en las estimaciones de presupuestos: Fallos en las estimaciones de presupuestos que provoquen una falta de capital a la hora de realizar el proyecto.
3. Se realizan funciones software innecesarias: El desarrollador realiza funciones que se pretenden sean útiles, pero no aportan ninguna ventaja significativa al proyecto y producen a su vez fallos en la estimación temporal.
4. Falta de usabilidad: Una mala usabilidad de la aplicación hace que los usuarios finales no comprendan el sistema y por lo tanto no sepan utilizarlo de forma adecuada.
5. Requisitos mal definidos: Surge la necesidad de modificar los requisitos iniciales por modificación del alcance o por una definición inicial errónea.
6. Inconsistencia a la hora de integrar el front-end y el back-end: A la hora de la integración de los módulos del sistema pueden producirse incompatibilidades.
7. Cambios en el alcance del proyecto: Surgen nuevos requisitos que debemos añadir al proyecto para poder completarlo.
8. Formación escasa del personal: Los empleados de Renfe reciben una escasa formación sobre el uso de la aplicación por lo que no son capaces de usar el nuevo sistema.
9. Falta de conocimiento del dominio o de la tecnología: Es posible que el desarrollador desconozca algunas peculiaridades del dominio del proyecto o alguna de las tecnologías necesarias para la realización del mismo, la búsqueda de esta información puede provocar a su vez retrasos en la planificación del proyecto.

10. Pérdida de datos: Es posible que durante la realización del proyecto se produzcan alguna pérdida de datos relativos al mismo.
11. Los empleados no se adaptan al nuevo sistema: Los empleados de Renfe no se terminan de adecuar al nuevo método de gestión de vacaciones o jornada laboral.
12. Fallos en la migración de datos: Se producen fallos en la migración de datos del antiguo sistema al nuevo.
13. Cambios en las leyes: Se producen cambios en las leyes que alteran ligeramente el funcionamiento del sistema. Por ejemplo, la ley de protección de datos.

4.1.3.3 Registro de Riesgos

El registro de riesgos se realiza en forma de tabla, indicando el identificador del riesgo, el riesgo en sí, su prioridad, la probabilidad de que ocurra y sus medidas de mitigación entre otros datos. Todo esto está incluido en la Tabla 113 del anexo *Plan de gestión de riesgos*.

4.1.4 Presupuesto Inicial

Para la realización del presupuesto inicial se han estudiado las tareas necesarias para completar el proyecto y los materiales necesarios para ello. Dicha información está obtenida principalmente del WBS definido anteriormente. Por otra parte, también se tiene en cuenta los recursos personales necesarios y el coste de los mismos.

En primer lugar, se ha analizado el personal que se requiere para determinar, cuántos empleados de cada perfil se requieren y el precio por hora por el que trabajarán.

4.1.4.1 Recursos de personal

En la tabla mostrada a continuación se indican los puestos de trabajo y el precio por hora. Estos precios han sido extraídos del proyecto realizado en la asignatura de *Dirección y Planificación de Proyectos Informáticos* (DPPI), haciendo uso de la herramienta *LinkedIn Salary*, se ha comprobado que los mismos siguen siendo coherentes actualmente.

Personal	Precio/hora
Jefe de proyecto	38,50€
Administrador BD	20,10€
Diseñador	28,90€
Desarrollador Software	22,90€
Arquitecto Software	30,20€
Analista	31,20€
Tester	29,40€

Tabla 7. Precio/hora recursos de personal

4.1.4.2 Presupuesto de Costes

Tras definir los precios de la mano de obra se clasifican por partidas las diferentes tareas que se deben realizar, haciendo uso del WBS para determinar las mismas. El presupuesto total de cada una de estas partidas depende de las subtareas que estas tengan, además del personal que esté implicado y cuánto cobre por hora.

- Partida 1: Documentación
- Partida 2: Análisis
- Partida 3: Diseño
- Partida 4: Desarrollo del software
- Partida 5: Pruebas
- Partida 6: Otros costes

Una vez tenemos el presupuesto de cada una de las partidas, el presupuesto de costes se obtiene como la suma de las seis partidas.

Presupuesto de costes inicial			
I1	Descripción		Precio
1	Documentación		3.834,40 €
2	Análisis		2.675,60 €
3	Diseño		2.491,20 €
4	Desarrollo		4.649,20 €
5	Pruebas		5.733,00 €
6	Otros costes		6.143,93 €
		TOTAL:	25.527,33 €

Tabla 8. Presupuesto de costes inicial resumen

A continuación, se procede a detallar los costes de cada una de las 6 partidas establecidas.

4.1.4.2.1 Partida 1: Documentación

En la tabla siguiente se muestra el presupuesto de costes inicial para la partida 1 correspondiente con la documentación del proyecto.

Presupuesto de costes inicial									
I1	I2	I3	Descripción	Canti dad	Uds.	Precio	Subtotal(3)	Subtotal(2)	Total
1			Documentación						3.834,40 €
	1		Documentación inicial					2.023,20 €	
			<i>Jefe de proyecto</i>	24	horas	38,50 €	924,00 €		
		1	Definición arquitectura tecnológica						
			<i>Arquitecto Software</i>	6	horas	30,20 €	181,20 €		
	2		Estudio viabilidad						

		<i>Analista</i>	6 horas	31,20 €	187,20 €		
		<i>Arquitecto Software</i>	6 horas	30,20 €	181,20 €		
	3	Planificación					
		Riesgos					
		<i>Arquitecto Software</i>	6 horas	30,20 €	181,20 €		
		Presupuesto inicial					
		<i>Arquitecto Software</i>	6 horas	30,20 €	181,20 €		
		<i>Analista</i>	6 horas	31,20 €	187,20 €		
	2	Documentación del análisis					499,20 €
		<i>Analista</i>	16 horas	31,20 €	499,20 €		
	3	Documentación del diseño					945,60 €
		<i>Diseñador</i>	16 horas	28,90 €	462,40 €		
		<i>Arquitecto Software</i>	16 horas	30,20 €	483,20 €		
	4	Manuales					366,40 €
	1	Manual de usuario					
		<i>Desarrollador Software</i>	4 horas	22,90 €	91,60 €		
	2	Manual de instalación					
		<i>Desarrollador Software</i>	4 horas	22,90 €	91,60 €		
	3	Manual de ejecución					
		<i>Desarrollador Software</i>	4 horas	22,90 €	91,60 €		
	4	Manual del programador					
		<i>Desarrollador Software</i>	4 horas	22,90 €	91,60 €		

Tabla 9. Presupuesto inicial documentación

4.1.4.2.2 Partida 2: Análisis

A continuación, se muestra el presupuesto de costes inicial para la partida 2, análisis del proyecto.

Presupuesto de costes inicial									
I1	I2	I3	Descripción	Cantidad	Uds.	Precio	Subtotal(3)	Subtotal(2)	Total
2			Análisis						2.675,60 €
	1		Alcance del sistema					122,80 €	
			<i>Analista</i>	2 horas		31,20 €	62,40 €		
			<i>Arquitecto Software</i>	2 horas		30,20 €	60,40 €		
	2		Requisitos					491,20 €	
			<i>Analista</i>	8 horas		31,20 €	249,60 €		
			<i>Arquitecto Software</i>	8 horas		30,20 €	241,60 €		
	3		Análisis de subsistemas					624,00 €	
	1		Gestión de usuarios						
			<i>Analista</i>	4 horas		31,20 €	124,80 €		
	2		Gestión jornada laboral						
			<i>Analista</i>	4 horas		31,20 €	124,80 €		
	3		Gestión vacaciones y días libres						

		<i>Analista</i>	4 horas	31,20 €	124,80 €		
4		Portal solicitudes entre empleados					
		<i>Analista</i>	4 horas	31,20 €	124,80 €		
	5	Gestión información para viajeros					
		<i>Analista</i>	4 horas	31,20 €	124,80 €		
4		Casos de uso				245,60 €	
		<i>Analista</i>	4 horas	31,20 €	124,80 €		
		<i>Arquitecto Software</i>	4 horas	30,20 €	120,80 €		
5		Análisis de clases				491,20 €	
		<i>Analista</i>	8 horas	31,20 €	249,60 €		
		<i>Arquitecto Software</i>	8 horas	30,20 €	241,60 €		
6		Definición Interfaces de Usuario				337,20 €	
		<i>Analista</i>	4 horas	31,20 €	124,80 €		
		<i>Arquitecto Software</i>	4 horas	30,20 €	120,80 €		
		<i>Desarrollador Software</i>	4 horas	22,90 €	91,60 €		
7		Especificación plan de pruebas				363,60 €	
		<i>Analista</i>	6 horas	31,20 €	187,20 €		
		<i>Tester</i>	6 horas	29,40 €	176,40 €		

Tabla 10. Presupuesto inicial análisis

4.1.4.2.3 Partida 3: Diseño

En este apartado se muestra la tabla con el presupuesto de costes inicial para la partida 3, correspondiente al diseño.

Presupuesto de costes inicial									
I1	I2	I3	Descripción	Cantidad	Uds.	Precio	Subtotal(3)	Subtotal(2)	Total
3			Diseño						2.491,20 €
	1		Casos de uso					354,60 €	
			<i>Diseñador</i>	6 horas		28,90 €	173,40 €		
			<i>Arquitecto Software</i>	6 horas		30,20 €	181,20 €		
	2		Clases					354,60 €	
			<i>Diseñador</i>	6 horas		28,90 €	173,40 €		
			<i>Arquitecto Software</i>	6 horas		30,20 €	181,20 €		
	3		Arquitectura de módulos del sistema					709,20 €	
			<i>Diseñador</i>	12 horas		28,90 €	346,80 €		
			<i>Arquitecto Software</i>	12 horas		30,20 €	362,40 €		
	4		Base de datos					588,00 €	
			<i>Diseñador</i>	12 horas		28,90 €	346,80 €		

			<i>Administrador BD</i>	12 horas	20,10 €	241,20 €		
5			Especificación técnica plan de pruebas				484,80 €	
			<i>Analista</i>	8 horas	31,20 €	249,60 €		
			<i>Tester</i>	8 horas	29,40 €	235,20 €		

Tabla 11. Presupuesto inicial diseño

4.1.4.2.4 Partida 4: Desarrollo del software

Se detalla a continuación el presupuesto para la fase de desarrollo, perteneciente a la partida 4.

Presupuesto de costes inicial									
11	12	13	Descripción	Cantidad	Uds.	Precio	Subtotal(3)	Subtotal(2)	Total
4			Desarrollo						4.649,20 €
	1		Repositorio					22,90 €	
			<i>Desarrollador Software</i>	1	horas	22,90 €	22,90 €		
	2		Proyecto					45,80 €	
			<i>Desarrollador Software</i>	2	horas	22,90 €	45,80 €		
	3		Integración BD					344,00 €	
			<i>Desarrollador Software</i>	8	horas	22,90 €	183,20 €		
			<i>Administrador BD</i>	8	horas	20,10 €	160,80 €		
	4		Implementación subsistemas					4.236,50 €	
		1	Gestión de usuarios						
			<i>Desarrollador Software</i>	40	horas	22,90 €	916,00 €		
		2	Gestión jornada laboral						
			<i>Desarrollador Software</i>	40	horas	22,90 €	916,00 €		
		3	Gestión vacaciones y días libres						
			<i>Desarrollador Software</i>	45	horas	22,90 €	1.030,50 €		
		4	Portal solicitudes entre empleados						
			<i>Desarrollador Software</i>	45	horas	22,90 €	1.030,50 €		
		5	Gestión información para viajeros						
			<i>Desarrollador Software</i>	15	horas	22,90 €	343,50 €		

Tabla 12. Presupuesto inicial desarrollo

4.1.4.2.5 Partida 5: Pruebas

En la tabla inferior se especifica la partida 5, relativa a la fase de pruebas.

Presupuesto de costes inicial									
11	12	13	Descripción	Cantidad	Uds.	Precio	Subtotal(3)	Subtotal(2)	Total
5			Pruebas						5.733,00 €

1	Pruebas unitarias				2.748,90 €
	Tester	93,5 horas	29,40 €	2.748,90 €	
2	Pruebas integración				2.748,90 €
	Tester	93,5 horas	29,40 €	2.748,90 €	
3	Pruebas usabilidad				235,20 €
	Tester	8 horas	29,40 €	235,20 €	

Tabla 13. Presupuesto inicial pruebas

4.1.4.2.6 Partida 6: Otros costes

Por último, se muestra la tabla con el presupuesto de costes inicial de la última partida, referente a otros costes del proyecto.

Presupuesto de costes inicial								
I1	I2	I3	Descripción	Cantidad	Uds.	Precio	Subtotal(2)	Total
6			Otros costes					6.143,93 €
	1		Consumo eléctrico	389	horas	0,37 €/kWh	143,93 €	
	2		Dietas	500	dietas	12,00 €	6.000,00 €	

Tabla 14. Presupuesto inicial otros costes

4.1.4.3 Presupuesto de Cliente

En este apartado se expone el presupuesto del cliente. Para la realización de este presupuesto se tienen en cuenta las 5 primeras partidas del presupuesto de costes detallado anteriormente. Los costes de la partida 6 (Otros costes) son específicos de este proyecto, por lo que han sido repartidos de manera proporcional entre las otras cinco partidas.

Teniendo en cuenta un presupuesto de costes de 25.527,33€ y aplicando un beneficio del 15% resulta una cantidad de 3.829,1€ asignada a este propósito. Sumando a esta cifra el presupuesto de la partida número 6 se obtiene un total de 9.973,03€ que deben ser promediados en las partidas de cliente seleccionadas para obtener el presupuesto de cliente.

Total proyecto	25.527,33 €
Beneficio (15%)	3.829,10 €
Otros costes	6.143,93 €
Porcentaje a compensar	39%

Ilustración 26. Importes intermedios presupuesto cliente

Para obtener el porcentaje a compensar, es decir, el incremento que debe sufrir el presupuesto de costes debemos aplicar la siguiente fórmula:

$$\text{Incremento} = \frac{\text{Beneficio} + \text{Otros costes}}{\text{Coste total}} = \frac{3.829,10 + 6.143,93}{25.527,33} = 39\%$$

Tras obtener el porcentaje de incremento calculamos los importes totales para el presupuesto del cliente



Presupuesto de costes inicial				
11	12	13	Descripción	Total
1			Documentación	5.332,43 €
2			Análisis	3.720,90 €
3			Diseño	3.464,46 €
4			Desarrollo	6.465,55 €
5			Pruebas	7.972,77 €
			TOTAL:	26.956,12 €

Ilustración 27. Presupuesto de cliente inicial



4.2 EJECUCIÓN DEL PROYECTO

4.2.1 Bitácora de Incidencias del Proyecto

El proyecto se inicia en la fecha esperada, teniendo una reunión con el tutor para aclarar todas las dudas existentes y realizar una explicación detallada de la documentación. A partir de ese momento se comienza a realizar la documentación y el estudio de las tecnologías.

Periódicamente según se va completando la documentación se van mandando versiones al tutor para unas primeras revisiones. Tras cada una de estas revisiones se modifica lo necesario antes de continuar.

El comienzo del proyecto en sí se demora más de lo esperado. Se comienza revisando prácticas de cursos anteriores de asignaturas como *Sistemas Distribuidos e Internet*, *Repositorios de Información* y *Diseño del Software* para refrescar algunos conocimientos. Una vez las ideas están claras se comienza la realización del código en sí.

El principal problema que se ha encontrado a lo largo del proyecto es el desconocimiento del framework. Esto en sí no sería un problema si se dispusiese de más tiempo, pero debido a la ajustada planificación, ha sido un gran impedimento, provocando considerables retrasos.

Otros factores que han hecho que la planificación se vea afectada son el comienzo de una jornada laboral, que provoca que no se dediquen tantas horas como las esperadas, y las clases en la universidad junto con los proyectos a realizar en cada una de las asignaturas. Todo lo anterior mencionado provoca además de retrasos en el proyecto, problemas personales que hacen que no se pueda estar al cien por cien.

En el mes de mayo se tiene una reunión con el tutor. En esta se le comunica que el tiempo para la entrega será muy justo y que se realizará una reestructuración de la planificación para entregar el proyecto en la siguiente convocatoria a la determinada en un primer momento.

Alguna de las incidencias que ocurren en la parte de desarrollo de la aplicación son problemas con el tratamiento de fechas y la decisión de qué tipo de dato utilizar para las mismas. Por otra parte, problemas iniciales de css y html provocan el desbordamiento de componentes en la página, y se emplean aproximadamente 8 horas en conseguir solucionar los problemas de front-end relacionados con ello.

A lo largo del proyecto han surgido infinidad de pequeños problemas, paulatinamente han ido afectando a la planificación. Finalmente se da por terminado el proyecto en el nuevo plazo establecido. Este proyecto permite tener una base bastante sólida, aunque seguramente mejorable en algún aspecto, de lo que podría ser en un futuro una aplicación móvil completamente funcional por la empresa Renfe Cercanías Asturias.



4.3 CIERRE DEL PROYECTO

En este apartado se detalla la planificación final del proyecto, el presupuesto final y en último lugar el informe de cierre del proyecto.

4.3.1 Planificación Final

Como se ha comentado anteriormente, a lo largo del proyecto han surgido varias incidencias que han hecho que la planificación se vea afectada de manera considerable. La fecha inicial de finalización del proyecto era el 27 de Junio de 2022, cuando se aproximaba la fecha, dado lo poco avanzado que estaba el proyecto se decide hacer una nueva planificación y establecer una fecha de finalización del proyecto cercana a la siguiente convocatoria de entrega del TFG.

A lo largo del proyecto se modifica el calendario asignado a la planificación y se establece una jornada laboral de 5 días a la semana 3 horas diarias en una franja horaria entre las 17h y las 20h. Los fines de semana no hay jornada laboral. Además, se establece el mes de agosto como mes no laborable.

Teniendo en cuenta lo mencionado y la adición de varias tareas nuevas, se ha obtenido la planificación final que se muestra a continuación. La fecha de finalización con esta nueva planificación se establece en el 15 de noviembre de 2022.



Número de esquema	Nombre de tarea	Duración	Comienzo	Fin	Predecesoras
1	Aplicación Web Renfe Cercanías	503 horas	jue 17/02/22	jue 10/11/22	
1.1	Documentación	503 horas	jue 17/02/22	jue 10/11/22	
1.1.1	Documentación inicial	51 horas	jue 17/02/22	vie 11/03/22	
1.1.1.1	Planificación del sistema de información	9 horas	jue 17/02/22	lun 21/02/22	
1.1.1.2	Definición arquitectura tecnológica	6 horas	mar 22/02/22	mié 23/02/22	4
1.1.1.3	Estudio viabilidad	9 horas	jue 24/02/22	lun 28/02/22	5
1.1.1.4	Planificación	27 horas	mar 01/03/22	vie 11/03/22	6
1.1.1.4.1	Planificación inicial	9 horas	mar 01/03/22	jue 03/03/22	
1.1.1.4.2	Riesgos	10 horas	vie 04/03/22	mié 09/03/22	8
1.1.1.4.3	Presupuesto inicial	8 horas	mié 09/03/22	vie 11/03/22	9
1.1.2	Documentación del análisis	12 horas	lun 14/03/22	jue 17/03/22	3
1.1.3	Documentación del diseño	9 horas	vie 18/03/22	mar 22/03/22	11
1.1.4	Cierre de proyecto	9 horas	lun 07/11/22	jue 10/11/22	16
1.1.4.1	Planificación final	6 horas	lun 07/11/22	mié 09/11/22	
1.1.4.2	Presupuesto final	3 horas	mié 09/11/22	jue 10/11/22	14
1.1.5	Manuales	16 horas	lun 31/10/22	lun 07/11/22	50
1.1.5.1	Manual de usuario	4 horas	lun 31/10/22	mar 01/11/22	
1.1.5.2	Manual de instalación	4 horas	mar 01/11/22	mié 02/11/22	17
1.1.5.3	Manual de ejecución	4 horas	jue 03/11/22	vie 04/11/22	18
1.1.5.4	Manual del programador	4 horas	vie 04/11/22	lun 07/11/22	19
1.2	Análisis	63 horas	mié 23/03/22	mié 20/04/22	12
1.2.1	Alcance del sistema	2 horas	mié 23/03/22	mié 23/03/22	
1.2.2	Requisitos	12 horas	mié 23/03/22	mar 29/03/22	22
1.2.3	Análisis de subsistemas	10 horas	mar 29/03/22	vie 01/04/22	23
1.2.3.1	Gestión de usuarios	2 horas	mar 29/03/22	mié 30/03/22	
1.2.3.2	Gestión jornada laboral	2 horas	mié 30/03/22	mié 30/03/22	25
1.2.3.3	Gestión vacaciones y días libres	2 horas	jue 31/03/22	jue 31/03/22	26
1.2.3.4	Portal solicitudes entre empleados	2 horas	jue 31/03/22	vie 01/04/22	27
1.2.3.5	Gestión información para viajeros	2 horas	vie 01/04/22	vie 01/04/22	28
1.2.4	Casos de uso	12 horas	lun 04/04/22	jue 07/04/22	24
1.2.5	Análisis de clases	12 horas	vie 08/04/22	mié 13/04/22	30
1.2.6	Definición Interfaces de Usuario	12 horas	jue 14/04/22	mar 19/04/22	31
1.2.7	Especificación plan de pruebas	3 horas	mié 20/04/22	mié 20/04/22	32

1.3	▸ Diseño	53 horas	jue 21/04/22	lun 16/05/22	21
1.3.1	Casos de uso	9 horas	jue 21/04/22	lun 25/04/22	
1.3.2	Clases	12 horas	mar 26/04/22	vie 29/04/22	35
1.3.3	Arquitectura de módulos del sistema	12 horas	lun 02/05/22	jue 05/05/22	36
1.3.4	Base de datos	8 horas	vie 06/05/22	mar 10/05/22	37
1.3.5	Especificación técnica plan de pruebas	12 horas	mar 10/05/22	lun 16/05/22	38
1.4	▸ Desarrollo	281 horas	lun 16/05/22	mié 26/10/22	34
1.4.1	Repositorio	1 hora	lun 16/05/22	lun 16/05/22	
1.4.2	Proyecto	6 horas	mar 17/05/22	mié 18/05/22	41
1.4.3	Integración BD	6 horas	jue 19/05/22	vie 20/05/22	42
1.4.4	▸ Implementación subsistemas	268 horas	lun 23/05/22	mié 26/10/22	43
1.4.4.1	Gestión de usuarios	40 horas	lun 23/05/22	jue 09/06/22	
1.4.4.2	Gestión jornada laboral	54 horas	jue 09/06/22	mar 05/07/22	45
1.4.4.3	Gestión vacaciones y días libres	60 horas	mar 05/07/22	vie 02/09/22	46
1.4.4.4	Portal solicitudes entre empleados	60 horas	vie 02/09/22	vie 30/09/22	47
1.4.4.5	Gestión información para viajeros	54 horas	vie 30/09/22	mié 26/10/22	48
1.5	▸ Pruebas	277 horas	lun 23/05/22	lun 31/10/22	
1.5.1	Pruebas unitarias	268 horas	lun 23/05/22	mié 26/10/22	
1.5.2	Pruebas funcionales	268 horas	lun 23/05/22	mié 26/10/22	
1.5.3	Pruebas usabilidad	7 horas	jue 27/10/22	lun 31/10/22	40

Ilustración 28. Planificación final

4.3.2 Presupuesto Final

En este apartado se muestra el presupuesto final de costes y de cliente. Para la realización del mismo se tienen en cuenta las mismas variables que en el Presupuesto de Costes y Presupuesto de Cliente iniciales. En caso de querer consultar los presupuestos de manera detallada, estos se encuentran como anexo de la entrega con el nombre *presupuestos_TFG_UO266047.xlsx*

Presupuesto de costes final			
11	Descripción		Total
1	Documentación		6.090,40 €
2	Análisis		3.838,60 €
3	Diseño		3.069,50 €
4	Desarrollo		6.555,50 €
5	Pruebas		8.114,40 €
6	Otros costes		6.425,00 €
		TOTAL:	34.093,40 €

Tabla 15. Presupuesto final de costes

A continuación, se muestra el presupuesto de cliente de manera más detallada teniendo en cuenta un beneficio del 15%.

Presupuesto de cliente final					
I1	I2	I3	Descripción	Subtotal(2)	Total
1			Documentación		8.151,71 €
	1		Documentación inicial	6.084,47 €	
		1	Planificación del sistema de infomación		
		2	Definición arquitectura tecnológica		
		3	Estudio viabilidad		
		4	Planificación		
			Planificación inicial		
			Riesgos		
			Presupuesto inicial		
	2		Documentación del análisis	501,12 €	
	3		Documentación del diseño	711,92 €	
	4		Cierre de proyecto	363,79 €	
		1	Planificación final		
		2	Presupuesto final		
	5		Manuales	490,41 €	
		1	Manual de usuario		
		2	Manual de instalación		
		3	Manual de ejecución		
		4	Manual del programador		
2			Análisis		5.137,79 €
	1		Alcance del sistema	164,36 €	
	2		Requisitos	986,17 €	
	3		Análisis de subsistemas	417,60 €	
		1	Gestión de usuarios		
		2	Gestión jornada laboral		
		3	Gestión vacaciones y días libres		
		4	Portal solicitudes entre empleados		
		5	Gestión información para viajeros		
	4		Casos de uso	986,17 €	
	5		Análisis de clases	986,17 €	
	6		Definición Interfaces de Usuario	1.353,98 €	
	7		Especificación plan de pruebas	243,33 €	
3			Diseño		4.108,38 €
	1		Casos de uso	711,92 €	
	2		Clases	949,23 €	
	3		Arquitectura de módulos del sistema	949,23 €	
	4		Base de datos	524,67 €	
	5		Especificación técnica plan de pruebas	973,32 €	
4			Desarrollo		8.774,23 €
	1		Repositorio	30,65 €	
	2		Proyecto	183,90 €	
	3		Integración BD	345,32 €	
	4		Implementación subsistemas	8.214,35 €	

		1	Gestión de usuarios		
		2	Gestión jornada laboral		
		3	Gestión vacaciones y días libres		
		4	Portal solicitudes entre empleados		
		5	Gestión información para viajeros		
5			Pruebas		10.860,74 €
	1		Pruebas unitarias	5.272,97 €	
	2		Pruebas integración	5.272,97 €	
	3		Pruebas usabilidad	314,80 €	
			TOTAL:		37.032,85 €

Tabla 16. Presupuesto final de cliente

Se concluye este apartado comparando el presupuesto inicial de cliente vs. el presupuesto final de cliente, la diferencia entre ambos oscila a 10.076,73€. Este significativo incremento se debe principalmente al aumento de horas empleadas en el proyecto.

4.3.3 Informe de Cierre de Proyecto

En este apartado se cumplimenta el informe de cierre de proyecto.

Informe de situación final

Aplicación de gestión laboral para empleados de Renfe Cercanías Asturias y visualización de información para pasajeros

Informe de proyecto	
Proyecto: TFG	Cliente: EII – Universidad de Oviedo
Título: Aplicación de gestión laboral para empleados de Renfe Cercanías Asturias y visualización de información para pasajeros	
Responsable: Raquel Santos Fernández	
Fecha comienzo: 17/ 02/2022	Fecha fin: 10/11/2022

Trabajo realizado. Alteraciones al alcance previsto
4.2.1 Bitácora de Incidencias del Proyecto
4.3.1 Planificación Final
4.3.2 Presupuesto Final
Dificultades encontradas
8.2 Conclusiones
Relaciones con terceros



Entrevista recogida en el apartado 1.3.1PSI 3.1: Selección y Análisis de Antecedentes

Acciones futuras

8.3 Ampliaciones

Resumen del estado		
	Sí	No
Modificaciones al alcance	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Retrasos	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Sobrecoste	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Insatisfacción del cliente	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Carencia de recursos	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Conflictos interpersonales	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Falta de formación y/o experiencia	<input checked="" type="checkbox"/>	<input type="checkbox"/>

Tabla 17. Informe de cierre de proyecto



Capítulo 5 ANÁLISIS DEL SISTEMA DE INFORMACIÓN

FASE DE DESARROLLO



A lo largo de este capítulo se realiza el análisis del sistema de información.

Este incluye una definición del alcance del mismo, el establecimiento de los requisitos (tanto la obtención de estos, como la identificación de los actores y la especificación de los casos de uso), la identificación de subsistemas y el análisis de los casos de uso y de clases. Además, se definen las interfaces de usuario y se realiza una especificación del plan de pruebas.

Con esta labor de análisis se podrá realizar posteriormente el diseño de manera más concisa y sencilla.

5.1 ASI 1: DEFINICIÓN DEL SISTEMA

5.1.1 Determinación del Alcance del Sistema

Actualmente la empresa Renfe Cercanías Asturias dispone de muchas aplicaciones para la gestión de los empleados, la gestión de la jornada laboral de los mismos y la gestión de información para consulta de los pasajeros de la compañía. El objetivo del sistema a desarrollar es crear una aplicación web que unifique todas las tareas anteriormente mencionadas de forma que pueda sustituir al sistema actual y facilite la labor de tanto el personal de Renfe como de los propios usuarios.

La aplicación web permitirá identificarse a usuarios empleados, haciendo una diferenciación entre administradores y empleados trabajadores en los propios trenes.

Los usuarios administradores se encargarán de la gestión de los usuarios, serán los encargados de la gestión laboral de los mismos y de aceptar las solicitudes de vacaciones y días libres que hagan los trabajadores.

Por otra parte, los usuarios trabajadores podrán consultar su perfil y solicitar un periodo de vacaciones entre los disponibles, que posteriormente será validado o rechazado por los administradores. Adicionalmente, podrán enviar solicitudes de cambio de jornada al portal específico para ello y aceptar las existentes si alguna le interesase. El usuario trabajador también tendrá la posibilidad de indicar cualquier anomalía que detecte en el funcionamiento del tren para que otros usuarios trabajadores tengan constancia de ello.

Los usuarios no registrados corresponderán a los pasajeros y, por tanto, únicamente tendrán disponible la opción de consultar información sobre los horarios de los trenes y sus trayectos.



5.2 ASI 2: ESTABLECIMIENTO DE REQUISITOS

En este apartado se especifican los requisitos del sistema, tanto funcionales como no funcionales. Se ha tomado como referencia la clasificación del estándar IEEE Std. 830 [11]

5.2.1 Obtención de los Requisitos del Sistema

5.2.1.1 *Requisitos de Interfaces Externas*

RIExt1. El sistema deberá seguir los estándares especificador por la W3C en Web Content Accessibility Guidelines (WCAG) 2.0AA [12]

RIExt2. El sistema dispondrá de una base de datos para:

RIExt 2.1. Consultar datos

RIExt 2.2. Almacenar datos

RIExt3. El sistema hará uso de la API de Google Maps para localizar estaciones

5.2.1.2 *Requisitos funcionales*

5.2.1.2.1 Inicio de sesión

RInicio1. El sistema debe permitir iniciar sesión a los usuarios empleados no identificados

RInicio1.1. El usuario debe indicar los siguientes datos:

RInicio1.1.1. Nombre de usuario. (obligatorio)

RInicio1.1.2. Contraseña. (obligatorio)

RInicio1.2. El sistema comprobará que ambos valores existen y que el par usuario, contraseña coincide.

RInicio1.2.1. En caso afirmativo:

RInicio1.2.1.1. El sistema notificará al usuario de que se ha iniciado sesión correctamente.

RInicio1.2.1.2. El usuario pasará a estado autenticado.

RInicio1.2.2. En caso negativo, el sistema notificará al usuario que los datos son incorrectos y los solicitará de nuevo

5.2.1.2.2 Consultar rutas y horarios

RConsultar 1. El sistema debe permitir tanto a usuarios identificados como no identificados consultar los horarios de los trenes indicando.

RConsultar 1.1. Estación de origen (obligatorio)

RConsultar 1.2. Estación de destino (obligatorio)

RConsultar 2. El sistema debe permitir a todos los usuarios consultar un mapa con las estaciones

RConsultar 3. El sistema debe permitir a todos los usuarios consultar el origen y el destino de la ruta seleccionada en RConsultar 1.

5.2.1.2.3 Gestión del perfil

RPerfil1. El sistema debe permitir al usuario empleado consultar su perfil

RPerfil1.1. El usuario empleado podrá comprobar sus datos personales

RPerfil1.1.1. Nombre de usuario

RPerfil1.1.2. Nombre

RPerfil1.1.3. Apellido

RPerfil1.1.4. NIF/NIE

RPerfil1.1.5. Puesto de trabajo (Rol)

RPerfil1.2. El usuario empleado podrá comprobar cuántos días libres restantes le quedan.

RPerfil2. El sistema debe permitir al usuario empleado modificar su contraseña de acceso.

5.2.1.2.4 Gestión de usuarios

RGUsuario1. El sistema debe permitir a los usuarios administradores gestionar los usuarios empleados

RGUsuario1.1. El administrador podrá dar de alta a nuevos usuarios.

RGUsuario1.1.1. Para la creación del perfil de nuevos usuarios el administrador debe indicar:

RGUsuario1.1.1.1. Nombre de usuario (obligatorio)

RGUsuario1.1.1.2. Nombre (obligatorio)

RGUsuario1.1.1.3. Apellido (obligatorio)

RGUsuario1.1.1.4. NIF/NIE (obligatorio)

RGUsuario1.1.1.5. Email

RGUsuario1.1.1.6. Puesto de trabajo (obligatorio)

RGUsuario1.1.2. En caso de que no exista ningún usuario con los siguientes criterios en la base de datos, se añadirá uno nuevo con los datos anteriores.

RGUsuario1.1.2.1. Mismo nombre de usuario

RGUsuario1.1.2.2. Mismo NIF/NIE

RGUsuario1.1.2.3. Mismo email

RGUsuario1.1.3. El sistema debe generar una contraseña aleatoria para el nuevo empleado

RGUsuario1.1.3.1. El empleado será notificado mediante correo electrónico de la misma.

RGUsuario1.1.4. En caso de que exista un usuario con alguno de estos valores se notificará al administrador y se solicitará uno nuevo.

RGUsuario1.2. El administrador podrá consultar los datos personales de los trabajadores

RGUsuario1.3. El administrador podrá filtrar los usuarios empleados por:

RGUsuario1.3.1. Nombre de usuario

RGUsuario1.4. El administrador podrá eliminar usuarios empleados del sistema

RGUsuario1.4.1. El usuario eliminado del sistema no podrá volver a acceder al mismo

RGUsuario1.4.2. El usuario será eliminado de la base de datos

RGUsuario1.4.3. Las jornadas asignadas a ese empleado serán eliminadas de la base de datos



- RGUsuario1.4.4. Las tareas asignadas a ese empleado serán eliminadas de la base de datos
- RGUsuario1.4.5. Las solicitudes asignadas a ese empleado serán eliminadas de la base de datos.

5.2.1.2.5 Gestión jornada laboral

RGJLaboral1. El sistema debe permitir a los usuarios trabajadores consultar su jornada laboral

RGJLaboral1.1. El usuario trabajador debe poder consultar:

RGJLaboral1.1.1. Los días laborales mensuales

RGJLaboral1.1.2. Las tareas que debe realizar cada día

RGJLaboral1.1.3. Detalles de las tareas diarias

RGJLaboral1.1.3.1. Hora de comienzo

RGJLaboral1.1.3.2. Descripción de la tarea

RGJLaboral1.1.3.3. Ruta a realizar

RGJLaboral1.1.3.3.1. Estación de inicio

RGJLaboral1.1.3.3.2. Estación final

RGJLaboral1.1.3.4. Especificaciones del tren

RGJLaboral1.1.3.4.1. Número identificador

RGJLaboral1.1.3.4.2. Andén en el que se encuentra situado

RGJLaboral1.1.3.4.3. Incidencias

RGJLaboral2. Para cada tarea diaria, el usuario empleado debe poder indicar incidencias detectadas en el tren.

RGJLaboral3. El administrador debe poder asignar una jornada laboral a un empleado

RGJLaboral3.1. El administrador debe poder asignar diferentes tareas dentro de una jornada laboral a un empleado

RGJLaboral3.1.1. Se debe indicar:

RGJLaboral3.1.1.1. Empleado (obligatorio)

RGJLaboral3.1.1.2. Día laboral (obligatorio)

RGJLaboral3.1.1.2.1. El día debe ser posterior a la fecha actual

RGJLaboral3.1.1.3. Hora de comienzo de la tarea (obligatorio)

RGJLaboral3.1.1.4. Hora de finalización de la tarea (obligatorio)

RGJLaboral3.1.1.4.1. La hora de finalización debe ser posterior a la hora de comienzo

RGJLaboral3.1.1.5. Andén en el que se encontrará situado el tren (obligatorio)

RGJLaboral3.1.1.6. Descripción de la tarea (obligatorio)

RGJLaboral3.1.1.7. Id del tren (obligatorio)

RGJLaboral3.1.1.8. Origen de la ruta (obligatorio)

RGJLaboral3.1.1.9. Final de la ruta (obligatorio)

RGJLaboral3.1.1.9.1. Debe existir una ruta entre la estación de origen y la estación final.

RGJLaboral3.1.1.9.2. En caso de no existir ruta entre la estación de origen y la estación final se solicitarán ambos datos de nuevo.



RGJLaboral3.2. El sistema debe comprobar que la nueva tarea no coincide con una existente en ese horario para ese empleado.

RGJLaboral3.2.1. En caso de que no coincida se añadirá la tarea del empleado a la base de datos

RGJLaboral3.2.2. En caso de que coincida

RGJLaboral3.2.2.1. El administrador será notificado

RGJLaboral3.2.2.2. Se solicitarán de nuevo los datos

5.2.1.2.6 Gestión vacaciones

RGVacaciones1. Para poder solicitar vacaciones el usuario empleado debe elegir:

RGVacaciones 1.1. Un periodo de vacaciones entre los disponibles (obligatorio).

RGVacaciones 1.2. El sistema debe comprobar que no existe ya una solicitud para ese periodo de vacaciones.

RGVacaciones 1.2.1. En caso de que exista una solicitud:

RGVacaciones 1.2.1.1. El sistema debe avisar al usuario y solicitar de nuevo los datos.

RGVacaciones 1.2.1.2. El sistema debe permitir seleccionar un nuevo periodo en caso de que la solicitud existente tenga un estado RECHAZADA.

RGVacaciones2. El usuario empleado debe poder solicitar días libres.

RGVacaciones 2.1. El empleado debe indicar:

RGVacaciones 2.1.1. Día deseado (obligatorio)

RGVacaciones 2.1.1.1. El día debe ser posterior a la fecha actual

RGVacaciones 2.1.2. Motivo de la ausencia (obligatorio).

RGVacaciones 2.1.2.1. Licencia

RGVacaciones 2.1.2.2. Formación

RGVacaciones 2.1.2.3. Visita médica

RGVacaciones 2.1.2.4. Otro motivo

RGVacaciones 2.2. El empleado no debe haber superado el número de días libres permitidos.

RGVacaciones 2.2.1. El número de días libres permitidos es MAX_DÍAS_LIBRES

RGVacaciones 2.2.1.1. El valor inicial de MAX_DÍAS_LIBRES es 100

RGVacaciones 2.2.1.2. El valor de MAX_DÍAS_LIBRES debe poder ser modificado por el administrador.

RGVacaciones 2.3. Si se cumplen RGVacaciones 2.1 y RGVacaciones 2.2

RGVacaciones 2.3.1. El sistema creará una nueva solicitud simple de día libre para el administrador.

RGVacaciones 2.3.1.1. La solicitud tendrá estado PENDIENTE

RGVacaciones 2.4. En caso contrario

RGVacaciones 2.4.1. El usuario será notificado

RGVacaciones 2.4.2. Los datos serán solicitados de nuevo.

5.2.1.2.7 Gestión de solicitudes entre empleados

RGsolicitudes1. El usuario empleado debe poder realizar solicitudes de cambio de jornada al portal de solicitudes entre compañeros.

RGsolicitudes 1.1. La solicitud debe incluir:

RGsolicitudes 1.1.1. Día que desea tener libre (obligatorio)

RGsolicitudes 1.1.1.1. El día debe ser posterior a la fecha actual

RGsolicitudes 1.1.2. Día que propone trabajar (obligatorio)

RGsolicitudes 1.1.2.1. El día debe ser posterior a la fecha actual

RGsolicitudes 1.1.3. Motivo de la solicitud (obligatorio)

RGsolicitudes 1.2. El empleado no debe haber superado el número de días libres permitidos (RGVacaciones 2.2.1)

RGsolicitudes 1.3. El empleado asignado a la solicitud debe tener el mismo Rol que el empleado que la acepta

RGsolicitudes 1.4. Si se cumplen RGsolicitudes 1.1, RGsolicitudes 1.2 y 69RGsolicitudes 1.3

RGsolicitudes 1.4.1. Se creará una nueva solicitud de intercambio con estado "PENDIENTE"

RGsolicitudes 1.4.2. La solicitud se almacenará en la base de datos

RGsolicitudes 1.5. En caso contrario

RGsolicitudes 1.5.1. El usuario será notificado

RGsolicitudes 1.5.2. Los datos serán solicitados de nuevo

RGsolicitudes2. El usuario empleado debe poder consultar las solicitudes de cambio de jornada realizadas por otros empleados con estado "PENDIENTE" en el portal.

RGsolicitudes3. El usuario empleado debe poder aceptar solicitudes de cambio de jornada del portal si:

RGsolicitudes 3.1. La tarea a cubrir no colisiona con una propia existente.

RGsolicitudes 3.2. Tiene asignada una tarea que coincide con el horario del nuevo descanso

RGsolicitudes4. Si se cumple RGsolicitudes 3.1 y RGsolicitudes 3.2 la solicitud pasará a estado "REASIGNADA" cuando el nuevo empleado la acepte.

5.2.1.2.8 Ver solicitudes (administrador)

RVerSol1. El administrador podrá ver las solicitudes de vacaciones y de días libres con estados "PENDIENTE".

RVerSol2. El administrador podrá aceptar solicitudes

RVerSol 2.1. La solicitud pasará a estado "ACEPTADA"

RVerSol 2.2. Si el tipo de la solicitud es "Solicitud Simple" el empleado pasará a tener la jornada vacía en el día indicado.

RVerSol3. El administrador podrá rechazar solicitudes

RVerSol 3.1. La solicitud pasará a estado "RECHAZADA"



5.2.1.3 Requisitos de rendimiento

N/A

5.2.1.4 Lógicos de Base de Datos

N/A

5.2.1.5 Restricciones de diseño

RNFDiseño 1. El sistema debe garantizar que se siguen los estándares para el desarrollo de la W3C.
RNFDiseño 2. El sistema debe de cumplir Las Pautas de Accesibilidad para el Contenido Web (WCAG) 2.0AA.

5.2.1.6 Atributos del sistema

RNFSistema 1. El sistema debe poder ser usado por cualquier usuario con conocimientos básicos en informática
RNFSistema 2. El sistema debe de poder desplegarse en los siguientes navegadores:
RNFSistema 2.1. Google Chrome
RNFSistema 2.2. Microsoft Edge
RNFSistema 2.3. Mozilla Firefox

5.2.1.7 Otros requisitos

RNFOtros 1. El sistema debe cumplir la Ley Orgánica de Protección de Datos (LOPD), ley 3/2018
RNFOtros 2. El sistema debe cumplir el Reglamento General de Protección de Datos (RGPD), (UE) 2016/679
RNFOtros 3. El sistema debe almacenar las contraseñas encriptadas mediante el algoritmo sha-256.

5.2.2 Identificación de Actores del Sistema

A continuación, se indican los actores del sistema, aquellos usuarios que interactuarán de manera directa o indirecta con el sistema, y cuál es la tarea que lo relaciona con el sistema.

5.2.2.1 Administrador

El administrador es el encargado de la gestión de los otros usuarios, podrá crear, eliminar y consultar la información personal de los empleados.

Este actor se encargará de la gestión de las jornadas y podrá crear tareas para cada una de ellas.



Además, el administrador gestionará todas las solicitudes de vacaciones y días libres que realicen los empleados, pudiendo aceptarlas o rechazarlas.

5.2.2.2 Empleado

Los empleados son aquellos usuarios trabajadores de la compañía que pueden iniciar sesión en la aplicación. Podrán consultar su jornada laboral y la ruta y horarios que deben realizar cada día, solicitar vacaciones y días libres a los administradores, podrán solicitar cambios de jornada con otros compañeros y pueden gestionar su perfil dentro de la aplicación. Dentro del perfil de usuario empleado existen varios roles.

5.2.2.2.1 Revisor

Representa a la persona encargada de ir revisando que los pasajeros dispongan de un ticket que se corresponda con el recorrido que realizan. Se relacionará con la aplicación con las funcionalidades descritas en el apartado 5.2.2.2 Empleado

5.2.2.2.2 Maquinista

Empleado que representa a los conductores de los trenes. Además de lo mencionado, los maquinistas disponen de una sección para notificar anomalías en el funcionamiento del tren a otros conductores de jornadas posteriores o al administrador para que avise a mantenimiento.

5.2.2.3 Pasajero

Los pasajeros son aquellos usuarios que solamente podrán interactuar con el sistema para la consulta de horarios y rutas de los trenes. Estos no podrán iniciar sesión en la aplicación.

5.2.3 Especificación de Casos de Uso

5.2.3.1 Usuario anónimo

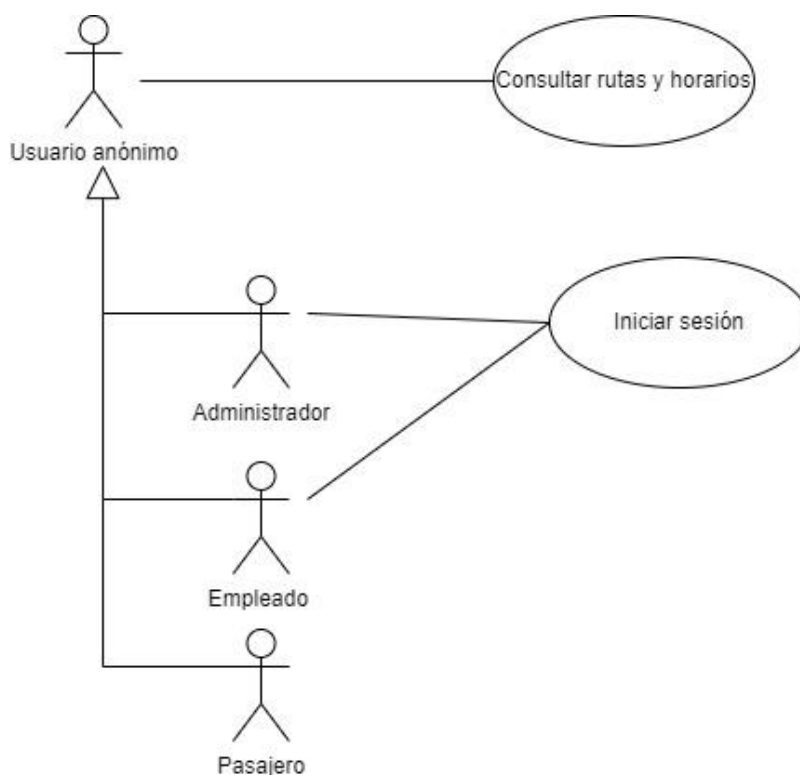


Ilustración 29. Diagrama de Casos de Uso. Usuario anónimo

Nombre del Caso de Uso
Iniciar sesión
Requisitos relacionados
RInicio1
Descripción
Los usuarios administradores y empleados pueden iniciar sesión en el sistema introduciendo el nombre de usuario y la contraseña. Se comprueba que el par introducido exista en la base de datos.

Tabla 18. Especificación Caso de Uso. Iniciar sesión

Nombre del Caso de Uso
Consultar rutas y horarios
Requisitos relacionados
RConsultar 1, RConsultar 2
Descripción
Los usuarios anónimos pueden consultar los horarios de los trenes indicando la estación de origen y la estación de destino. Además, pueden ver la localización de todas las estaciones en un mapa.

Tabla 19. Especificación Caso de Uso. Consultar rutas y horarios

5.2.3.2 Usuario administrador identificado

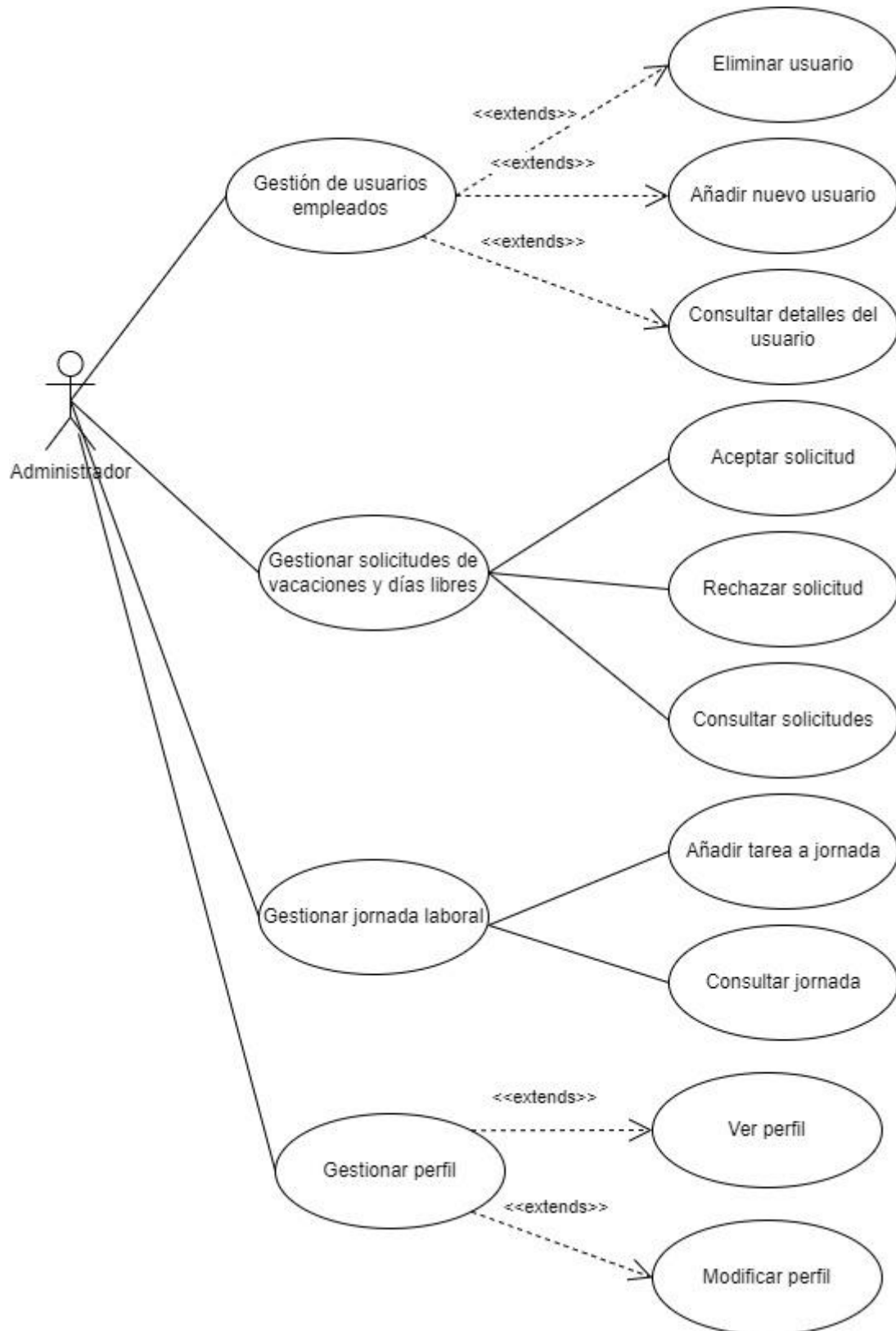


Ilustración 30. Diagrama de Casos de Uso. Usuario administrador identificado



Nombre del Caso de Uso
Añadir nuevo usuario
Requisitos relacionados
RGUsuario1.1.1
Descripción
El administrador es el encargado de la creación de un nuevo usuario aportando: <ul style="list-style-type: none">- Nombre de usuario- Nombre- Apellido- NIF/NIE- Email- Puesto de trabajo (Rol)

Tabla 20. Especificación Caso de Uso. Añadir nuevo usuario

Nombre del Caso de Uso
Gestionar solicitudes de vacaciones y días libres
Requisitos relacionados
RVerSol1, RVerSol2, RVerSol3
Descripción
Los usuarios administradores podrán ver, aceptar y rechazar las solicitudes de vacaciones y días libres que realizan los empleados. En caso de que el empleado disponga de días libres, el motivo de la solicitud sea coherente con la petición y su ausencia pueda ser cubierta por otro empleado, se le concederá el día; en caso contrario, se denegará la petición.

Tabla 21. Especificación Caso de Uso. Gestionar solicitudes de vacaciones y días libres

Nombre del Caso de Uso
Gestionar jornada laboral
Requisitos relacionados
RGJLaboral3
Descripción
Los usuarios administradores pueden asignar una jornada laboral a un empleado, añadiendo varias tareas a las mismas siempre y cuando la nueva tarea no coincida con ninguna existente en el mismo horario.

Tabla 22. Especificación Caso de Uso. Gestión jornada laboral

5.2.3.3 Usuario empleado identificado

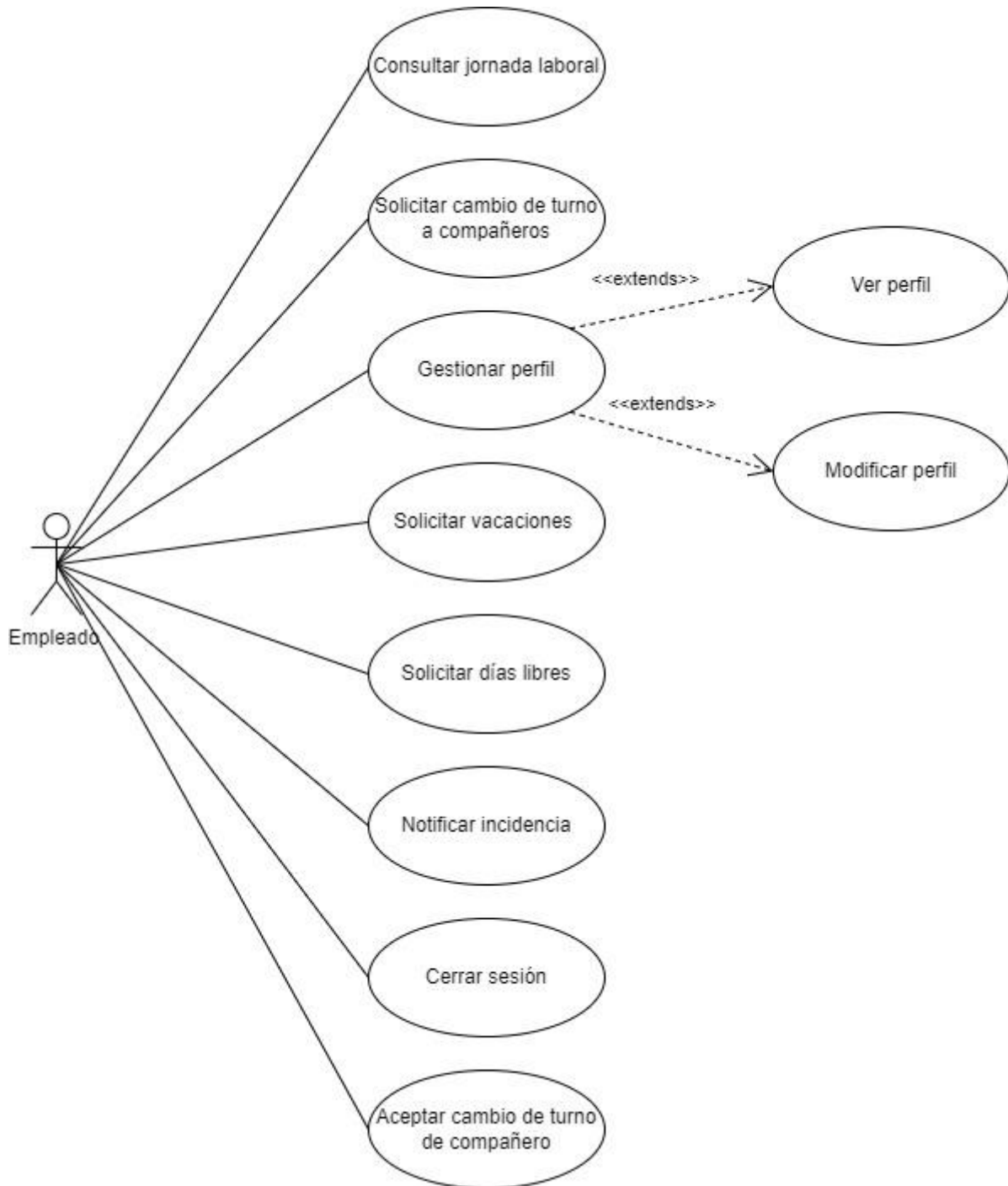


Ilustración 31. Diagrama de Casos de Uso. Usuario empleado identificado



Nombre del Caso de Uso
Solicitar cambio de jornada a compañero
Requisitos relacionados
RGSolicitudes1
Descripción
Mediante el portal de solicitudes de cambio de jornada, los empleados pueden enviar peticiones de cambio indicando: <ul style="list-style-type: none">- La fecha de la jornada que se desea tener libre (la que debe trabajar el nuevo empleado)- Fecha de la jornada por la que se cambia (el que tendrá libre el nuevo empleado)- El motivo del cambio

Tabla 23. Especificación Caso de Uso. Solicitar cambio de jornada a compañero

Nombre del Caso de Uso
Aceptar cambio de jornada de compañero
Requisitos relacionados
RGSolicitudes3
Descripción
Dada una petición de cambio de jornada que ha enviado un compañero, los usuarios empleados autenticados disponen de la posibilidad de aceptar dicho cambio si el día que se ofrece trabajar corresponde con uno de sus días libres, o bien sus jornadas son compatibles. Además, el empleado debe tener el mismo Rol que el empleado que ha realizado la solicitud.

Tabla 24. Especificación Caso de Uso. Aceptar cambio de jornada de compañero

Nombre del Caso de Uso
Solicitar vacaciones
Requisitos relacionados
RGVacaciones1
Descripción
Los empleados pueden solicitar vacaciones eligiendo uno de los periodos establecidos; esta solicitud quedará en estado pendiente hasta que sea resuelta por el administrador.

Tabla 25. Especificación Caso de Uso. Solicitar vacaciones



Nombre del Caso de Uso
Solicitar días libres
Requisitos relacionados
RGVacaciones2
Descripción
Los empleados pueden solicitar días libres que serán concedidos o rechazados una vez sean revisadas las peticiones por el administrador. Para la realización de estas solicitudes el usuario debe indicar: <ul style="list-style-type: none">- Fecha que desea seleccionar- Motivo por el cuál desea se solicita el día libre La solicitud se llevará a cabo siempre y cuando el empleado no haya consumido todos los días libres de los que dispone. La solicitud se creará con estado pendiente.

Tabla 26. Especificación Caso de Uso. Solicitar días libres

Nombre del Caso de Uso
Notificar y comprobar incidencias
Requisitos relacionados
RGJLaboral2
Descripción
Los empleados maquinistas pueden notificar incidencias o anomalías que detecten en los trenes para que se añadan al sistema y el resto de los empleados tengan constancia de ellas.

Tabla 27. Especificación Caso de Uso. Notificar incidencia

Nombre del Caso de Uso
Consultar jornada laboral
Requisitos relacionados
RGJLaboral1
Descripción
Los empleados pueden consultar su jornada laboral en un calendario mensual. Seleccionando un día del mismo pueden consultar la jornada laboral correspondiente. Esta consta de una o varias tareas con sus detalles. <ul style="list-style-type: none">- Hora de salida- Hora de finalización- Andén- Descripción- Ruta- Identificador del tren- Incidencias del tren

Tabla 28. Especificación Caso de Uso. Consultar jornada laboral



5.3 ASI 3: IDENTIFICACIÓN DE SUBSISTEMAS DE ANÁLISIS

En esta sección se analiza el sistema y se descompone en subsistemas más pequeños para poder hacer un análisis de manera más sencilla.

La aplicación está compuesta de cinco subsistemas que se detallan a continuación.

5.3.1 Descripción de los Subsistemas

5.3.1.1 *Gestión de usuarios*

El subsistema de la gestión de usuarios engloba todo aquello relacionado con la gestión del personal de la empresa, principalmente tareas llevadas a cabo por el administrador. Se compone de funcionalidades como el inicio de sesión, la creación de nuevos usuarios, eliminar empleados existentes y la consulta de información de los perfiles.

5.3.1.2 *Gestión jornada laboral*

El subsistema de gestión de la jornada laboral se compone de las tareas de consulta a través de un calendario mensual de todas las rutas que debe hacer el empleado en el mes, junto con el horario de cada una de las tareas que componen la jornada y toda la información relativa a la propia ruta.

5.3.1.3 *Gestión de vacaciones y días libres*

El sistema de gestión de vacaciones y días libres es el encargado de llevar a cabo todas las tareas de solicitud, aceptación/rechazo y consulta de vacaciones y días libres. Los empleados deben seleccionar un periodo de vacaciones entre los disponibles y pueden solicitar días libres. El administrador será el encargado de revisar las solicitudes y aceptarlas o denegarlas en función de disponibilidad.

5.3.1.4 *Portal solicitudes entre empleados*

El módulo del portal de solicitudes entre empleados está formado por aquellas tareas relacionadas con la solicitud de intercambios de jornadas entre los empleados. Por un lado, está el envío de una solicitud al portal por parte de un empleado, la consulta de la lista de todas las solicitudes, y por otro lado la parte de aceptación de una solicitud por parte de otro empleado. Además, incluye la parte de consulta de las propias solicitudes del empleado, tanto al portal, como días libres y vacaciones.

5.3.1.5 Gestión de información para viajeros

Este módulo está principalmente enfocado a los pasajeros de la compañía, aunque también pueden intervenir los propios empleados. Está formado por la tarea de consulta de horarios y rutas de Renfe sin necesidad de iniciar sesión en la aplicación.

5.3.2 Descripción de los Interfaces entre Subsistemas

En este apartado se describe cómo será la comunicación entre los cinco subsistemas identificados en el apartado anterior.

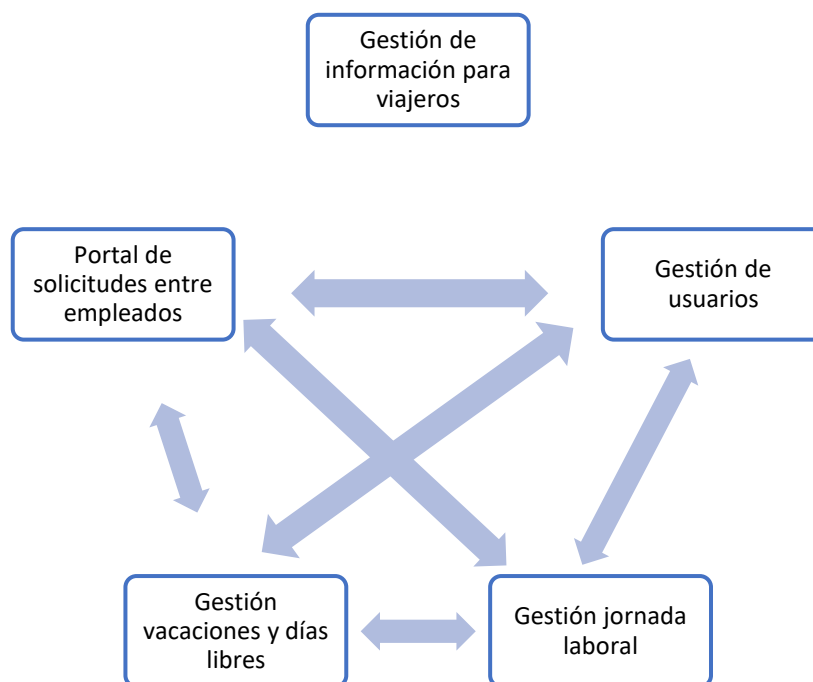


Ilustración 32. Descripción de los Interfaces entre Subsistemas

El subsistema de gestión de información para viajeros es el único que no está relacionado con ninguno de los otros subsistemas de la aplicación. Sin embargo, el resto se comunican todos entre sí. Estas comunicaciones se realizan a través de la invocación a métodos.

5.4 ASI 4: ANÁLISIS DE LOS CASOS DE USO

En este apartado se realizará el análisis de los Casos de Uso, para ello se hará uso de escenarios que describan las interacciones entre los usuarios y el sistema, permitiendo de esta forma concretar el comportamiento deseado del sistema.

5.4.1 Caso de Uso 1: Iniciar de Sesión

INICIAR SESIÓN	
PRECONDICIONES	El usuario debe estar en estado no autenticado (no debe haber iniciado sesión en el sistema) El usuario debe ser un empleado
POSCONDICIONES	El usuario pasará a estado autenticado. El usuario tendrá acceso a toda la parte de la aplicación que corresponda con su rol.
ACTORES	Usuario no autenticado
DESCRIPCIÓN	<ol style="list-style-type: none">1. El sistema muestra la pantalla de inicio de sesión.2. El usuario introduce su nombre de usuario de usuario y contraseña3. El sistema comprueba que existe en la BD un usuario cuyos datos coinciden con los introducidos.4. El usuario entra correctamente en el sistema.
VARIACIONES (ESCENARIOS SECUNDARIOS)	Escenario alternativo 1: No existe en la BD ningún usuario cuyo nombre de usuario coincida con el introducido. Se muestra mensaje de error y se vuelve al paso 2 del escenario principal. Escenario alternativo 2: La contraseña almacenada en la BD correspondiente con ese nombre de usuario no coincide con la introducida. Se muestra mensaje de error y se vuelve al paso 2 del escenario principal. Escenario alternativo 3: El usuario deja alguno de los campos vacío. Se muestra mensaje de error y se vuelve al paso 2 del escenario principal.
EXCEPCIONES	La base de datos no responde, por lo que no se pueden corroborar los datos de inicio y el usuario no se puede autenticar.

Tabla 29. Análisis Casos de Uso. Iniciar sesión

5.4.2 Caso de Uso 2: Consultar rutas y horarios

CONSULTAR RUTAS Y HORARIOS	
PRECONDICIONES	No
POSCONDICIONES	El usuario tendrá los horarios de salida para la ruta seleccionada.
ACTORES	Usuario no autenticado Usuario autenticado
DESCRIPCIÓN	<ol style="list-style-type: none">1. El usuario selecciona el botón de consulta de rutas y horarios2. El usuario comprueba el mapa de estaciones.3. El usuario indica la estación de origen de la ruta4. El usuario indica la estación de destino de la ruta5. El sistema muestra la lista de horarios6. El sistema muestra en el mapa las estaciones de origen y destino.
VARIACIONES (ESCENARIOS SECUNDARIOS)	<p>Escenario alternativo 1: El usuario no selecciona ninguna estación de origen/destino. Se muestra un aviso al usuario de que debe seleccionar alguna estación y se vuelve al paso 2 del escenario principal</p> <p>Escenario alternativo 2: El usuario selecciona como origen y destino dos estaciones entre las que no existe ruta. Se muestra un aviso al usuario indicando que entre las estaciones seleccionadas no existe ruta y se vuelve al paso 3 del escenario principal.</p>
EXCEPCIONES	La base de datos no responde, por lo que no se puede realizar la consulta de horario

Tabla 30. Análisis Casos de Uso. Consultar rutas y horarios

5.4.3 Caso de uso 3: Añadir nuevo usuario

AÑADIR NUEVO USUARIO	
PRECONDICIONES	El usuario debe estar autenticado (debe haber iniciado sesión en el sistema) y debe ser un administrador.
POSCONDICIONES	Se añadirá un nuevo usuario a la BD.
ACTORES	Usuario autenticado con rol administrador
DESCRIPCIÓN	<ol style="list-style-type: none">1. El administrador selecciona la opción de añadir un nuevo usuario al sistema.2. El sistema muestra la pantalla de añadir nuevo usuario.3. El administrador rellena el formulario.4. Se comprueba que los datos introducidos sean correctos.5. Un nuevo usuario se añade a la BD.

VARIACIONES (ESCENARIOS SECUNDARIOS)	<p>Escenario alternativo 1: Ya existe un usuario con ese nombre de usuario, email o DNI en la BD. Se muestra mensaje de error y se vuelve al paso 3 del escenario principal.</p> <p>Escenario alternativo 2: Alguno de los valores introducidos no es válido. Se muestra mensaje de error y se vuelve al paso 2 del escenario principal.</p> <p>Escenario alternativo 3: El usuario deja alguno de los campos obligatorios vacío. Se muestra mensaje de error y se vuelve al paso 3 del escenario principal.</p>
EXCEPCIONES	La base de datos no responde, por lo que no el usuario no puede ser creado.

Tabla 31. Análisis Casos de Uso. Añadir nuevo usuario

5.4.4 Caso de uso 4: Gestionar solicitudes de vacaciones y días libres (administrador)

GESTIONAR SOLICITUDES DE VACACIONES Y DÍAS LIBRES	
PRECONDICIONES	El usuario debe estar autenticado (debe haber iniciado sesión en el sistema) y debe ser un administrador.
POSCONDICIONES	La solicitud pasará a un estado “ACEPTADA” o “RECHAZADA”
ACTORES	Usuario autenticado con rol administrador
DESCRIPCIÓN	<ol style="list-style-type: none"> 1. El administrador selecciona la opción de ver las solicitudes 2. El administrador comprueba la disponibilidad para la fecha seleccionada 3. El administrador acepta la petición 4. La solicitud pasa a estado ACEPTADA y se elimina de la lista
VARIACIONES (ESCENARIOS SECUNDARIOS)	<p>Escenario alternativo 1: El administrador rechaza la petición. La petición pasa a estado RECHAZADA y se elimina de la lista.</p>
EXCEPCIONES	-

Tabla 32. Análisis Casos de Uso. Gestionar solicitud de vacaciones y días libres

5.4.5 Caso de uso 5: Gestionar jornada laboral

GESTIONAR JORNADA LABORAL	
PRECONDICIONES	El usuario debe estar autenticado (debe haber iniciado sesión en el sistema) y debe ser un administrador.
POSCONDICIONES	-
ACTORES	Usuario autenticado con rol administrador
DESCRIPCIÓN	<ol style="list-style-type: none">1. El administrador selecciona la opción de gestionar la jornada laboral2. Selecciona el día que desea organizar3. Selecciona el empleado para el cuál se quiere gestionar la jornada4. Añade una tarea rellenando toda la información solicitada.5. La nueva tarea se añade a la base de datos
VARIACIONES (ESCENARIOS SECUNDARIOS)	<p>Escenario alternativo 1: Ya existe una tarea para el horario seleccionado. Se muestra un mensaje de error y se vuelve al paso 4 del escenario principal</p> <p>Escenario alternativo 2: Se seleccionan como origen y destino de la tarea dos estaciones entra las cuales no existe ruta. Se muestra un mensaje de error y se vuelve al paso 4 del escenario principal</p> <p>Escenario alternativo 3: Se selecciona una hora de fin anterior a la hora de inicio. Se muestra un mensaje de error y se vuelve al paso 4 del escenario principal.</p>
EXCEPCIONES	-

Tabla 33. Análisis Casos de Uso. Gestionar jornada laboral

5.4.6 Caso de uso 6: Solicitar cambio de jornada a compañero

SOLICITAR CAMBIO DE JORNADA	
PRECONDICIONES	El usuario debe estar autenticado (debe haber iniciado sesión en el sistema) y debe ser un usuario empleado. El empleado debe seleccionar una de las tareas asignadas y el día libre que ofrezca no debe tener ninguna jornada asignada
POSCONDICIONES	-
ACTORES	Usuario autenticado con rol empleado
DESCRIPCIÓN	<ol style="list-style-type: none">1. El usuario selecciona la opción de portal de solicitudes2. El usuario selecciona crear una nueva solicitud3. Selecciona la jornada que desea cambiar4. Selecciona el día que desea cubrir5. Añade el motivo del cambio6. La nueva solicitud se añade a la base de datos con estado PENDIENTE

VARIACIONES (ESCENARIOS SECUNDARIOS)	<p>Escenario alternativo 1: El empleado no tiene una jornada asignada para el día que desea cambiar. Se muestra un mensaje de error y se vuelve al paso 3 del escenario principal</p> <p>Escenario alternativo 2: El empleado tiene una jornada asignada para el día que desea cubrir. Se muestra un mensaje de error y se vuelve al paso 3 del escenario principal</p> <p>Escenario alternativo 3: El empleado ya tiene una solicitud creada para la fecha que desea cambiar. Se muestra un mensaje de error y se vuelve al paso 3 del escenario principal</p>
EXCEPCIONES	La base de datos no responde, por lo que la solicitud no se puede llevar a cabo.

Tabla 34. Análisis Casos de Uso. Solicitar cambio de jornada

5.4.7 Caso de uso 7: Aceptar cambio de jornada

ACEPTAR CAMBIO DE JORNADA	
PRECONDICIONES	<p>El usuario debe estar autenticado (debe haber iniciado sesión en el sistema) y debe ser un usuario empleado.</p> <p>El empleado no debe tener ninguna tarea que coincida en horario con la nueva tarea, y el nuevo día de descanso debe tener una tarea asignada.</p>
POSCONDICIONES	-
ACTORES	Usuario autenticado con rol empleado
DESCRIPCIÓN	<ol style="list-style-type: none"> 1. El usuario selecciona la opción de portal de solicitudes 2. El usuario acepta la petición 3. La jornada de ambos se actualiza 4. La solicitud pasa a estado REASIGNADA
VARIACIONES (ESCENARIOS SECUNDARIOS)	-
EXCEPCIONES	-

Tabla 35. Análisis Casos de Uso. Aceptar cambio de jornada

5.4.8 Caso de uso 8: Solicitar vacaciones

SOLICITAR VACACIONES	
PRECONDICIONES	El usuario debe estar autenticado (debe haber iniciado sesión en el sistema) y debe ser un usuario empleado. El empleado no debe haber seleccionado ningún periodo de vacaciones para la misma época (verano o invierno) con anterioridad.
POSCONDICIONES	-
ACTORES	Usuario autenticado con rol empleado
DESCRIPCIÓN	<ol style="list-style-type: none">1. El usuario selecciona la opción de solicitar vacaciones2. Elige un periodo de vacaciones entre los disponibles
VARIACIONES (ESCENARIOS SECUNDARIOS)	Escenario alternativo 1: El usuario intenta solicitar las vacaciones sin haber seleccionado un periodo concreto. Se muestra un mensaje de error y se vuelve al paso 2 del escenario principal. Escenario alternativo 2: El usuario intenta solicitar las vacaciones para un periodo en el cuál ya tiene otra solicitud con estado "RECHAZADA". Se muestra un mensaje de error y se vuelve al paso 2 del escenario principal.
EXCEPCIONES	-

Tabla 36. Análisis Casos de Uso. Solicitar vacaciones

5.4.9 Caso de uso 9: Solicitar días libres

SOLICITAR DÍAS LIBRES	
PRECONDICIONES	El usuario debe estar autenticado (debe haber iniciado sesión en el sistema) y debe ser un usuario empleado. El empleado debe disponer de días libres
POSCONDICIONES	-
ACTORES	Usuario autenticado con rol empleado
DESCRIPCIÓN	<ol style="list-style-type: none">1. El usuario selecciona la opción de solicitar días libres2. Selecciona el día deseado3. Selecciona el motivo de la usencia4. Envía la petición5. Se añade una nueva solicitud a la base de datos con estado pendiente
VARIACIONES (ESCENARIOS SECUNDARIOS)	Escenario alternativo 1: El usuario intenta hacer una solicitud sin haber cumplimentado todos los campos del formulario. Se muestra un mensaje de error y se vuelve al paso 2 del escenario principal. Escenario alternativo 2: Ya existe una solicitud para ese día realizada por ese empleado en el sistema.

	Se muestra un mensaje de error y se vuelve al paso 2 del escenario principal.
EXCEPCIONES	-

Tabla 37. Análisis Casos de Uso. Solicitar días libres

5.4.10 Caso de uso 10: Notificar y comprobar incidencias

NOTIFICAR Y COMPROBAR INCIDENCIAS	
PRECONDICIONES	El usuario debe estar autenticado (debe haber iniciado sesión en el sistema) y debe ser un usuario empleado.
POSCONDICIONES	-
ACTORES	Usuario autenticado con rol empleado
DESCRIPCIÓN	<ol style="list-style-type: none">1. El usuario selecciona la opción de consultar jornada2. Se selecciona un día del calendario con una jornada asignada3. Selecciona una de las tareas que componen la jornada4. Inspecciona los detalles de la tarea5. Comprueba las incidencias del tren6. Añade una nueva incidencia7. La incidencia de registra en la base de datos
VARIACIONES (ESCENARIOS SECUNDARIOS)	-
EXCEPCIONES	La base de datos no responde, por lo que no se puede consultar la jornada e incidencias de sus tareas

Tabla 38. Análisis Casos de Uso. Notificar y comprobar incidencias

5.4.11 Caso de uso 11: Consultar jornada laboral

CONSULTAR JORNADA LABORAL	
PRECONDICIONES	El usuario debe estar autenticado (debe haber iniciado sesión en el sistema) y debe ser un usuario empleado.
POSCONDICIONES	-
ACTORES	Usuario autenticado con rol empleado
DESCRIPCIÓN	<ol style="list-style-type: none">1. El usuario selecciona la opción de consultar jornada2. Selecciona un día del calendario mensual3. Comprueba las tareas que conforman la jornada4. Inspecciona los detalles de una de las tareas
VARIACIONES (ESCENARIOS SECUNDARIOS)	Escenario alternativo 1: El usuario selecciona un día que no tiene jornada laboral asignada. Se muestra un mensaje notificando que no hay jornada aún asignada para el día seleccionado y se vuelve al paso 2 del escenario principal
EXCEPCIONES	La base de datos no responde, por lo que no se puede consultar la jornada.

Tabla 39. Análisis Casos de Uso. Consultar jornada laboral



5.5 ASI 5: ANÁLISIS DE CLASES

En este apartado se identifican las clases del sistema, teniendo como referencia los casos de uso especificados en el apartado anterior. Este análisis inicial servirá como base para el esquema completo que se desarrollará en la fase de diseño.

Como se puede ver en el diagrama siguiente existirán 15 clases principales, cada una de ellas representadas en una entidad.

Además, habrá controladores encargados de gestionar las peticiones que el cliente realiza; para proporcionar una respuesta se comunicarán con los servicios. Existirán seis controladores que recojan estas peticiones (EmpleadoController, EstacionController, JornadaController, RutaController, SolicitudController y TrenController).

Los servicios son los encargados de la lógica de negocio, se comunican con los repositorios para obtener los datos y proporcionárselos a los controladores. Los servicios que se crearán corresponderán con los controladores, añadiendo tres adicionales para la gestión de incidencias (IncidenciasService), para la gestión de las tareas (TareaService) y para la gestión de las “TareaEstacion”; es decir, las estaciones que componen una tarea.

En último lugar están los repositorios son los encargados de las conexiones con la base de datos. Para cada una de las entidades existirá un repositorio correspondiente que gestionará esta conexión.

Como se ha mencionado anteriormente tanto los controladores, como los servicios y los repositorios se detallarán en la fase de diseño.

5.5.1 Diagrama de Clases

A continuación, se muestra un diagrama de clases global que muestra las relaciones que hay entre todas ellas.

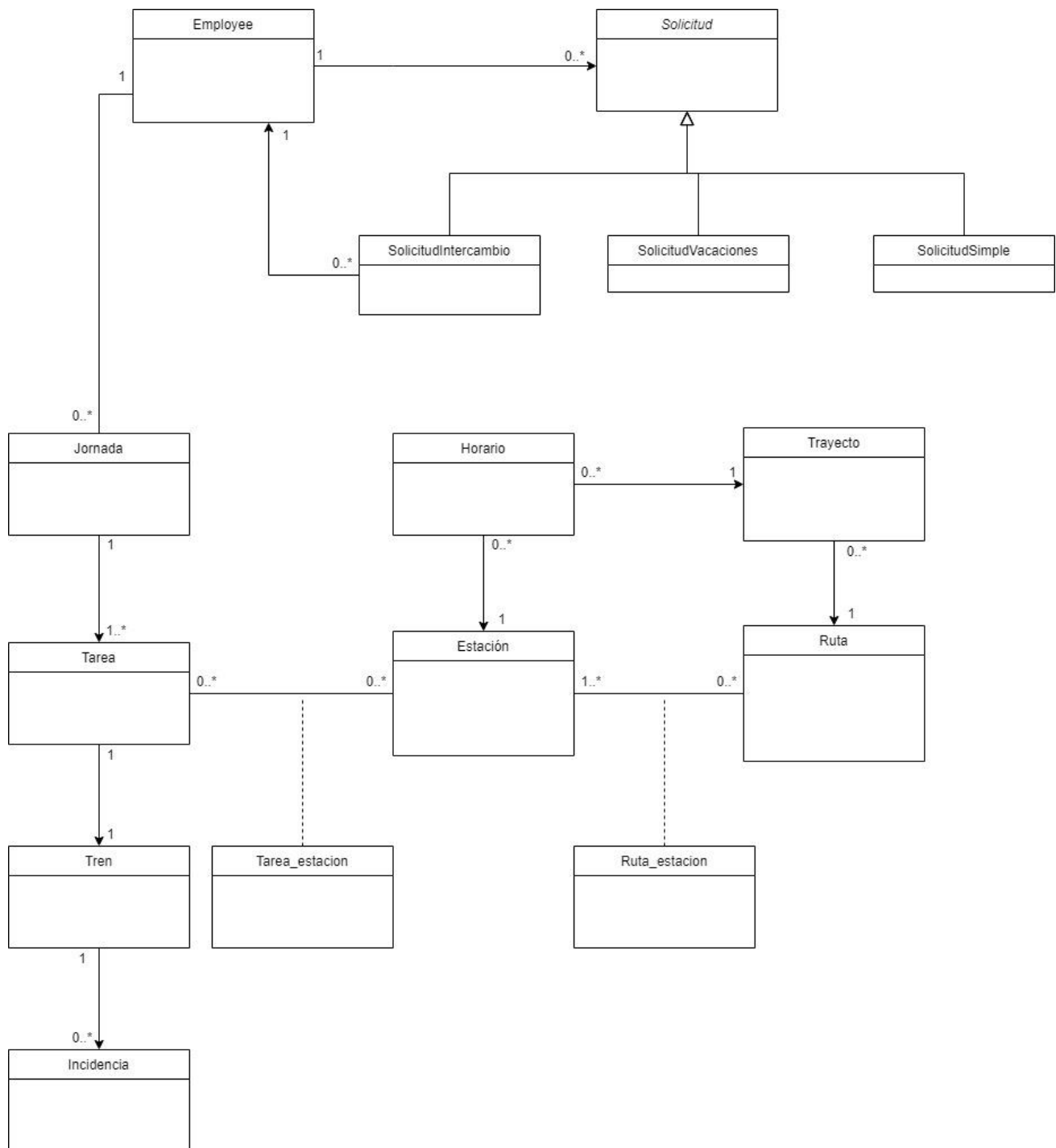


Ilustración 33. Análisis. Diagrama de clases

5.5.2 Descripción de las Clases

5.5.2.1 Entidades

En este apartado se procede a detallar cada una de las clases del diagrama de la Ilustración 33.

5.5.2.1.1 Subsistema de gestión de usuarios

Empleado
Descripción
Representa un empleado de la compañía
Atributos propuestos
<ul style="list-style-type: none">- id: Identificador único- username: Nombre de usuario- name: Nombre del empleado- surname: Apellido del empleado- role: Rol del empleado- password: Contraseña- numeroDiasLibres: Número de días libres disponibles

Tabla 40. Descripción de la clase Empleado

5.5.2.1.2 Subsistema de gestión jornada laboral

Jornada
Descripción
Representa una jornada laboral correspondiente a un día concreto
Atributos propuestos
<ul style="list-style-type: none">- id: Identificador único- date: Fecha de la jornada

Tabla 41. Descripción de la clase Jornada

Tarea
Descripción
Representa una tarea dentro de una jornada laboral
Atributos propuestos
<ul style="list-style-type: none">- id: Identificador único- descripcion: Descripción de la tarea a realizar- horaSalida: Hora de comienzo de la tarea- horaFin: Hora de finalización de la tarea- anden: Anden en el que se encuentra situado el tren

Tabla 42. Descripción de la clase Tarea

Tren
Descripción
Representa un tren de la compañía
Atributos propuestos
<ul style="list-style-type: none">- id: Identificador único

Tabla 43. Descripción de la clase Tren



Incidencia

Descripción

Representa una incidencia de un tren

Atributos propuestos

- **id:** Identificador único
- **estado:** Estado en el que se encuentra la incidencia
- **descripción:** Descripción de la incidencia

Tabla 44. Descripción de la clase Incidencia

Tarea_estacion

Descripción

Representa el enlace entre una tarea y la estación de origen y de destino de la misma

Atributos propuestos

- **id:** Identificador único
- **situación:** Enumerado que indica si la estación es el inicio o el final de la ruta de la tarea

Tabla 45. Descripción de la clase Tarea_estacion

5.5.2.1.3 Subsistema de gestión de vacaciones y días libres

Solicitud

Descripción

Representa una solicitud general

Atributos propuestos

- **id:** Identificador único
- **estado:** Estado en el que se encuentra la solicitud
- **motivo:** Motivo de la ausencia
- **fecha:** Fecha

Tabla 46. Descripción de la clase Solicitud

SolicitudSimple

Descripción

Representa una solicitud de día libre o de vacaciones

Atributos propuestos

- **id:** Identificador único

Tabla 47. Descripción de la clase SolicitudSimple

SolicitudVacaciones

Descripción

Representa una solicitud de día libre o de vacaciones

Atributos propuestos

- **id:** Identificador único
- **fechaFinVacaciones:** fecha en la que finalizan las vacaciones

Tabla 48. Descripción de la clase SolicitudVacaciones

5.5.2.1.4 Subsistema del portal de solicitudes

SolicitudIntercambio
Descripción
Representa una solicitud de intercambio de jornada
Atributos propuestos
<ul style="list-style-type: none">- id: Identificador único- fechaDescanso: Fecha en la que actualmente se descansa

Tabla 49. Descripción de la clase SolicitudIntercambio

5.5.2.1.5 Subsistema de gestión de información para viajeros

Horario
Descripción
Representa los horarios
Atributos propuestos
<ul style="list-style-type: none">- id: Identificador único- hora_llegada: Hora de llegada- hora_salida: Hora de salida

Tabla 50. Descripción de la clase Horario

Estacion
Descripción
Representa las estaciones
Atributos propuestos
<ul style="list-style-type: none">- id: Identificador único- nombre: Nombre de la estación- latitud: Latitud de la estación- longitud: Longitud de la estación

Tabla 51. Descripción de la clase Estacion

Ruta
Descripción
Representa una línea de ruta
Atributos propuestos
<ul style="list-style-type: none">- id: Identificador único- ruta_corto: Nombre de ruta reducido- ruta_largo: Nombre de ruta completo

Tabla 52. Descripción de la clase Ruta



Ruta_estacion

Descripción

Representa el enlace entre la ruta y las paradas y permite la relación entre las mismas

Atributos propuestos

- **id**: Identificador único
- **orderParada**: Orden de la estación en la ruta

Tabla 53. Descripción de la clase Ruta_estacion

Trayecto

Descripción

Representa un trayecto en una ruta

Atributos propuestos

- **id**: Identificador único

Tabla 54. Descripción de la clase Trayecto

5.6 ASI 8: DEFINICIÓN DE INTERFACES DE USUARIO

En este apartado se definirán de manera general las interfaces de usuarios de la aplicación. Esta será una primera definición, propia de la etapa de análisis, que probablemente posteriormente se modifique. Permite comprender cómo será la aplicación y cómo podrán interactuar los usuarios con ella.

A continuación, se muestran las futuras pantallas de la aplicación dividiéndolas según los roles de los usuarios. En primer lugar, se muestra una pantalla de “Home” o bienvenida, a continuación, las pantallas para los usuarios identificados como empleados normales y por último las pantallas para los usuarios administradores.

5.6.1 Descripción de la Interfaz

5.6.1.1 Esquema general de la aplicación

En este apartado se muestra un esquema general de la aplicación con la disposición que tendrán los componentes y las diferentes zonas de trabajo de las que consta. La barra de navegación y el pie de página serán estáticos en todas las ventanas de la aplicación; mientras que será el contenedor principal lo único que se verá modificado.

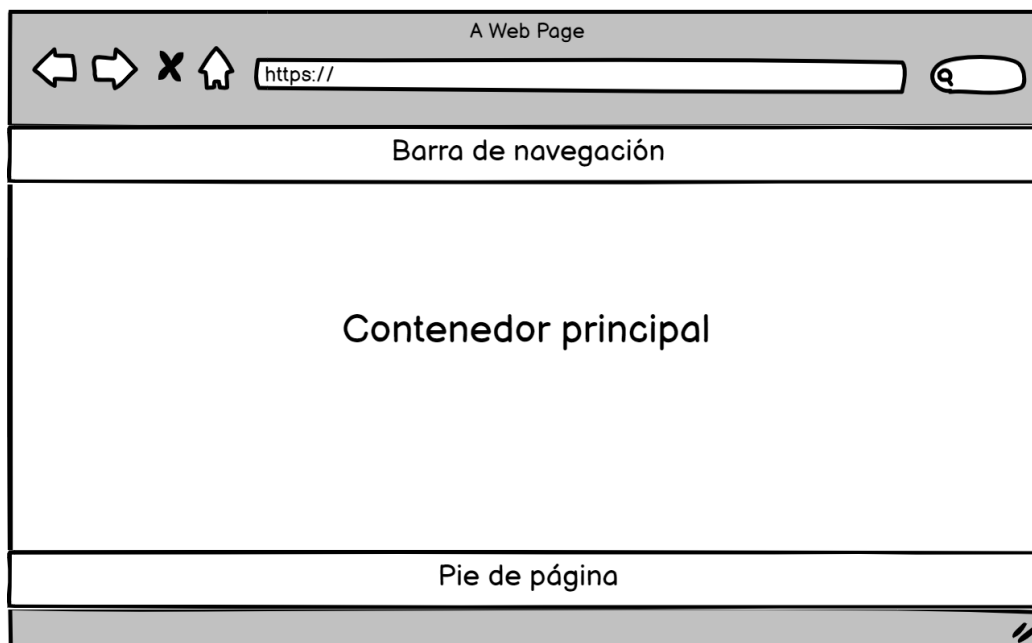


Ilustración 34. Interfaz esquema general

5.6.1.2 Prototipos usuario no identificado

5.6.1.2.1 Prototipo pantalla principal para usuario no identificado

Una vez se entre en la página, la visión que tendrá el usuario es la de la pantalla representada a continuación. Una vez ahí, podrá decidir si inicia sesión o si consulta los horarios y las rutas.

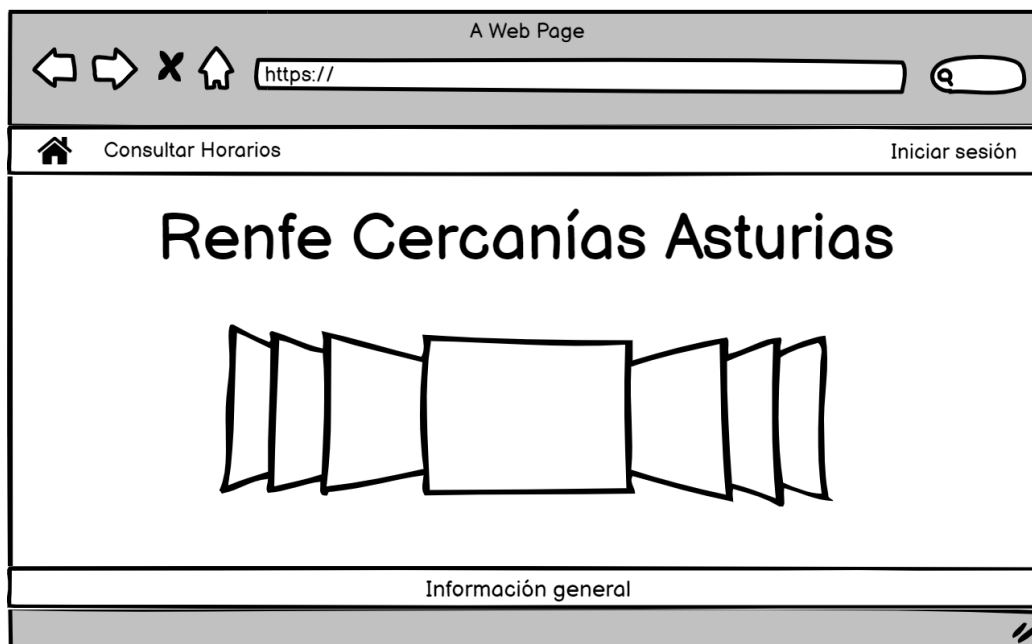


Ilustración 35. Prototipo pantalla principal

5.6.1.2.2 Prototipo consultar horarios



Ilustración 36. Prototipo consultar horarios 1.

En esta primera pantalla del prototipo de “consultar horarios” el mapa señala todos los puntos donde hay una estación de cercanías. Una vez el usuario selecciona la ruta que desea y pulsa el botón “Consultar” se pasa a la siguiente pantalla y en el mapa únicamente se muestran aquellas estaciones correspondientes al recorrido.

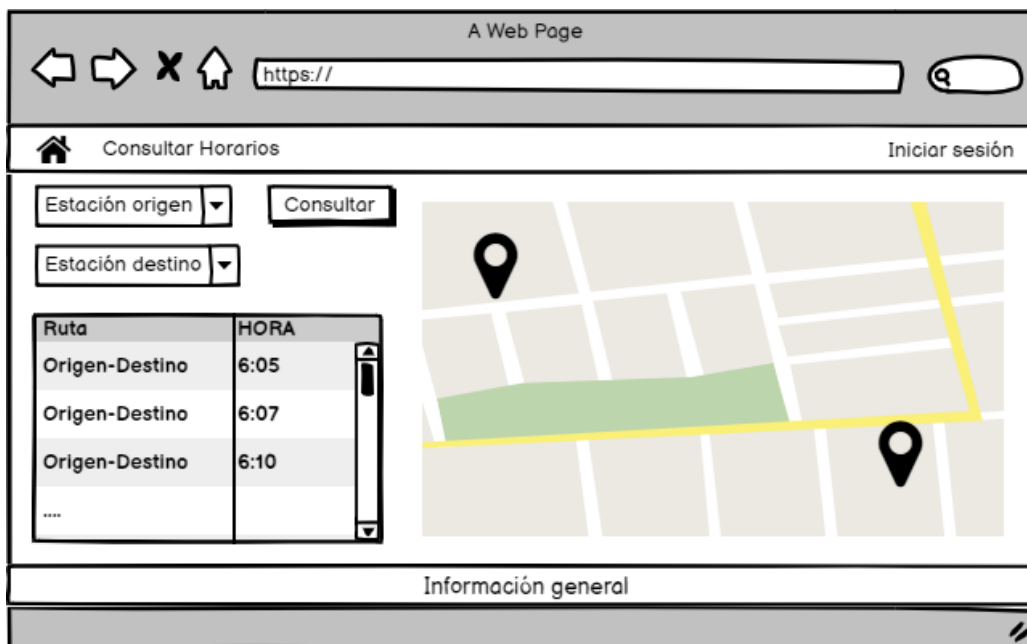


Ilustración 37. Prototipo consultar horario 2

5.6.1.2.3 Prototipo iniciar sesión

Los usuarios empleados podrán iniciar sesión en la aplicación a través del botón “Iniciar sesión” este les redirigirá a la pantalla mostrada a continuación. Para poder iniciar sesión en la aplicación únicamente habrá que proporcionar el nombre de usuario y la contraseña.

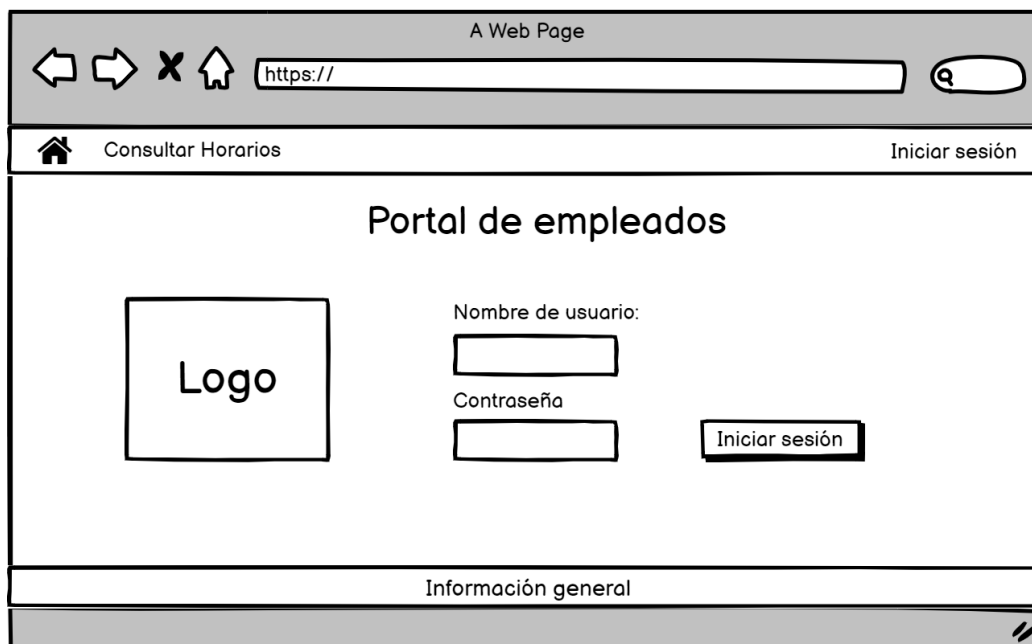


Ilustración 38. Prototipo iniciar sesión

5.6.1.3 Prototipos usuario empleado trabajador identificado

Una vez el usuario ha iniciado sesión como empleado tendrá acceso a las siguientes pantallas:

5.6.1.3.1 Prototipo consultar jornada laboral

Desplegando el menú de Gestión de jornada y seleccionando la opción de consultar jornada el usuario accederá a la siguiente página:

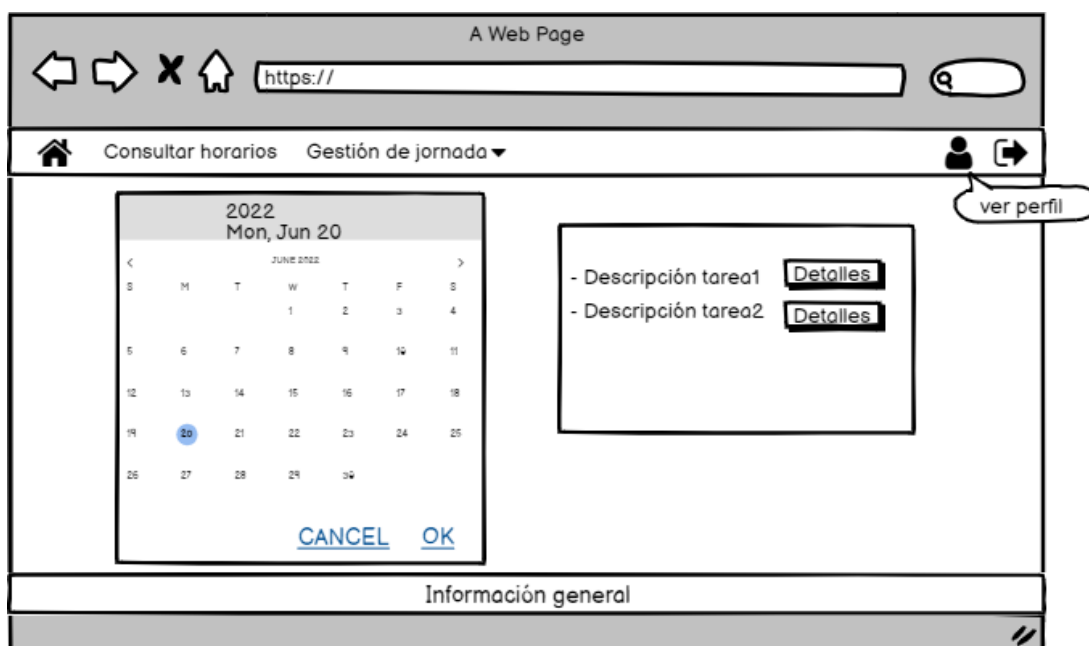


Ilustración 39. Prototipo consultar jornada laboral

Cuando el empleado pulsa el botón de “Detalles” para consultar las especificaciones concretas de una tarea se muestra la ventana emergente que se expone en el siguiente punto.

5.6.1.3.2 Prototipo detalles de la tarea

Detalles tarea

Descripción	Descripción de la tarea
Horario	6:00 - 7:15
Ruta	Avilés - Oviedo
Andén	1
Id tren	153
Incidencias	<div style="border: 1px solid black; padding: 2px;">Incidencia 1 Incidencia 2 Incidencia 3</div> <div style="border: 1px solid black; padding: 2px; margin-top: 5px;">+ Nueva incidencia</div>

añadir nueva incidencia

Ilustración 40. Prototipo detalles de una tarea

En esta ventana emergente se podrán ver los detalles de una tarea concreta y además, el usuario podrá indicar nuevas incidencias que haya detectado en el tren.

5.6.1.3.3 Prototipo solicitar vacaciones y días libres

A Web Page

https://

Consultar horarios Gestión de jornada

Solicitar día libre Solicitar vacaciones

	Verano	Invierno
<input type="radio"/>	01/06/2022 - 15/06/2022	21/12/2022 - 09/01/2023
<input type="radio"/>	01/07/2022 - 15/07/2022	01/10/2022 - 20/10/2022
<input type="radio"/>	16/06/2022 - 30/06/2022	01/12/2022 - 20/12/2022
<input type="radio"/>	16/07/2022 - 31/07/2022	21/10/2022 - 08/11/2022
<input type="radio"/>	01/09/2022 - 15/09/2022	12/05/2022 - 31/05/2022

Solicitar

Información general

Ilustración 41. Prototipo solicitar vacaciones

La ventana anterior engloba tanto la funcionalidad de solicitar vacaciones como la de seleccionar días libres. En la pestaña mostrada podemos ver la solicitud de vacaciones; seleccionando un periodo entre los disponibles se asignarán las mismas. Por otra parte, en la pestaña de solicitar día libre únicamente se indicará el día a solicitar y el motivo de la ausencia.

5.6.1.3.4 Prototipo portal solicitudes

En la siguiente imagen se muestra la pestaña de “todas las solicitudes”, la pestaña de “mis solicitudes” será similar, pero en lugar de una acción a realizar se indicará el estado de la solicitud.

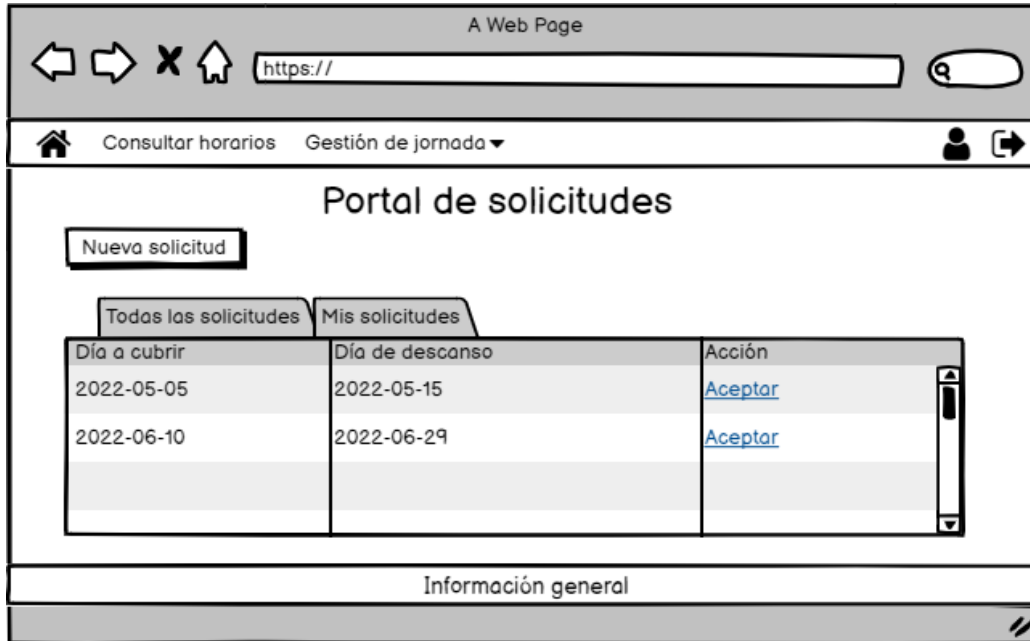


Ilustración 42. Prototipo portal de solicitudes

5.6.1.3.5 Prototipo gestionar perfil

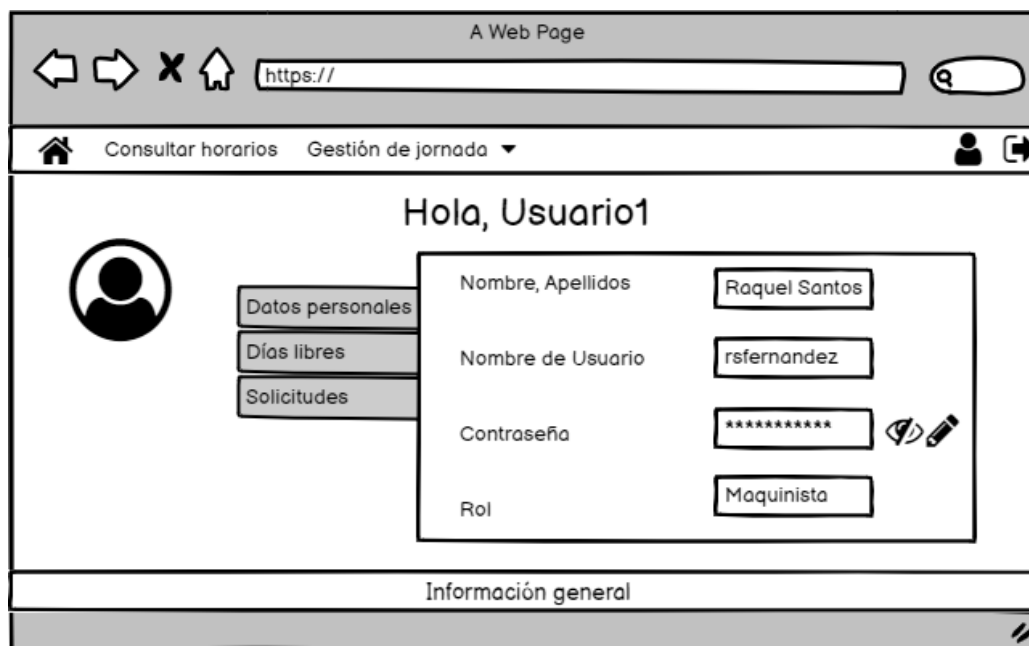


Ilustración 43. Prototipo gestionar perfil

La ventana de gestionar el perfil del usuario incluye un componente con tres pestañas, en primer lugar, la relativa a los datos personales, después la de los días libres, donde se puede ver el número de días libres consumidos y restantes y, por último, la de las solicitudes que al igual que la anterior, incluye el número de solicitudes realizadas y restantes.

5.6.1.4 Prototipos usuario administrador identificado

5.6.1.4.1 Prototipo gestionar usuarios y jornada laboral



Ilustración 44. Prototipo gestionar usuarios y jornada

5.6.1.4.2 Prototipo gestionar solicitudes

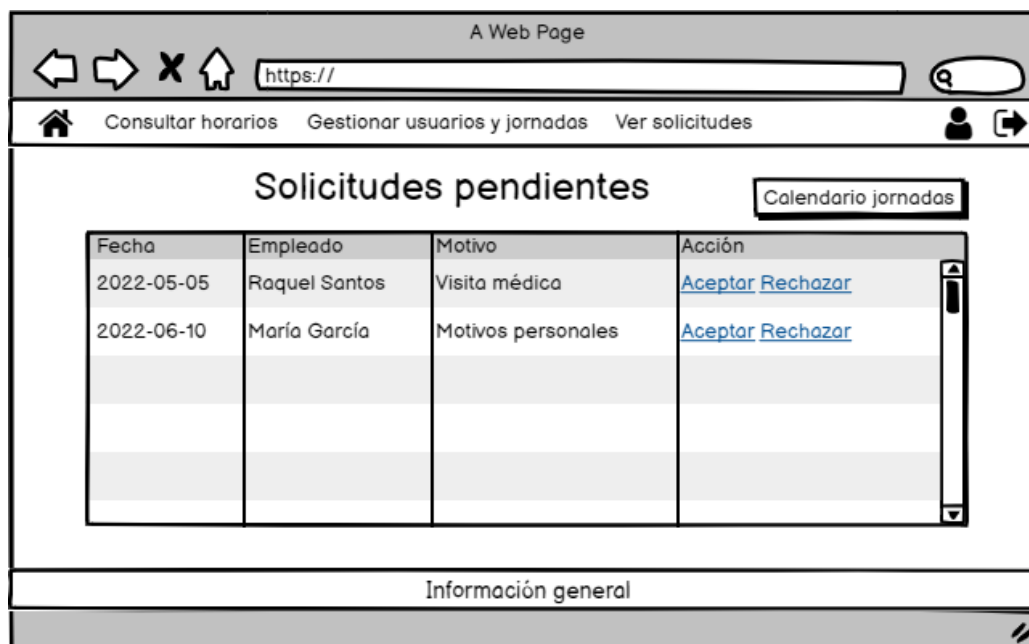


Ilustración 45. Prototipo gestionar solicitudes

5.6.2 Descripción del Comportamiento de la Interfaz

Las validaciones de datos en los formularios de entrada se realizarán para comprobar tanto el formato correcto como los valores adecuados. Además, se comprobará que aquellos datos de carácter obligatorio sean cubiertos.

Algunas de estas validaciones se llevan a cabo en la parte del cliente y otras en el servidor. En caso de error, este será notificado al usuario con un formato comprensible.

5.6.3 Diagrama de Navegabilidad

En este apartado se explica la navegación que los usuarios, dependiendo de si son empleados o no, pueden hacer en la aplicación.

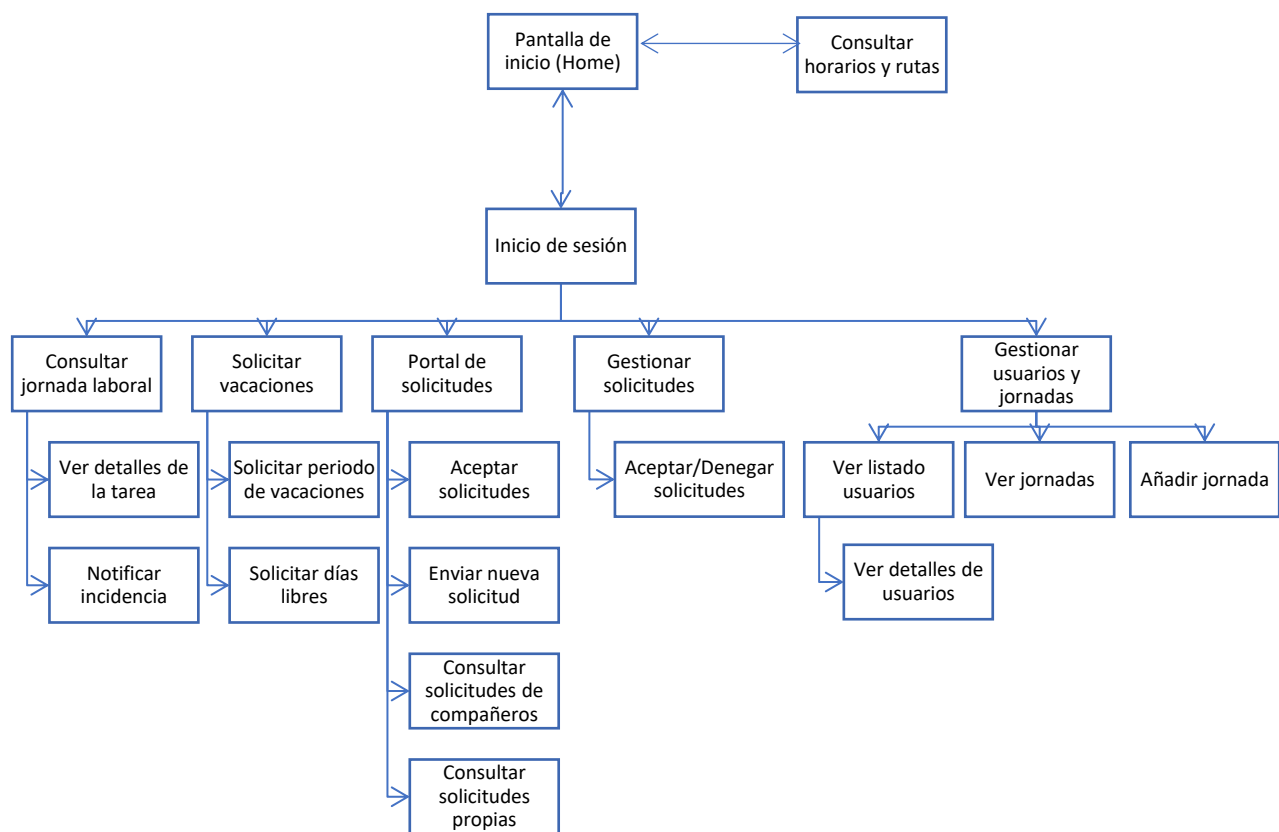


Ilustración 46. Diagrama de navegabilidad



5.7 ASI 10: ESPECIFICACIÓN DEL PLAN DE PRUEBAS

En este apartado se crea y diseña el plan de pruebas de la aplicación, que permiten detectar errores y corregirlos en la fase de implementación. Además, estas pruebas permiten comprobar la usabilidad de la aplicación.

5.7.1 Pruebas unitarias

Las pruebas unitarias permiten comprobar el correcto funcionamiento de funcionalidades concretas. Se realizan en la parte del servidor y, por lo tanto, comprueban la lógica de negocio de la aplicación.

Se realizarán a la vez que el desarrollo como se muestra en la *Planificación inicial*, esto permite que los errores posteriores se reduzcan notablemente.

Las pruebas unitarias se realizan a partir de los casos de uso y los escenarios y clases detallados anteriormente.

5.7.2 Pruebas de integración

Las pruebas de integración permiten comprobar el correcto funcionamiento de grupos de componentes. Con ellas se verifica que las funcionalidades probadas de manera unitaria en las pruebas anteriores sean correctas también cuando se trata del conjunto de componentes.

5.7.3 Pruebas de usabilidad

Dados los objetivos establecidos por el cliente para la aplicación, las pruebas de usabilidad comprueban que estos se hayan cumplido; es decir, comprueban la calidad del software desde el punto de vista del usuario final.

Permiten determinar cómo de bien interactúan los usuarios finales con la interfaz de la aplicación diseñada. Dado que la aplicación se realizará orientada a varios perfiles de usuarios (desde gente sin conocimientos avanzados en el ámbito de los trenes, hasta empleados de la propia empresa), estas pruebas serán realizadas a personas con distintos niveles de conocimiento también.

Tras la realización de esta prueba de usabilidad a los usuarios, estos deberán rellenar un formulario cuya puntuación determinará cómo de usable es la aplicación.



Capítulo 6 DISEÑO DEL SISTEMA DE INFORMACIÓN

FASE DE DESARROLLO

DSI

A partir del análisis realizado en el apartado anterior, se procede a detallar el diseño del sistema.

6.1 DSI 3: DISEÑO DE CASOS DE USO REALES

En este apartado se detallan, mediante diagramas de interacción, los casos con mayor relevancia en la aplicación, ya descritos en la fase de análisis.

6.1.1 Caso de Uso 2: Consultar rutas y horarios

Una vez el usuario accede a la página de consulta de rutas y horarios debe seleccionar la estación de origen y la estación de destino. A continuación, se procede a la consulta del horario a través de una petición HTTP GET al REST controller. Cuando la petición llega al controlador, este llama al servicio, donde se lleva a cabo la lógica de negocio. Por último, tras las comprobaciones pertinentes, se llama al repositorio para realizar la consulta. El proceso finaliza con el retorno de una lista de horarios.

A continuación, se muestra el diagrama de interacción correspondiente con el Caso de Uso 2: Consultar rutas y horarios.

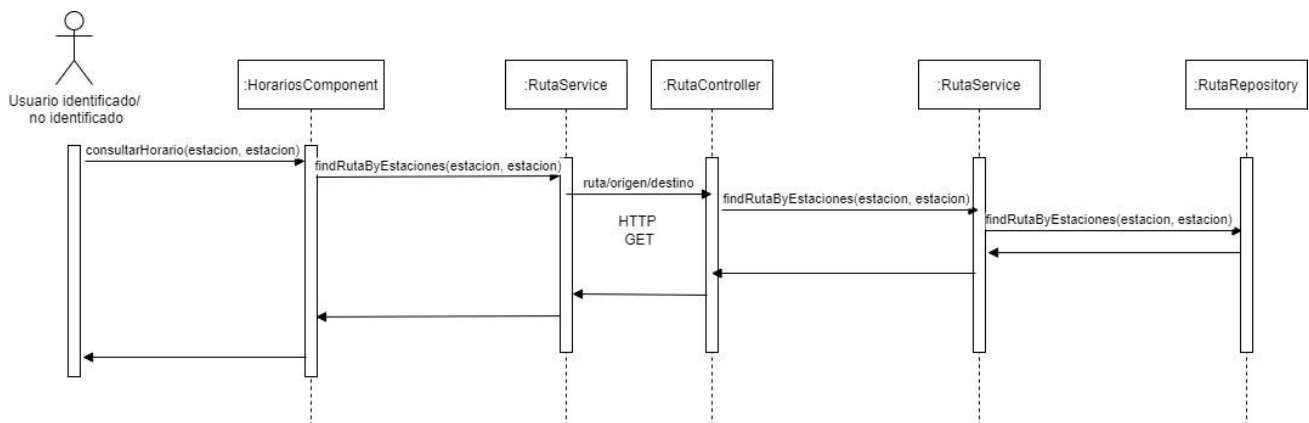


Ilustración 47. Diagrama de interacción caso de uso 2

6.1.2 Caso de uso 4: Gestionar solicitudes de vacaciones y días libres (administrador)

El usuario administrador identificado accede a través del menú a la ventana de gestionar solicitudes de vacaciones y días libres pulsando el botón de “Ver solicitudes”.

El mecanismo para aceptar o rechazar una solicitud es el mismo; se accede mediante una petición PUT al SolicitudController, que a su vez llama al servicio. En el servicio se realizan varias operaciones; en primer lugar, se buscan las tareas asignadas a la jornada y en caso de que existan se eliminan, por otra parte, marcamos el día como libre actualizando el campo en la jornada a través del

JornadaService y JornadaRepository; por otra parte, decrementamos el número de días libres disponibles para ese empleado; y, por último, marcamos la solicitud como aceptada.

Además de esto, el administrador puede consultar los detalles de los empleados de cada solicitud y la fecha de la misma.

A continuación, se muestra el diagrama de interacción correspondiente al Caso de uso 4: Gestionar solicitudes de vacaciones y días libres (administrador)

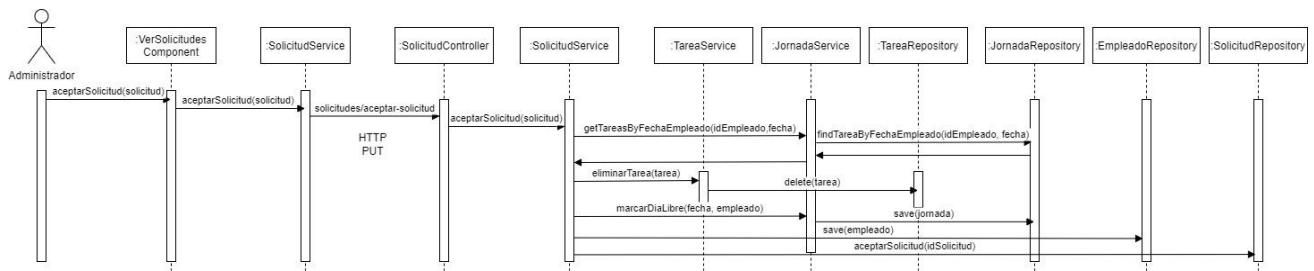


Ilustración 48. Diagrama de interacción caso de uso 4

6.1.3 Caso de uso 5: Gestionar jornada laboral

El usuario administrador accede a través del menú al apartado de “Gestionar usuarios y jornada”. Debe seleccionar un empleado de la lista y un día del calendario para poder añadir una nueva tarea a la jornada del empleado. Una vez el administrador rellena todos los campos de la ventana modal para crear una nueva tarea, se llama al servicio de la jornada para comprobar si ya existe una jornada para ese empleado en ese día.

En caso de que la jornada ya exista esta se actualiza con la nueva tarea, si por el contrario la jornada no existe se crea, en ambos casos esta se almacena en la base de datos con una petición POST al back-end. Además, se asigna a la tarea creada las estaciones de origen y destino.

A continuación, se muestra el diagrama de interacción correspondiente al Caso de uso 5: Gestionar jornada laboral.

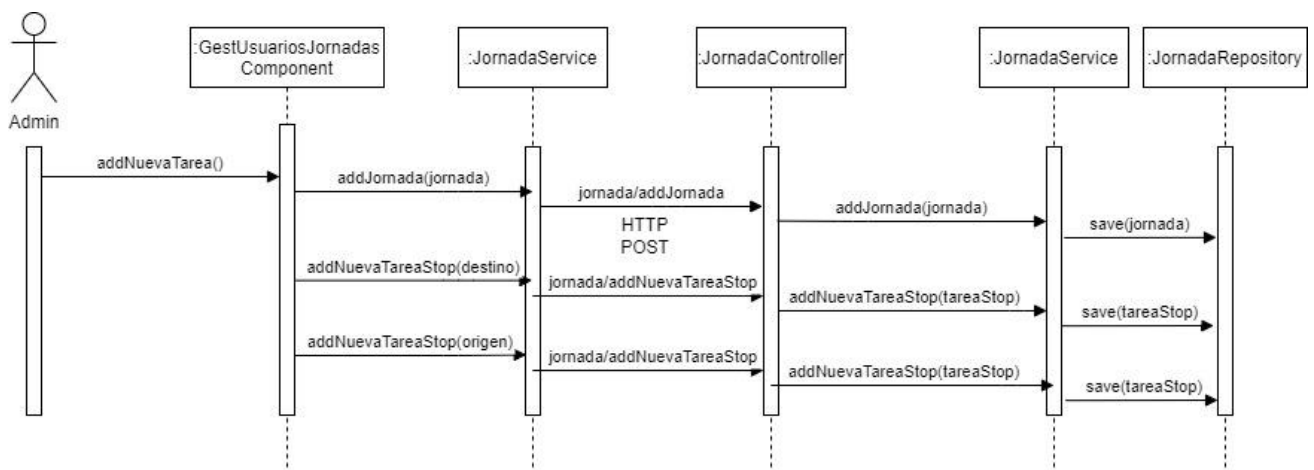


Ilustración 49. Diagrama de interacción caso de uso 5

6.1.4 Caso de uso 6: Solicitar cambio de jornada a compañero

Para solicitar un cambio de jornada a un compañero el empleado debe acceder mediante el menú a la sección de Gestión de jornada, para posteriormente acceder al portal de solicitudes.

Una vez el empleado está en el portal de solicitudes puede hacer una nueva petición de intercambio a través del botón “Nueva solicitud”. Tras cumplimentar el formulario debidamente se llama al servicio para acceder al back-end y almacenar la solicitud. En primer lugar, se comprueba que el empleado tiene una jornada de trabajo asignada al día que solicita, al igual que se comprueba no tenga una jornada de trabajo asignada para el día que propone de trabajo. En caso de que ambas condiciones se cumplan se pasa a almacenar la solicitud.

La siguiente imagen corresponde con el diagrama de interacción del Caso de uso 6: Solicitar cambio de jornada a compañero

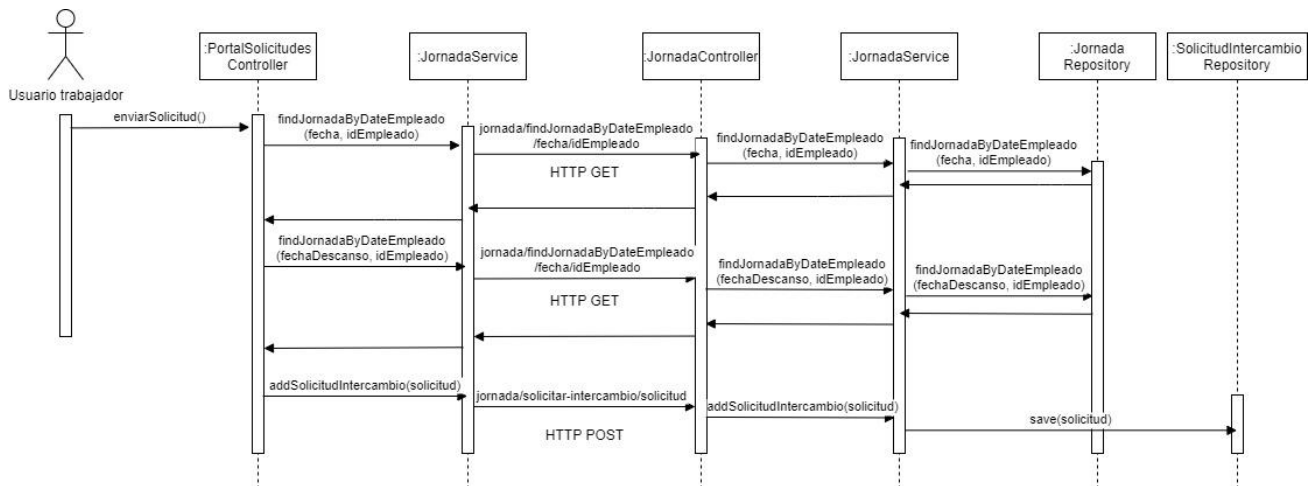


Ilustración 50. Diagrama de interacción caso de uso 6

6.1.5 Caso de uso 7: Aceptar cambio de jornada

Al igual que en el apartado anterior, el empleado debe acceder al portal de solicitudes para poder aceptar un cambio de jornada.

En primer lugar, mediante el servicio de las solicitudes se obtienen únicamente aquellas que el nuevo empleado puede cubrir por coincidencia de fechas. Una vez el empleado acepta una de estas solicitudes se accede al back-end mediante una petición POST. Es en el JornadaService donde se asignará el nuevo empleado a la solicitud, donde se realizará el intercambio de empleados para las jornadas; y donde por último se marcará la solicitud como “REASIGNADA”.

A continuación, se muestra el diagrama de interacción correspondiente al Caso de uso 7: Aceptar cambio de jornada

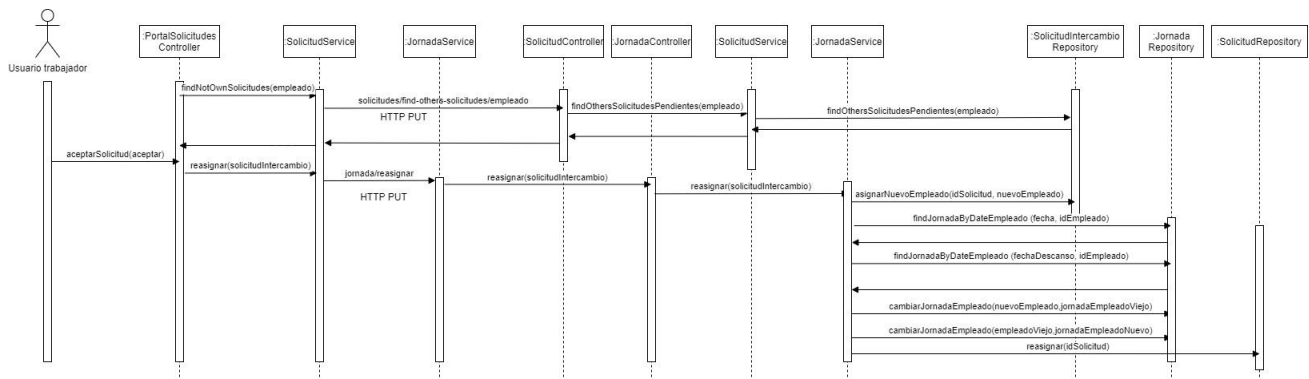


Ilustración 51. Diagrama de interacción caso de uso 7

6.1.6 Caso de uso 8: Solicitar vacaciones

Para solicitar vacaciones el usuario trabajador debe seleccionar el menú la opción de “Gestión de jornada” y el subapartado “Solicitar vacaciones”.

Al cargar la página se comprueba que el usuario no tiene ya una solicitud de vacaciones en estado distinto de rechazada. Cuando se solicita uno de los periodos disponibles se hace a través del “JornadaService” con una petición HTTP POST al controlador.

A continuación, se muestra el diagrama de interacción del Caso de uso 8: Solicitar vacaciones

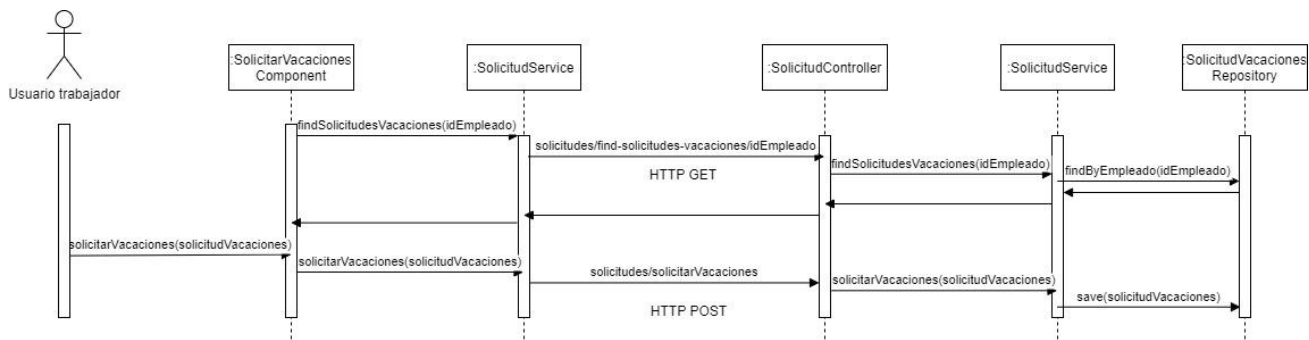


Ilustración 52. Diagrama de interacción caso de uso 8

6.1.7 Caso de uso 9: Solicitar días libres

Desde la barra de menú el usuario debe acceder a “Gestión de jornada” y posteriormente “Consultar jornada” para poder solicitar días libres.

Una vez que el usuario selecciona el día del calendario que desea tener libre, se llama al JornadaService y a través de una petición HTTP POST se envía una solicitud para el día seleccionado, esta será recogida en el SolicitudController, que comunicándose con el SolicitudService y el SolicitudSimpleRepository almacenarán la solicitud con estado “PENDIENTE” en la base de datos.

El diagrama de interacción mostrado en la parte inferior representa el Caso de uso 9: Solicitar días libres

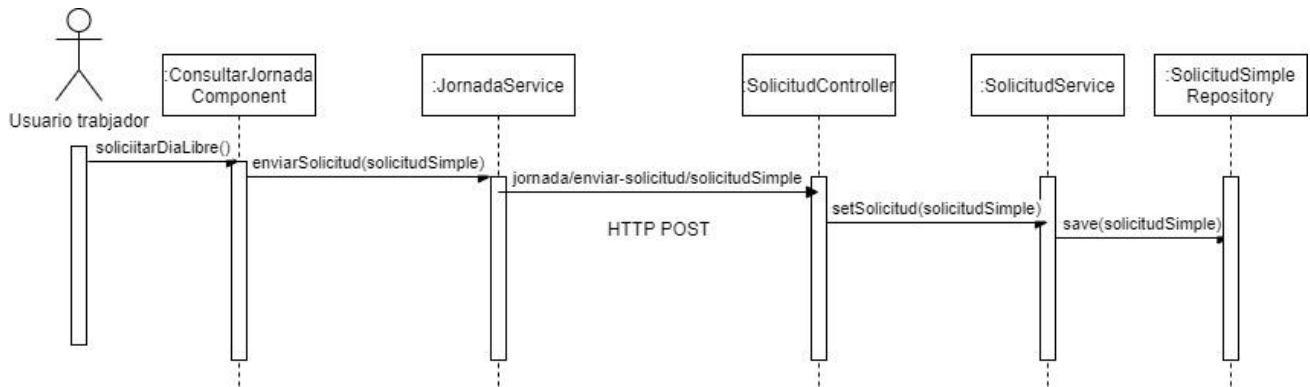


Ilustración 53. Diagrama de interacción caso de uso 9

6.2 DSI 4: DISEÑO DE CLASES

En este apartado se muestran diagramas que representan las clases del proyecto y las relaciones que existen entre las mismas

6.2.1 Diagrama de Clases

En este apartado se muestran diferentes diagramas de clases, divididos por paquetes debido al gran tamaño de un único diagrama de clases global.

El primer diagrama (Ilustración 54) que podemos ver representa una vista general del diagrama de clases global de forma muy simplificada, separando estas en los cuatro paquetes principales.

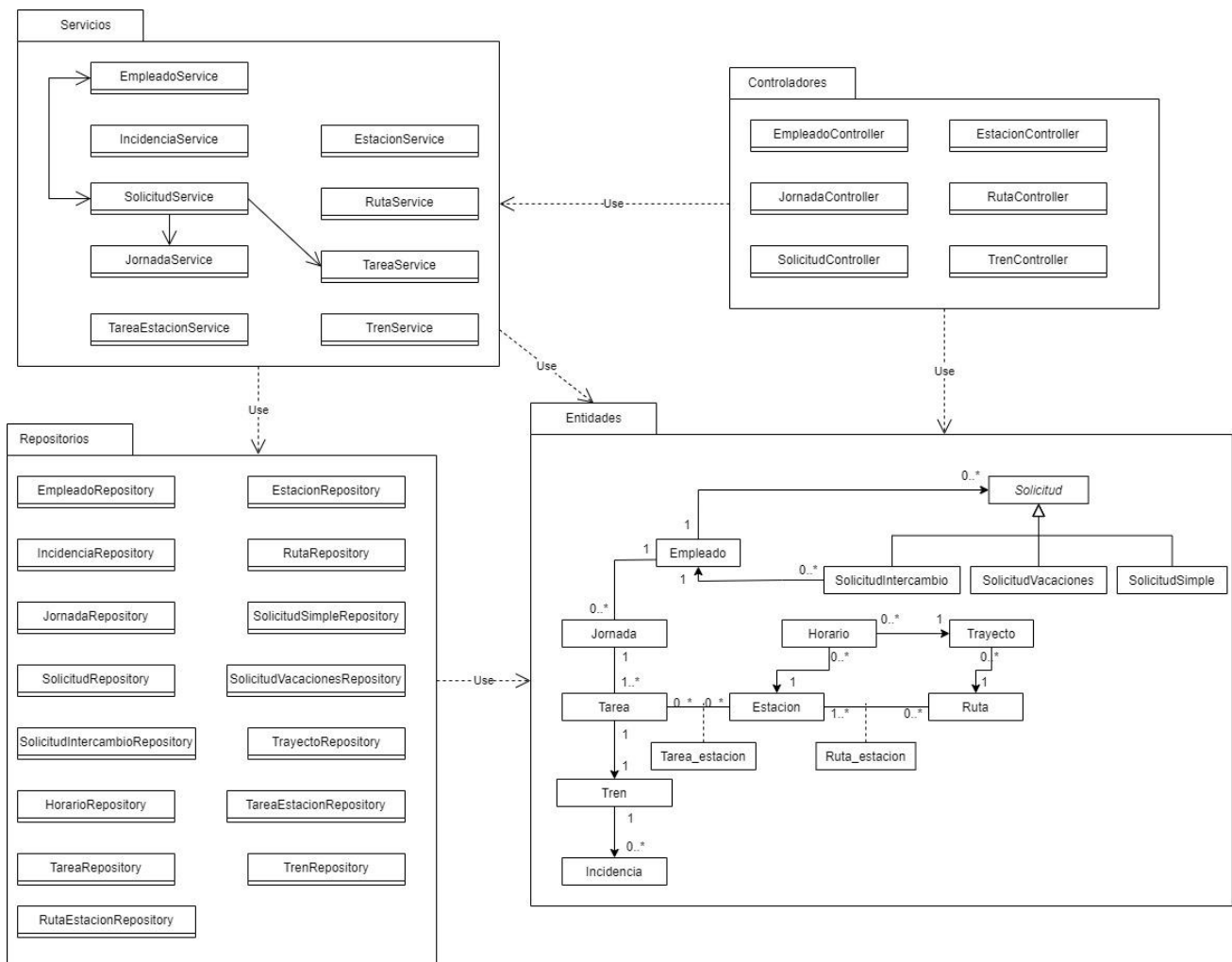


Ilustración 54. Diagrama de clases general

Una vez tenemos una visión general del sistema, procedemos a explorar cada uno de ellos más concretamente.

6.2.1.1 Entidades

Se representa a continuación el diagrama de clases de las entidades, incluyendo los atributos correspondientes

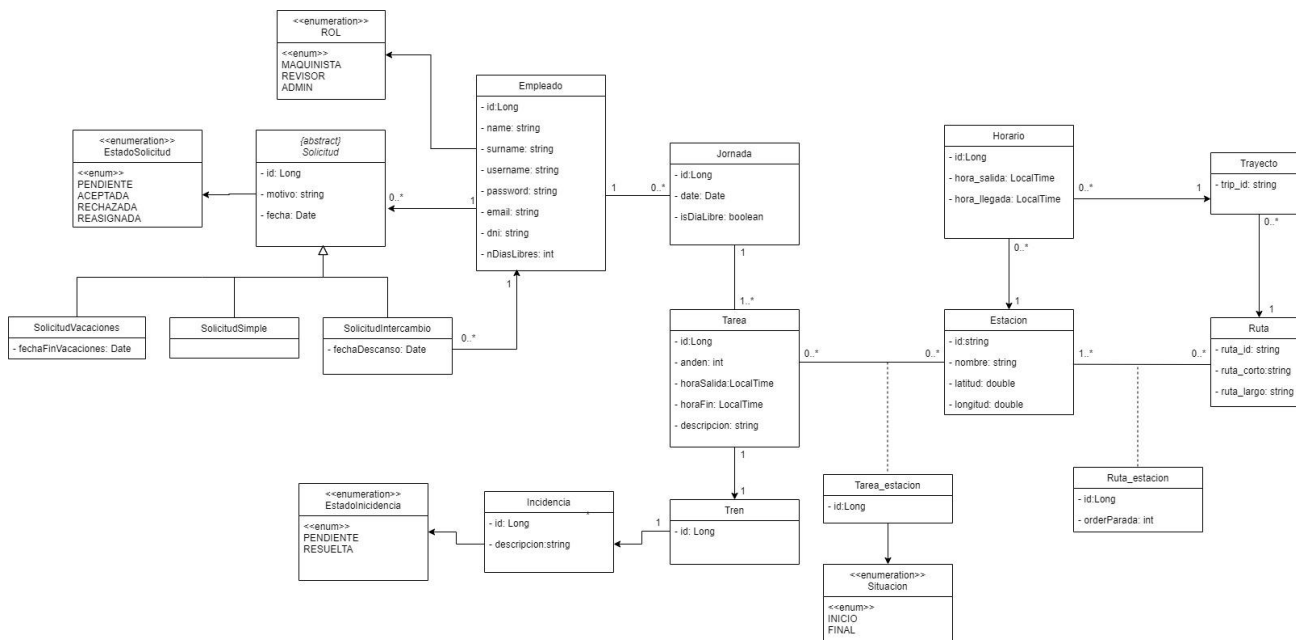


Ilustración 55. Diagrama de clases – entidades

La mayoría de las clases representadas anteriormente ya han sido explicadas en el apartado 5.5.2.1 Entidades, sin embargo; en este apartado se exponen algunas nuevas que han surgido.

Rol
Descripción
Enumerado que indica el ROL que tiene un empleado dentro de la empresa
Atributos
<ul style="list-style-type: none"> - MAQUINISTA - REVISOR - ADMIN

Tabla 55. Descripción de la clase Rol

EstadoIncidencia
Descripción
Enumerado que representa el estado en el que se encuentra una de las incidencias asociadas a un tren
Atributos
<ul style="list-style-type: none"> - PENDIENTE - RESUELTA

Tabla 56. Descripción de la clase EstadoIncidencia

EstadoSolicitud



Descripción
Enumerado que representa el estado en el que se encuentra una solicitud
Atributos
<ul style="list-style-type: none">- PENDIENTE- ACEPTADA- RECHAZADA- REASIGNADA: La solicitud ha sido asignada a un nuevo compañero

Tabla 57. Descripción de la clase EstadoSolicitud

Situación
Descripción
Enumerado que representa la situación de una estación en una tarea.
Atributos
<ul style="list-style-type: none">- INICIO: La tarea empieza en esa estación- FINAL: La tarea finaliza en esa estación

Tabla 58. Descripción de la clase Situación

6.2.1.2 Controladores

En este apartado se muestra de manera detallada el diagrama de clases de los controladores, con sus respectivos atributos y métodos

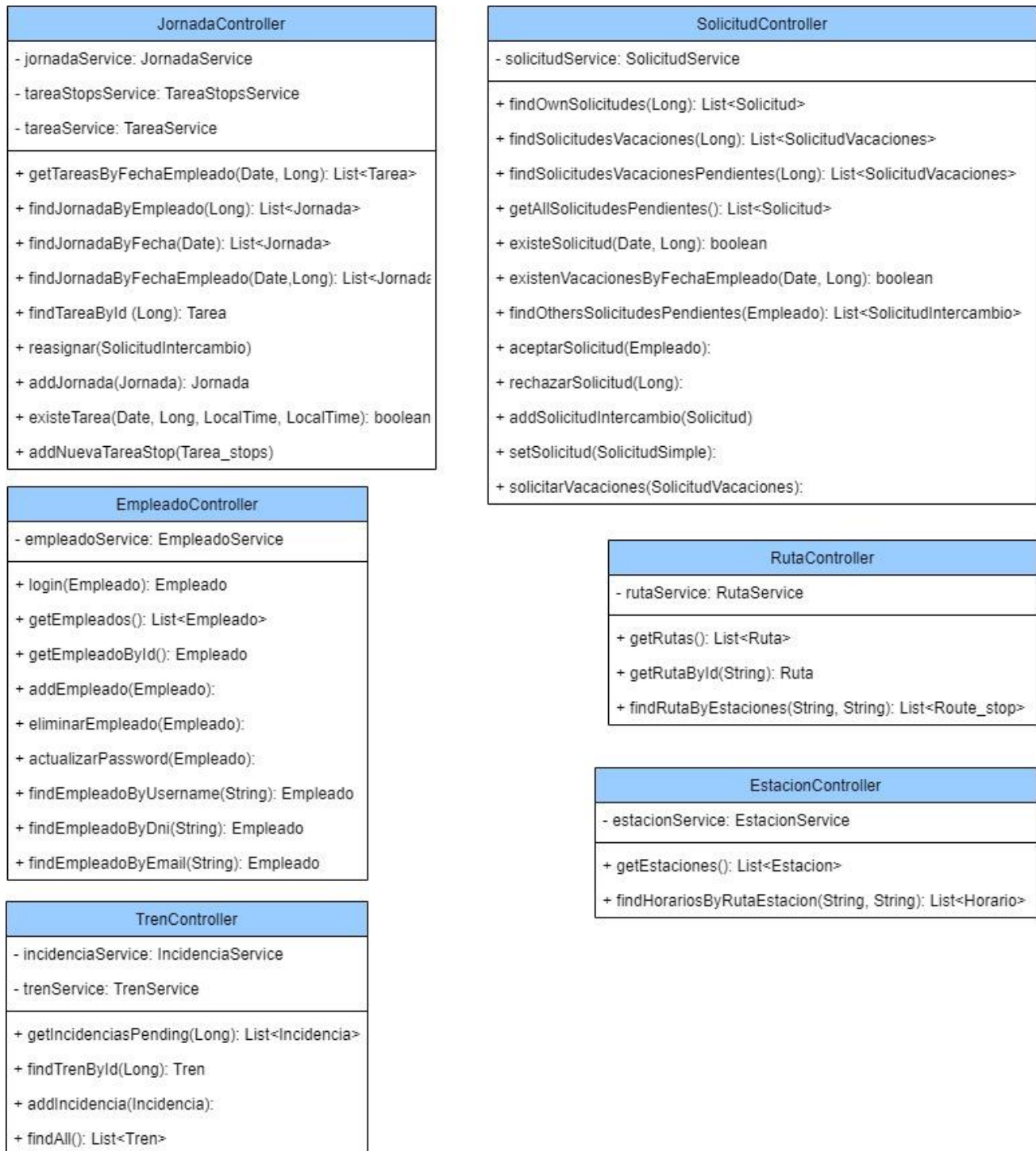


Ilustración 56. Diagrama de clases – controladores

A continuación se detallan los métodos de cada una de las clases y se explica su función.

6.2.1.2.1 JornadaController

JornadaController
Descripción
Controlador que gestiona las peticiones relacionadas con las jornadas.

Métodos

- **getTareasByFechaEmpleado(Date, Long): GET jornada/consultar/{date}/{id}**: Busca las tareas dada una fecha y un empleado y las devuelve en forma de lista.
- **findJornadaByEmpleado(Long): GET jornada/findByEmpleado/{id}**: Busca las jornadas de un empleado dado su id y las devuelve en forma de lista.
- **findJornadaByFecha(Date): GET jornada/findByDate/{date}**: Busca la jornadas asignadas a una fecha dada y las devuelve en forma de lista.
- **findJornadaByFechaEmpleado(Date, Long): GET jornada/findJornadaByDateEmpleado/{date}/{id}**: Busca las jornadas para un empleado en una fecha concreta y las devuelve en forma de lista.
- **findTareaById(Long): GET tarea/{id}**: Busca una tarea dado su id.
- **existeTarea(Date, Long, LocalTime, LocalTime): GET jornada/existeTarea/{fecha}/{idEmpleado}/{hSalida}/{hFin}**: Comprueba si existe una tarea asignada a un empleado en una fecha concreta que tenga coincidencia horaria con los datos pasados como parámetro.
- **reasignar(SolicitudIntercambio): PUT jornada/reasignar**: Realiza el proceso de reasignación cuando un empleado acepta una solicitud de intercambio.
- **addJornada(Jornada): POST jornada/addJornada**: Gestiona la petición post para añadir una nueva jornada.
- **addNuevaTareaEstacion(Tarea_estacion): POST tareaEstacion/addNuevaTareaEstacion**: Gestiona la petición post para añadir una nueva "tarea_estacion" .

Tabla 59. Descripción de la clase JornadaController

6.2.1.2.2 SolicitudController

SolicitudController

Descripción

Controlador que gestiona las peticiones relacionadas con las solicitudes.

Métodos

- **findOwnSolicitudes(Long): GET solicitudes/find-own-solicitudes/{id}**: Busca las solicitudes de un empleado dado se id.
- **findSolicitudesVacaciones(Long): GET solicitudes/find-solicitudes-vacaciones/{id}**: Busca las solicitudes de vacaciones de un empleado dado su id.
- **findSolicitudesVacacionesPendientes(Long): GET solicitudes/find-solicitudes-vacaciones-pendientes/{id}**: Busca las solicitudes de vacaciones pendientes de un empleado dado su id.
- **getAllSolicitudesPendientes(): GET solicitudes/ver-solicitudes-pendientes**: Busca todas las solicitudes en estado pendiente y las devuelve en formato de lista al front.
- **existeSolicitud(Date, Long): GET solicitudes/existeSolicitud/{fecha}/{idEmpleado}**: Comprueba si existe una solicitud para ese empleado en esa fecha.

- **existenVacacionesByFechaEmpleado(Date, Long): GET solicitudes/existenVacacionesByFechaEmpleado/{fecha}/{idEmpleado}**: Comprueba si existe una solicitud de vacaciones para la fecha y empleado proporcionados.
- **findOthersSolicitudesPendientes(Empleado): PUT solicitudes/find-others-solicitudes**: Busca las solicitudes de intercambio de compañeros en estado pendiente.
- **aceptarSolicitud(Solicitud): PUT solicitudes/aceptar-solicitud**: Gestiona la petición PUT que se realiza cuando el administrador acepta una solicitud.
- **rechazarSolicitud(Solicitud): PUT solicitudes/rechazar-solicitud**: Gestiona la petición PUT que se realiza cuando el administrador rechaza una solicitud.
- **setSolicitud(SolicitudSimple): POST solicitudes/enviar-solicitud**: Gestiona la petición para registrar una nueva solicitud simple.
- **solicitarVacaciones(SolicitudVacaciones): POST solicitudes/solicitar-vacaciones**: Gestiona la petición para registrar una nueva solicitud de vacaciones
- **addSolicitudIntercambio(Solicitud): POST solicitudes/solicitar-intercambio**: Añade una nueva solicitud de tipo intercambio.

Tabla 60. Descripción de la clase SolicitudController

6.2.1.2.3 RutaController

RutaController
Descripción
Controlador que gestiona las peticiones relacionadas con las rutas.
Métodos
<ul style="list-style-type: none">- getRutas(): GET /rutas: Busca todas las rutas y las devuelve en forma de lista.- findRutaByEstaciones(String, String): GET ruta/{origenId}/{destinoId}: Dadas la estación de origen y la estación de destino devuelve una lista con las "ruta_estacion" que la componen.- getRutaById(String): GET rutas/{id}: Dado el id de una ruta devuelve la ruta en sí.

Tabla 61. Descripción de la clase RutaController

6.2.1.2.4 EmpleadoController

EmpleadoController
Descripción
Controlador que trata las peticiones relacionadas con los empleados.
Métodos
<ul style="list-style-type: none">- getEmpleados(): GET /empleados: Devuelve una lista de todos los empleados- getEmpleadoById(Long): GET empleado/{id}: Dado un id devuelve el empleado correspondiente- findEmpleadoByUsername(String) GET empleados/findEmpleadoByUsername/{username}: Busca un empleado con el username solicitado.- findEmpleadoByDni(String) GET empleados/findEmpleadoByDni/{dni}: Busca un empleado con el dni solicitado.

- **findEmpleadoByEmail(String) GET empleados/findEmpleadoByEmail/{email}**: Busca un empleado con el email solicitado.
- **login(Empleado): POST /login**: Gestiona la petición relativa al inicio de sesión de un empleado en la aplicación.
- **addEmpleado(Empleado): POST empleados/addEmpleado**: Añade un nuevo empleado en el sistema.
- **eliminarEmpleado(Empleado): DELETE empleados/eliminarEmpleado/{id}**: Dado el id de un empleado gestiona una petición DELETE para borrarlo del sistema.
- **actualizarPassword(Empleado): PUT empleados/actualizarEmpleado**: Se encarga de gestionar la petición PUT para actualizar en la base de datos la contraseña de un empleado.

Tabla 62. Descripción de la clase EmpleadoController

6.2.1.2.5 EstacionController

EstacionController
Descripción
Controlador que trata las peticiones relacionadas con las estaciones
Métodos
<ul style="list-style-type: none">- getEstaciones(): GET /estaciones: Devuelve la lista de todas las estaciones-- findHorariosByRutaEstacion(String, String): GET horarios/{rutaId}/{estacionId}: Método que dados el id de la ruta y el id de la estación devuelve una lista de horarios.

Tabla 63. Descripción de la clase EstacionController

6.2.1.2.6 TrenController

TrenController
Descripción
Controlador que trata las peticiones relacionadas con los trenes
Métodos
<ul style="list-style-type: none">- getIncidenciasPending(Long): GET tren/getIncidenciasPending/{id}: Devuelve una lista con las incidencias cuyo estado es pendiente.- findTrenById(Long): GET tren/findTrenById/{id}: Dado un id devuelve el tren correspondiente.- addIncidencia(Incidencia): POST tren/addIncidencia: Gestiona la petición para añadir una nueva incidencia a un tren.- findAll(): GET tren/findAll: Obtiene todos los trenes

Tabla 64. Descripción de la clase TrenController

6.2.1.3 Servicios

Se detallan a continuación los servicios creados para la aplicación y las relaciones entre ellos.

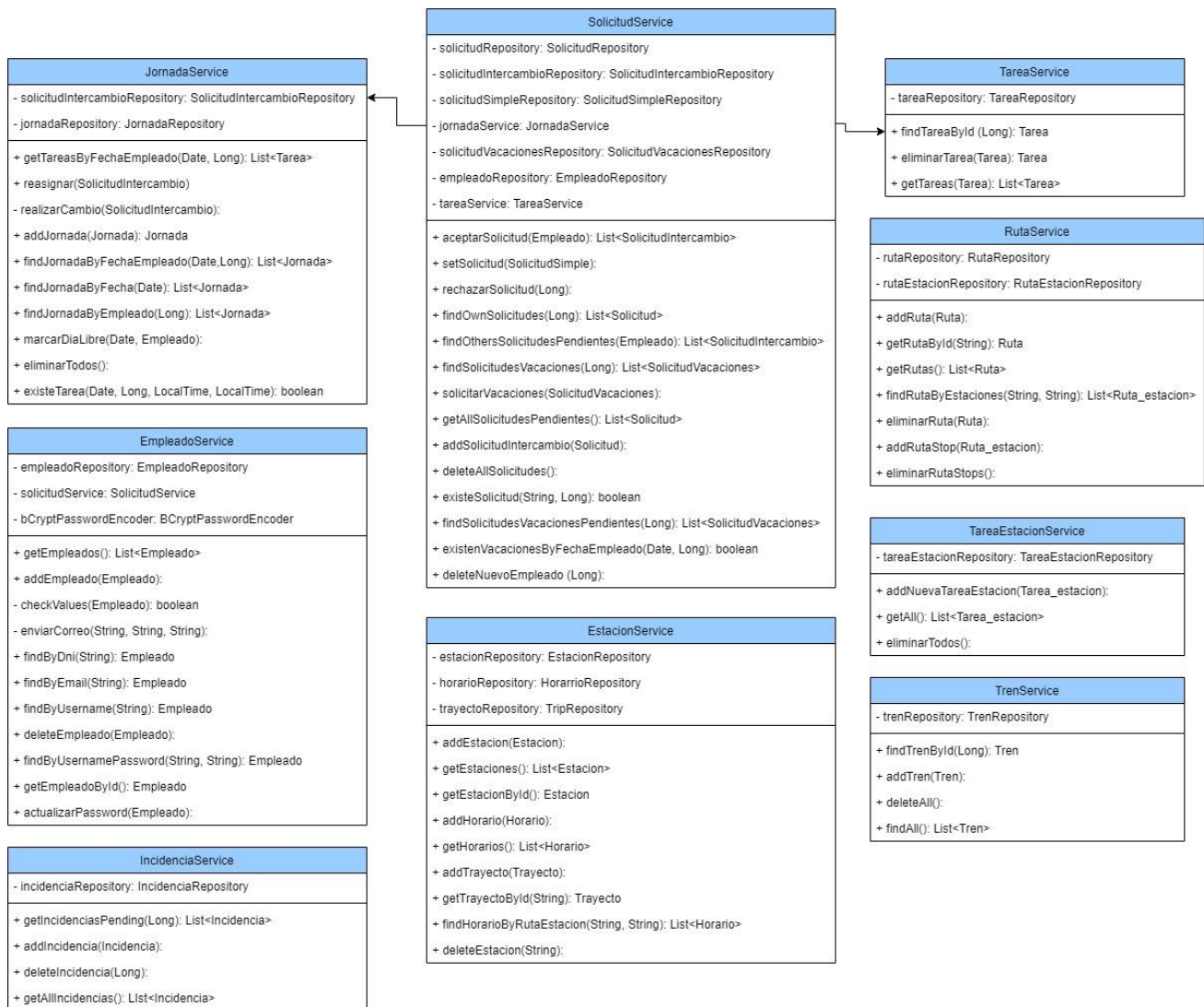


Ilustración 57. Diagrama de clases – servicios

Una vez se tiene una visión general de los Servicios y sus métodos se procede a describir en qué consiste cada uno de ellos de manera específica.

6.2.1.3.1 JornadaService

JornadaService	
Descripción	Servicio que gestiona las jornadas.
Métodos	<ul style="list-style-type: none"> - getTareasByFechaEmpleado(Long, Date): Método que dada una fecha y un empleado y las devuelve las tareas que les corresponden. - reasignar(SolicitudIntercambio): Realiza el proceso de reasignación cuando un empleado acepta una solicitud de intercambio. En primer lugar, asigna el nuevo empleado de la solicitud, después realiza el cambio y por último cambia el estado de la solicitud

- **realizarCambio(SolicitudIntercambio):** Método privado que intercambia las jornadas entre el empleado asignado a la solicitud y el nuevo empleado.
- **findJornadaByEmpleado(Long):** Devuelve la lista de jornadas de un empleado buscando por el id del empleado.
- **findJornadaByFecha(Date):** Devuelve la lista de jornadas de un empleado buscando por la fecha de la jornada.
- **findJornadaByFechaEmpleado(Date, Long):** Devuelve la lista de jornadas de un empleado buscando por la fecha de la jornada y el id del empleado.
- **addJornada(Jornada):** Añade una nueva jornada.
- **marcarDiaLibre(Date, Empleado):** Método que marca un día como libre; en primer lugar modifica el atributo booleano de la clase Jornada, vacía la lista de tareas de la jornada y por último actualiza la jornada modificada.
- **eliminarTodos():** Elimina todas las jornadas existentes.
- **existeTarea(Date, Long, LocalTime, LocalTime):** Comprueba si existe una tarea para un empleado en una fecha concreta que coincida con el horario pasado por parámetro.

Tabla 65. Descripción de la tabla JornadaService

6.2.1.3.2 SolicitudService

SolicitudService

Descripción

Servicio que gestiona las solicitudes.

Métodos

- **aceptarSolicitud(Solicitud):** Gestiona la aceptación de una solicitud por parte del administrador. Independientemente del tipo de solicitud se marca la jornada o jornadas como libre, se eliminan las tareas que pudiesen tener asignadas y se cambia el estado de la solicitud a "Aceptada". En caso de que sea una solicitud de tipo simple, adicionalmente se decrementa el número de días libres de los que dispone el empleado correspondiente.
- **findOwnSolicitudes(Long):** Devuelve las solicitudes simples y de tipo intercambio de un empleado dado su id.
- **findSolicitudesVacaciones(Long):** Método que devuelve una lista de las solicitudes de vacaciones de un empleado cuyo id se pasa por parámetro.
- **getAllSolicitudesPendientes():** Devuelve todas las solicitudes simples y de vacaciones en estado pendiente.
- **findOthersSolicitudesPendientes(Empleado):** Devuelve las solicitudes de tipo intercambio que sean compatibles con las jornadas del empleado que se pasa por parámetro.
- **rechazarSolicitud(Solicitud):** Método que marca la solicitud como rechazada.
- **setSolicitud(SolicitudSimple):** Añade una nueva solicitudSimple.

- **solicitarVacaciones(SolicitudVacaciones)**: Añade una nueva solicitud de vacaciones.
- **addSolicitudIntercambio(Solicitud)**: Añade una nueva solicitud de intercambio.
- **deleteAllSolicitudes()**: Elimina todas las solicitudes.
- **existeSolicitud(String, Long)**: Busca una solicitud por fecha y empleado asociado y devuelve true o false dependiendo de si existe o no.
- **findSolicitudesVacacionesPendientes(Long)**: Busca las solicitudes de vacaciones con estado pendiente dado el id de un empleado.
- **existenVacacionesByFechaEmpleado(Date, Long)**: Comprueba si existen solicitudes de vacaciones para la fecha y empleado dados.
- **deleteNuevoEmpleado(Long)**: Elimina el nuevo empleado asignado de todas las solicitudes de vacaciones en las que aparezca.

Tabla 66. Descripción de la clase SolicitudService

6.2.1.3.3 RutaService

RutaService
Descripción
Servicio que gestiona las rutas.
Métodos
<ul style="list-style-type: none">- getRutas(): Devuelve una lista con todas las rutas.- findRutaByEstaciones(String, String): Dadas la estación de origen y la estación de destino devuelve una lista con las "Ruta_estacion" que la componen.- getRutaById(String): Método que devuelve una ruta dado su id.- addRuta(Ruta): Añade una nueva ruta.- eliminarRuta(Ruta): Elimina la ruta pasada por parámetro.- addRutaEstacion(Ruta_estacion): Añade una " Ruta_estacion".- eliminarRutaEstaciones(): Elimina todas las "Ruta_estacion".

Tabla 67. Descripción de la clase RutaService

6.2.1.3.4 EmpleadoService

EmpleadoController
Descripción
Servicio que gestiona lo relativo a los empleados.
Métodos
<ul style="list-style-type: none">- getEmpleados(): Devuelve una lista de todos los empleados- getEmpleadoById(Long): Busca un empleado dado su id.- checkValues(Empleado): Comprueba que no existe otro empleado en el sistema con ese nombre de usuario.- enviarCorreo(String, String, String): Envía al email del nuevo empleado a crear un correo con su contraseña.- findByDni(String): Busca un empleado por dni

- **findByEmail(String):** Busca un empleado por email
- **findByUsername(String):** Busca un empleado por nombre de usuario
- **addEmpleado(Empleado):** Añade un nuevo empleado, encriptando su contraseña.
- **deleteEmpleado(Empleado):** Elimina el empleado que se pasa por parámetro
- **actualizarPassword(Empleado):** Actualiza la contraseña de un empleado.

Tabla 68. Descripción de la clase EmpleadoController

6.2.1.3.5 EstacionService

EstacionService
Descripción
Servicio que gestiona lo relativo a las estaciones.
Métodos
<ul style="list-style-type: none">- getEstaciones(): Devuelve la lista de todas las estaciones- addEstacion(Estacion): Añade una nueva estación.- getEstacionById(String): Busca una estación por su id.- addHorario(Horario): Añade un nuevo horario.- getHorarios(): Devuelve la lista de horarios.- addTrayecto(Trayecto): Añade un nuevo trayecto.- getTrayectoById(String): Busca un trayecto por su id.- findHorarioByRutaEstacion(String, String): Método que dados el id de la ruta y el id de la estación devuelve una lista de horarios de la ruta que incluye esa estación.- deleteEstacion(String): Elimina una estación dado su id.

Tabla 69. Descripción de la clase EstacionService

6.2.1.3.6 TrenService

TrenService
Descripción
Servicio que trata lo relativo a los trenes.
Métodos
<ul style="list-style-type: none">- findTrenById(Long): Busca un tren dado su id.- addTren(Tren): Añade un tren- deleteAll(): Elimina todos los trenes- findAll(): Busca todos los trenes y los devuelve en forma de lista

Tabla 70. Descripción de la clase TrenService

6.2.1.3.7 IncidenciaService

IncidenciaService
Descripción
Servicio que gestiona las incidencias.
Métodos



- **getIncidenciasPending(Long):** Devuelve una lista de las incidencias cuyo estado sea pendiente.
- **addIncidencia(Incidencia):** Añade una nueva incidencia.
- **deleteIncidencia(Long):** Elimina una incidencia dado su id.
- **getAllIncidencias():** Devuelve todas las incidencias.

Tabla 71. Descripción de la clase Incidencia

6.2.1.3.8 TareaService

TareaService
Descripción
Servicio que gestiona las tareas.
Métodos
<ul style="list-style-type: none">- findTareaById(Long): Busca una tarea por su id.- eliminarTarea(Tarea): Elimina una tarea que se pasa por parámetro.- getTareas(): Busca todas las tareas

Tabla 72. Descripción de la clase TareaService

6.2.1.3.9 TareaEstacionService

TareaEstacionService
Descripción
Servicio que gestiona las Tarea_estacion.
Métodos
<ul style="list-style-type: none">- addNuevaTareaEstacion(Tarea_estacion): Añade una nueva Tarea_estacion.- getAll(): Busca todas las "Tarea_estacion".- eliminarTodos(): Elimina todas las "Tarea_estacion".

Tabla 73. Descripción de la clase TareaEstacionService

6.2.1.4 Repositorios

En último lugar, este apartado recoge los repositorios que componen la aplicación.

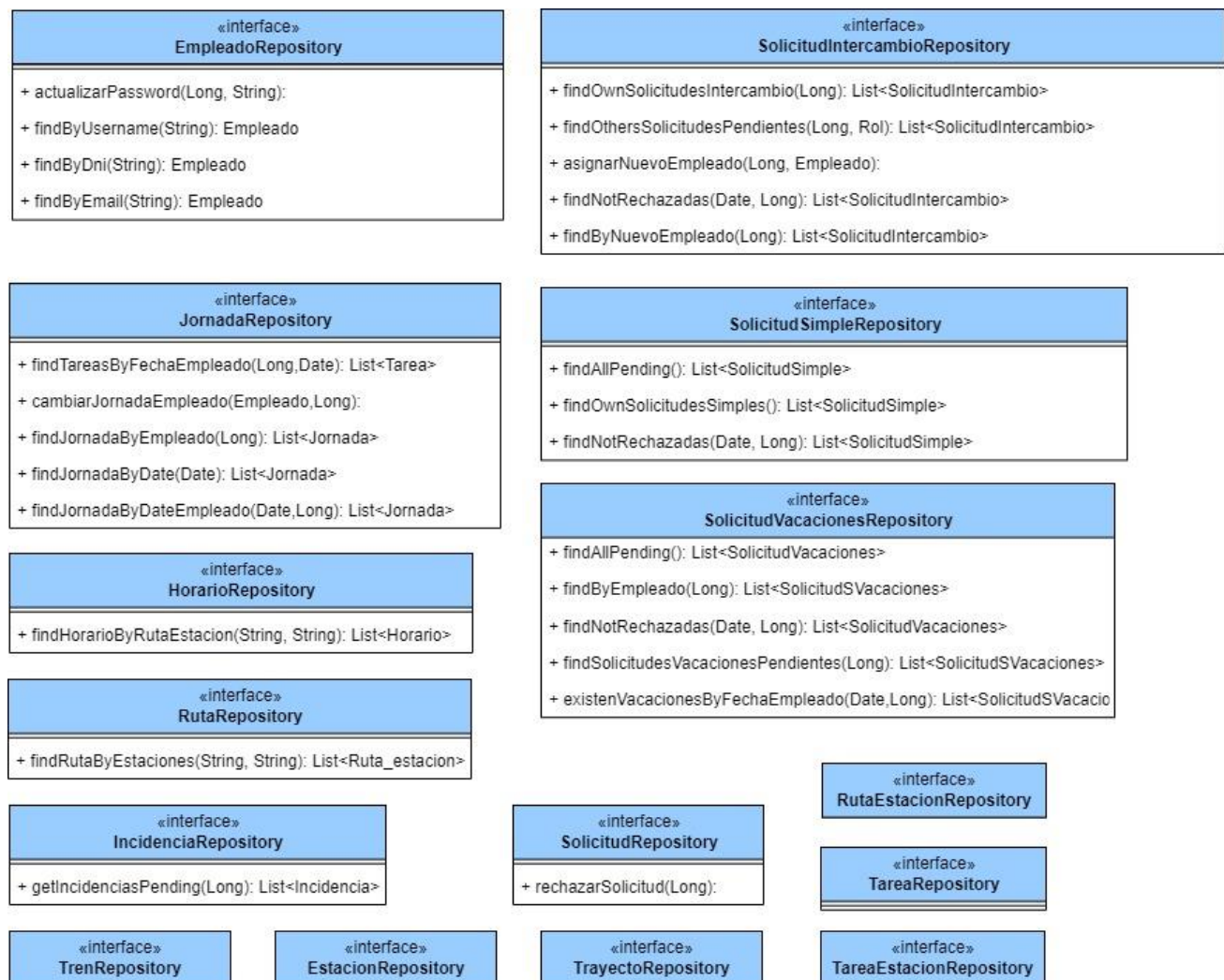


Ilustración 58. Diagrama de clases – repositorios

A continuación, se explica en qué consiste cada uno de los métodos que componen las interfaces de los repositorios.

6.2.1.4.1 JornadaRepository

JornadaRepository
Descripción
Repositorio que gestiona las jornadas de la BD.
Métodos
<ul style="list-style-type: none"> - findTareasByFechaEmpleado(Long, Date): Método que dada una fecha y un empleado y las devuelve las tareas que les corresponden. - cambiarJornadaEmpleado(Empleado, Id): Método que actualiza el empleado de una jornada dados el propio empleado y el id de la jornada.

- **findJornadaByEmpleado(Long)**: Busca en la BD las jornadas cuyo empleado asignado sea el pasado por parámetro.
- **findJornadaByFecha(Date)**: Busca en la BD las jornadas cuya fecha sea la pasada por parámetro.
- **findJornadaByFechaEmpleado(Date, Long)**: Busca en la BD las jornadas cuya fecha y empleado asignado sean los pasados por parámetro.

Tabla 74. Descripción de la tabla JornadaRepository

6.2.1.4.2 SolicitudRepository

SolicitudService

Descripción

Repository que gestiona las solicitudes de la BD.

Métodos

- **rechazarSolicitud(Long)**: Método que actualiza el estado de una solicitud a “RECHAZADA” pasando el id de la misma como parámetro.

Tabla 75. Descripción de la clase SolicitudService

6.2.1.4.3 SolicitudSimpleRepository

SolicitudSimpleRepository

Descripción

Repository que gestiona las solicitudes simples de la BD.

Métodos

- **findAllPending()**: Método que busca en la base de datos todas aquellas solicitudes simples con estado “PENDIENTE”.
- **findOwnSolicitudes(Long)**: Busca en la base de datos aquellas solicitudes simples cuyo empleado asignado tenga el mismo id que el pasado por parámetro.
- **findNotRechazadas(Date, Long)**: Busca en la base de datos aquellas solicitudes simples cuya fecha y empleado coincidan con los pasados y su estado sea distinto a rechazada.

Tabla 76. Descripción de la clase SolicitudSimpleRepository

6.2.1.4.4 SolicitudIntercambioRepository

SolicitudIntercambioRepository
Descripción
Repositorio que gestiona las solicitudes de intercambio de la BD.
Métodos
<ul style="list-style-type: none">- findOwnSolicitudesIntercambio(Long): Busca en la base de datos aquellas solicitudes de intercambio cuyo empleado asignado tenga el mismo id que el pasado por parámetro.- findOthersSolicitudesPendientes(Long, Rol): Busca en la BD aquellas solicitudes de intercambio cuyo empleado tenga un id y un rol coincidente con los pasados y además cuyo estado sea pendiente.- asignarNuevoEmpleado(Long, Empleado): Actualiza en la BD aquella solicitud de intercambio cuyo id coincida con el primer parámetro, actualizando el campo de nuevoEmpleado al segundo parámetro.- findNotRechazadas(Date, Long): Busca en la base de datos aquellas solicitudes de intercambio cuya fecha y empleado coincidan con los pasados y su estado sea distinto a rechazada.- findByNuevoEmpleado(Long): Busca las solicitudes de intercambio cuyo nuevo empleado coincida en id con el parámetro.

Tabla 77. Descripción de la clase SolicitudIntercambioRepository

6.2.1.4.5 SolicitudVacacionesRepository

SolicitudVacacionesRepository
Descripción
Repositorio que gestiona las solicitudes de vacaciones de la BD.
Métodos
<ul style="list-style-type: none">- findAllPending(): Método que busca en la base de datos todas aquellas solicitudes de vacaciones con estado "PENDIENTE".- findByEmpleado(Long): Busca en la base de datos aquellas solicitudes cuyo empleado asignado tenga un id que coincida con el pasado por parámetro.- findNotRechazadas(Date, Long): Busca en la base de datos aquellas solicitudes de vacaciones cuya fecha y empleado coincidan con los pasados y su estado sea distinto a rechazada.- findSolicitudesVacacionesPendientes(Long): Busca en la base de datos las solicitudes de vacaciones de un empleado con estado "PENDIENTE".- existenVacacionesByFechaEmpleado(Date, Long): Comprueba si existen solicitudes de vacaciones para la fecha y empleado seleccionados.

Tabla 78. Descripción de la clase SolicitudVacacionesRepository

6.2.1.4.6 RutaRepository

RutaRepository
Descripción
Repositorio que gestiona las rutas de la BD.
Métodos
<ul style="list-style-type: none">- findRutaByEstaciones(String, String): Método que busca en la base de datos las “Ruta_estacion” cuya estación de origen y de destino coincida con los parámetros pasados.

Tabla 79. Descripción de la clase RutaService

6.2.1.4.7 EmpleadoRepository

EmpleadoRepository
Descripción
Repositorio que gestiona los empleados de la BD.
Métodos
<ul style="list-style-type: none">- actualizarPassword(Long, String): Actualiza la contraseña de un empleado cuyo id debe coincidir con el primer parámetro del método.- findByUsername(String): Busca un empleado cuyo nombre de usuario coincida con el pasado por parámetro.- findByDni(String): Busca un empleado cuyo dni coincida con el pasado por parámetro.- findByEmail(String): Busca un empleado cuyo email coincida con el pasado por parámetro.

Tabla 80. Descripción de la clase EmpleadoRepository

6.2.1.4.8 IncidenciaRepository

IncidenciaRepository
Descripción
Repositorio que gestiona las incidencias de la BD.
Métodos
<ul style="list-style-type: none">- getIncidenciasPending(Long): Busca en la BD aquellas incidencias cuyo estado sea PENDIENTE y el id del tren que tengan asignado sea igual al parámetro pasado.

Tabla 81. Descripción de la clase IncidenciaRepository

6.2.1.4.9 HorarioRepository

HorarioRepository
Descripción
Repositorio que gestiona los horarios de la BD.
Métodos
<ul style="list-style-type: none">- findHorarioByRutaEstacion(String, String): Método que busca en la BD aquellos que tengas asignada una ruta cuyo id corresponda con el primer parámetro del método y cuyo id de la estación coincida con el segundo parámetro del método.

Tabla 82. Descripción de la clase HorarioRepository



6.3 DSI 5: DISEÑO DE LA ARQUITECTURA DE MÓDULOS DEL SISTEMA

En este apartado se definen los módulos del sistema de información y la manera en que interactúan entre ellos.

6.3.1 DSI 5.1 Diseño de Módulos del Sistema

El sistema está dividido en dos módulos principales, por una parte está el back-end de la aplicación y por otra parte está el front-en. La comunicación entre los mismos se realiza mediante peticiones HTTP. El back-end recupera los objetos de la base de datos y los convierte a para pasarlos al front.

6.3.1.1 *Diagrama de paquetes*

El sistema se divide en paquetes; estos agrupan las clases de manera lógica dividiéndolas según su funcionalidad. A continuación, podemos ver la estructura de paquetes tanto del back-end como del front-end de la aplicación y su comunicación entre ellos.

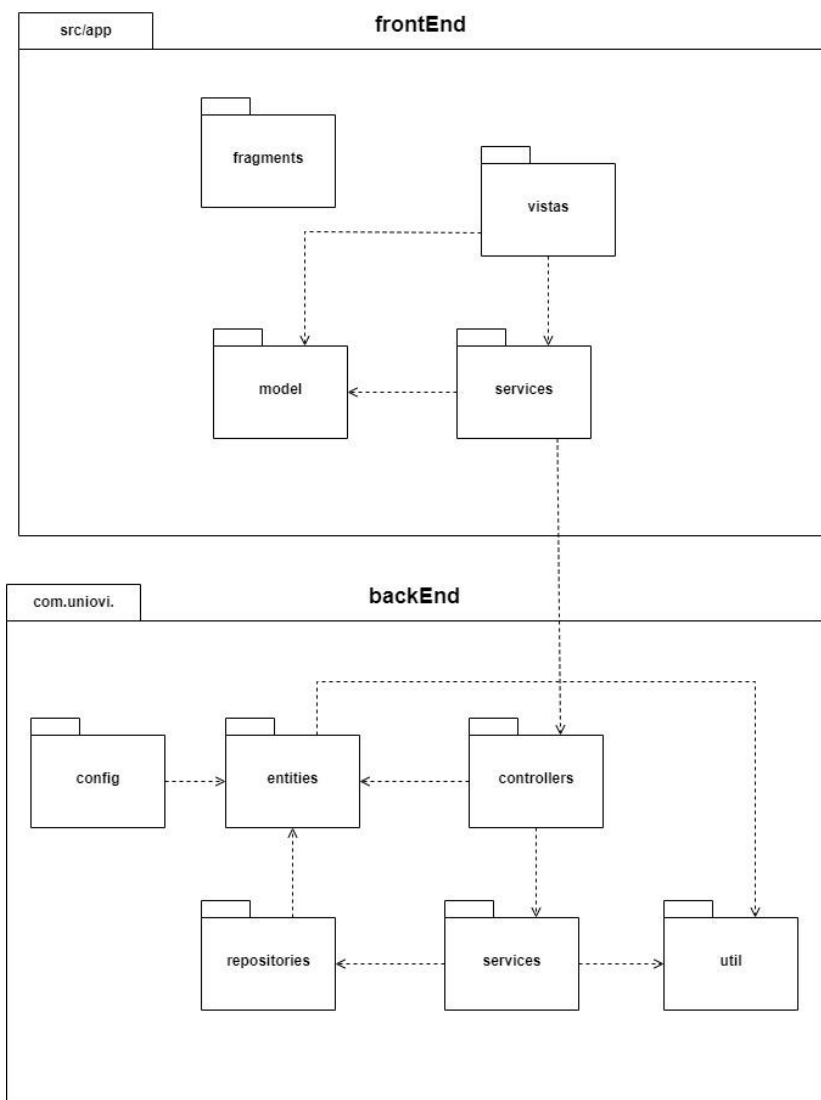


Ilustración 59. Diagrama de paquetes

En el front-end de la aplicación la carpeta principal se encuentra bajo la ruta `src/app`. En el interior de este paquete se encontrarán otros cuatro paquetes:

- El paquete **fragments** contendrá las vistas de la barra de navegación en la app y el pie de página
- El paquete **vistas** contendrá el resto de componentes de la página web.
- El paquete **model** encapsulará las entidades.
- El paquete **servicies** contendrá los servicios que permiten la comunicación con el back-end haciendo las peticiones correspondientes.

Por otra parte, en el back-end el paquete principal se encuentra situado bajo la ruta `src/main/java/com/uniovi`. En su interior podemos ver otros seis paquetes:

- El paquete **entities** contendrá todas las entidades
- El paquete **controllers** contendrá los controladores de la aplicación
- El paquete **services** incluirá los servicios.

- La carpeta **repositories** encapsulará los repositorios.
- Las clases de configuración se encontrarán en el paquete **config**.
- Las clases de **util** estarán localizadas en el paquete con el mismo nombre. En este se incluirán las clases que leen los ficheros CSV con datos iniciales de la compañía Renfe; esta clase únicamente será utilizada para crear el script de creación para cargar los datos en la base de datos. Además, el paquete util contendrá un validador de argumentos del cuál se hace uso en las entidades.

6.3.1.2 Diagrama de componentes

En el siguiente diagrama de componentes se muestran los componentes del sistema y sus dependencias

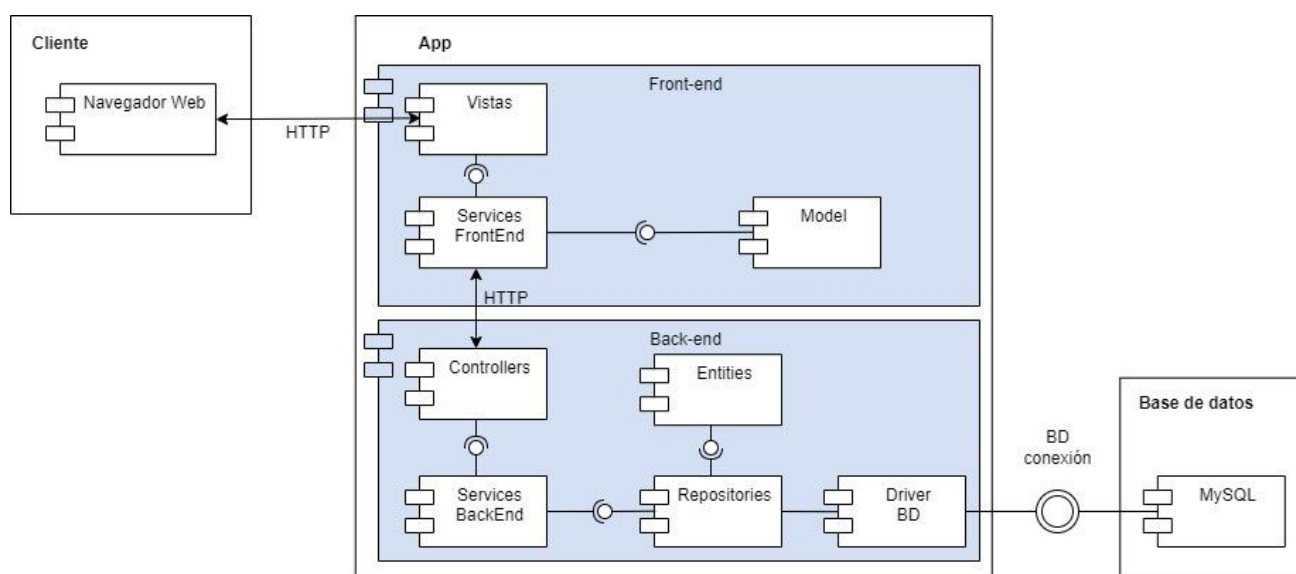


Ilustración 60. Diagrama de componentes

6.3.1.2.1 Cliente

Representa el producto final con el que el usuario puede interactuar a través de su navegador web.

6.3.1.2.2 Servidor – App

El servidor representa la aplicación en sí. En su interior tenemos tanto la parte de front-end como la parte de back-end de la aplicación.

En el diagrama se puede ver como el cliente se comunica con el front-end, mediante peticiones HTTP. Los servicios del front a su vez realizan peticiones HTTP a los controladores del back-end.

La obtención de los datos se realiza gracias a la conexión entre los repositorios del back-end y la base de datos, haciendo uso del Driver. Estos datos que se obtienen se convierten a JSON mediante las anotaciones propias de Spring. Una vez los datos están en el formato correcto se pasan al front y con

el servicio HttpClient, que es integrado en el front-end como clase, se parsean estos JSON a un objeto de la clase deseada; correspondiente a las entidades del proyecto.

6.3.1.2.3 Base de datos

Este componente trata la persistencia de datos, haciendo uso de la base de datos MySQL.

6.3.1.3 Diagrama de despliegue

A continuación, se muestra el diagrama de despliegue de la aplicación.

En primer lugar, tenemos el cliente, representa el navegador web y permite al usuario interactuar con la aplicación y realizar peticiones al servidor. En el centro del diagrama se puede ver el servidor, este es el encargado de ejecutar la aplicación web y procesar las peticiones; en último lugar está la base de datos que es la responsable de la persistencia de datos en la aplicación.

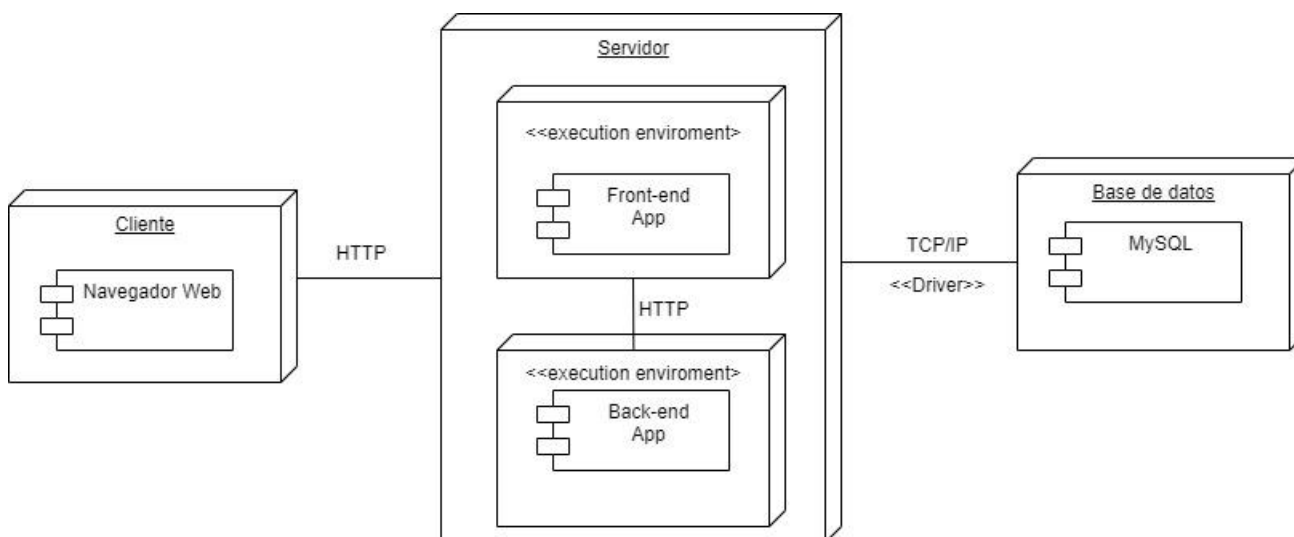


Ilustración 61. Diagrama de despliegue

6.3.2 DSI 5.3 Revisión de la Interfaz de Usuario

En este apartado se muestra el diseño detallado de la UI siguiendo el prototipo creado en la sección 5.6.1 Descripción de la Interfaz. La descripción del diseño de las pantallas se divide según los roles de los usuarios.

Los colores elegidos para la aplicación corresponden con el color corporativo de la compañía Renfe.

Cuando se accede a la aplicación se muestra la página de inicio que se observa en la

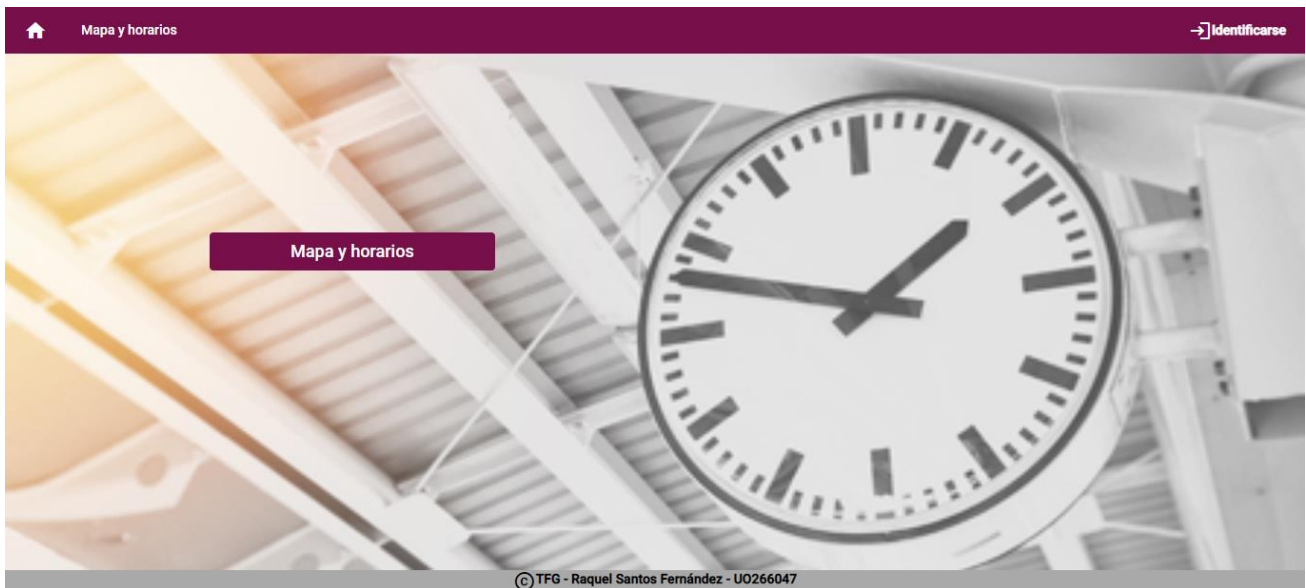


Ilustración 62. Pantalla de inicio

6.3.2.1 Usuario no identificado

6.3.2.1.1 Consulta de horarios

Para acceder a la página de consulta de horarios el usuario debe seleccionar el botón “Mapa y horarios” que se muestra en la barra superior de navegación.

En la siguiente pantalla el usuario no identificado podrá consultar el mapa con las estaciones y consultar los horarios entre un origen y un destino. Para consultar los horarios el usuario debe seleccionar en los desplegados las estaciones deseadas y pulsar sobre el botón “Consultar”. Los horarios se mostrarán por pantalla.

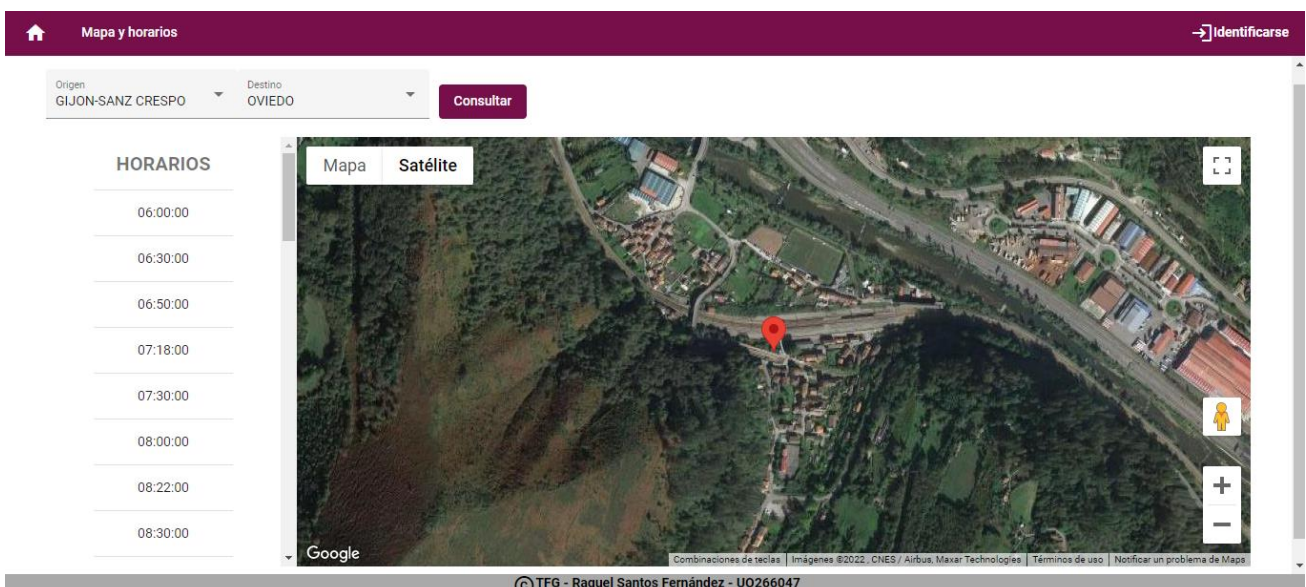


Ilustración 63. Pantalla consulta de horarios

6.3.2.1.2 Inicio de sesión

Para iniciar sesión el usuario debe seleccionar el botón “Identificarse” de la barra de navegación superior.

Introduciendo el nombre de usuario y la contraseña correspondientes y pulsando el botón “Login”, tanto los administradores como los usuarios empleados accederán a la aplicación siendo redirigidos a la página de inicio de nuevo.

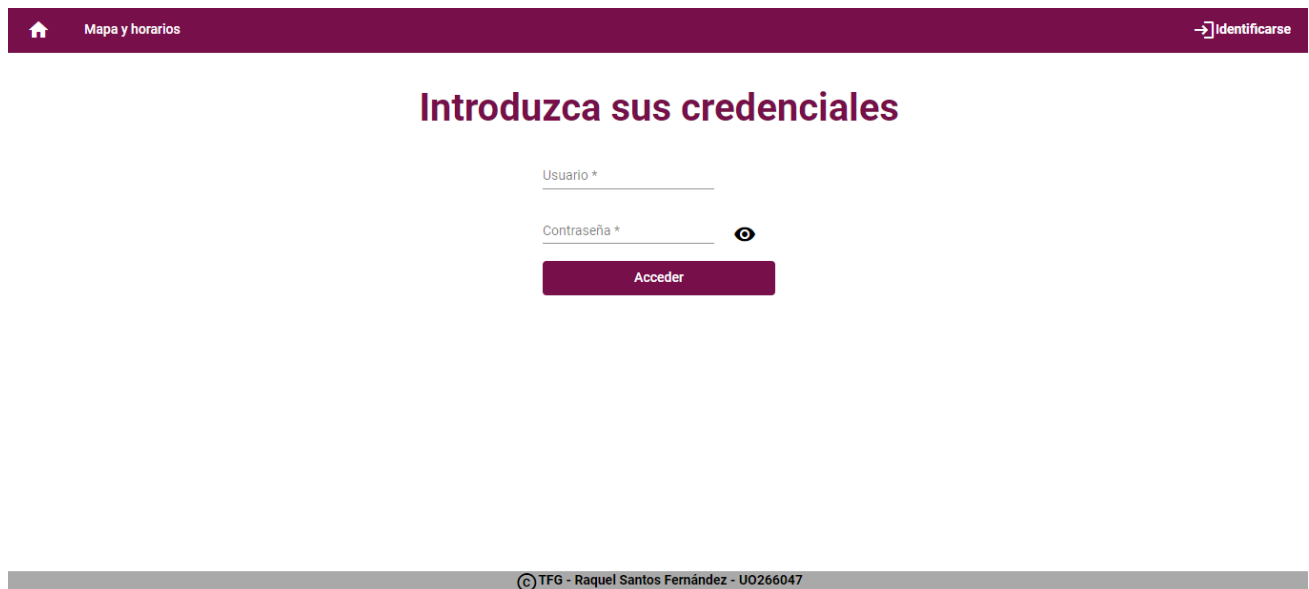


Ilustración 64. Pantalla inicio de sesión

6.3.2.2 Usuario empleado trabajador identificado

6.3.2.2.1 Pantalla de inicio - trabajador identificado

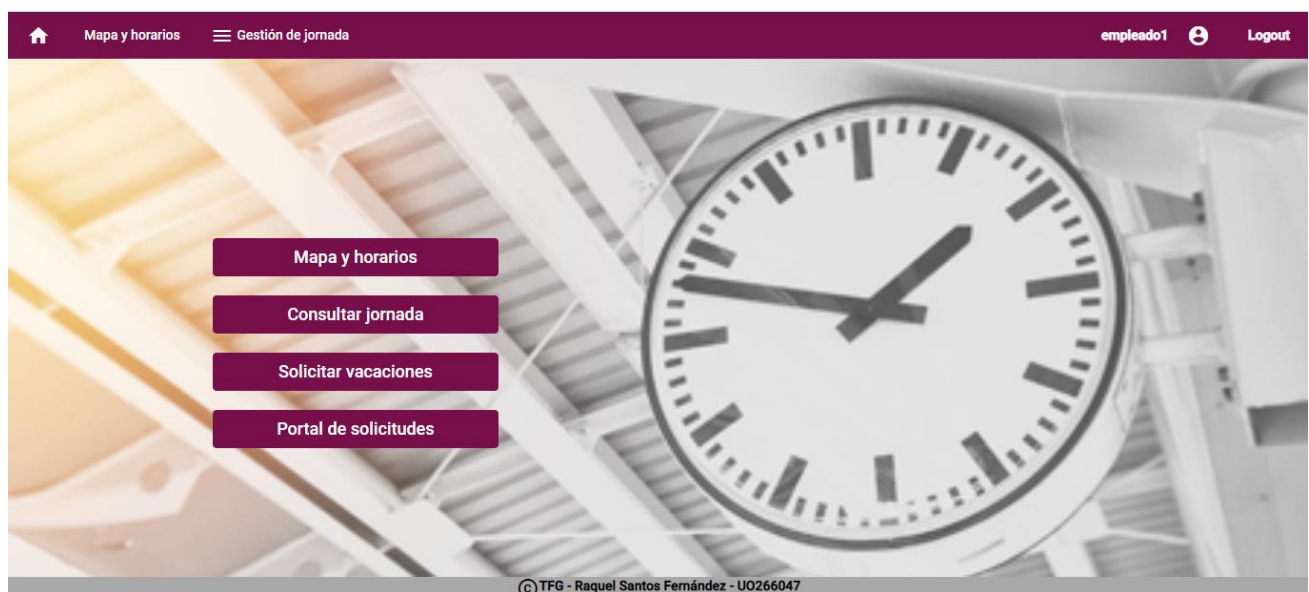


Ilustración 65. Pantalla de inicio - trabajador identificado

6.3.2.2.2 Consultar jornada

Para consultar la jornada laboral el empleado debe seleccionar “Gestión de jornada” de la barra de navegación superior y pulsar la opción de consultar jornada en el menú desplegable.

Seleccionando un día libre en el calendario el empleado podrá ver la jornada laboral asignada a ese día. Adicionalmente cumplimentando el formulario de solicitar día libre, situado dentro del desplegable con el mismo nombre, y pulsando sobre el botón “Solicitar” podrá enviar una nueva solicitud.



© TFG - Raquel Santos Fernández - UO266047

Ilustración 66. Pantalla consultar jornada

6.3.2.2.3 Solicitar vacaciones

Para solicitar las vacaciones el empleado debe seleccionar “Gestión de jornada” de la barra de navegación superior y pulsar la opción de solicitar vacaciones en el menú desplegable.

En esta pantalla existen dos posibilidades.

- El empleado puede no tener ninguna solicitud de vacaciones realizada (Ilustración 67. Pantalla solicitud de vacaciones caso 1 Ilustración 67). En este caso el empleado podrá solicitar un periodo de vacaciones en la tabla y solicitarlo pulsando el botón “Solicitar”.
- El empleado puede tener una o varias solicitudes de vacaciones (Ilustración 68). En este caso el empleado puede ver las solicitudes de vacaciones que ha realizado y en el estado en el que se encuentra cada una de ellas. Si todas las solicitudes están en estado “RECHAZADA” el empleado puede volver a seleccionar un periodo pulsando el botón “Solicitar vacaciones”.



Mapa y horarios | Gestión de jornada | empleado1 | Logout

Solicitud de vacaciones

Selecciona un periodo de vacaciones

Periodo de invierno	Periodo de verano
21/12/2022 - 09/01/2023	01/06/2022 - 15/06/2022
01/10/2022 - 20/10/2022	01/07/2022 - 15/07/2022
01/12/2022 - 20/12/2022	16/06/2022 - 30/06/2022
21/10/2022 - 08/11/2022	16/07/2022 - 31/07/2022
01/09/2022 - 15/09/2022	12/05/2022 - 31/05/2022
01/08/2022 - 15/08/2022	12/03/2022 - 31/03/2022
16/09/2022 - 30/09/2022	02/04/2022 - 21/04/2022
16/08/2022 - 31/08/2022	23/04/2022 - 11/05/2022

Solicitar

© TFG - Raquel Santos Fernández - UO266047

Ilustración 67. Pantalla solicitud de vacaciones caso 1



Mapa y horarios | Gestión de jornada | empleado1 | Logout

Solicitud de vacaciones

Las solicitudes de vacaciones existentes son:

Periodo	Estado
01-06-2022 - 15-06-2022	RECHAZADA
21-10-2022 - 08-11-2022	RECHAZADA
16-07-2022 - 31-07-2022	RECHAZADA
21-12-2022 - 09-01-2023	RECHAZADA

Solicitar vacaciones

© TFG - Raquel Santos Fernández - UO266047

Ilustración 68. Pantalla solicitud de vacaciones caso 2

6.3.2.2.4 Portal de solicitudes

Para consultar el portal de solicitudes el empleado debe seleccionar “Gestión de jornada” de la barra de navegación superior y pulsar la opción con este nombre en el menú desplegable.

En esta pantalla se pueden comprobar todas las solicitudes que ha realizado el empleado, aceptar las solicitudes de intercambio propuestas por otro empleado, pulsando el botón “Aceptar” de la solicitud deseada y crear una nueva solicitud haciendo clic en el botón “Solicitar intercambio”. Al pulsar el botón para crear una nueva solicitud se abrirá una ventana emergente para cumplimentar el formulario indicando el día que se desea tener libre, el nuevo día que se desea trabajar a cambio y el motivo de la ausencia (Ilustración 70).

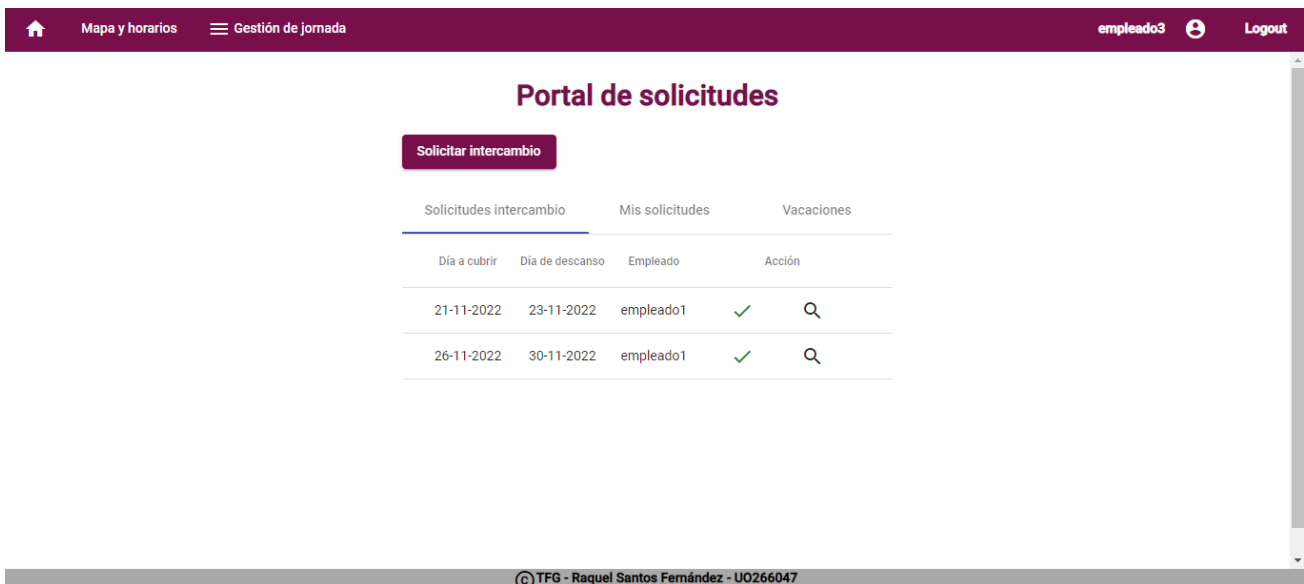


Ilustración 69. Pantalla portal de solicitudes

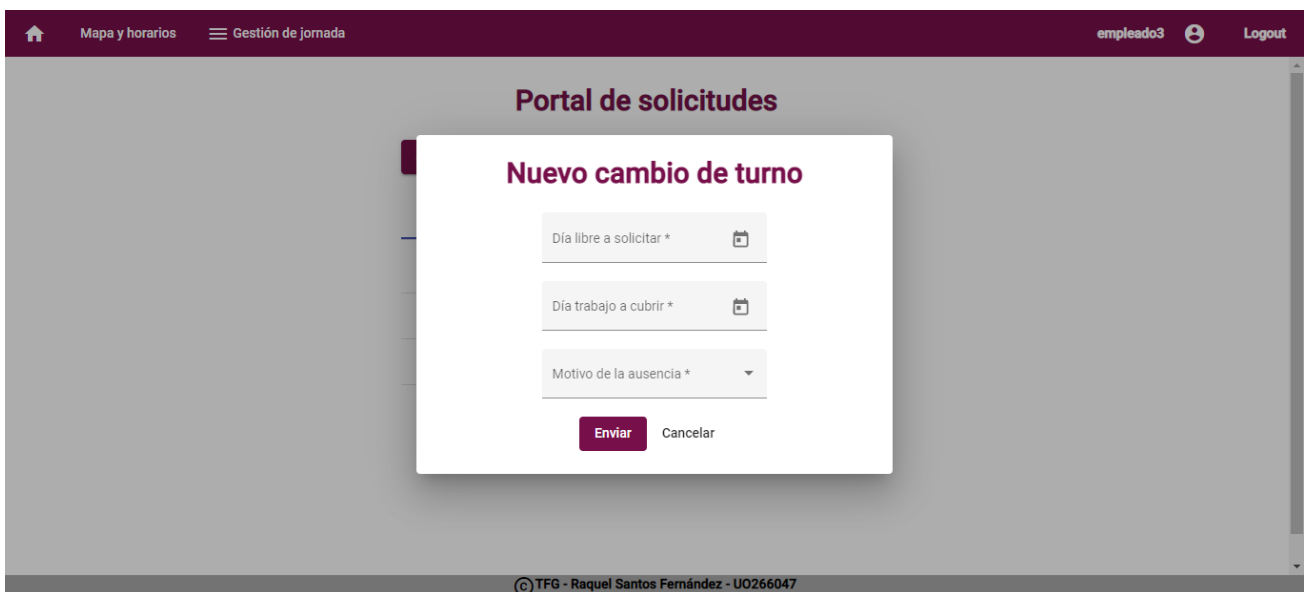


Ilustración 70. Pantalla portal de solicitudes - nueva solicitud

6.3.2.2.5 Ver perfil

Para acceder a la pantalla de ver perfil el empleado debe seleccionar el botón con silueta de persona situado en la barra de navegación superior. Este botón dispone de una etiqueta emergente o tooltip cuando el ratón se pasa por encima.

En esta página podrá consultar toda su información personal y modificar la contraseña pulsando en el botón "Modificar contraseña".

Identificado como empleado3

Nombre	Empleado3
Apellidos	Prueba
Nombre de usuario	empleado3
Rol	MAQUINISTA
Días libres disponibles	100
Modificar contraseña	

© TFG - Raquel Santos Fernández - UO266047

Ilustración 71. Pantalla ver perfil

6.3.2.3 Usuario administrador identificado

6.3.2.3.1 Ver solicitudes

Un empleado que ha iniciado sesión como usuario administrador puede ver las solicitudes realizadas por los empleados pulsando en el botón “Ver solicitudes” de la barra superior de navegación.

Seleccionando un día del calendario puede obtener un resumen de las jornadas asignadas para esa fecha.

Situado en la parte derecha de la pantalla el administrador puede ver una tabla paginada con todas las solicitudes existentes. Estas pueden ser filtradas por motivo de la solicitud y por empleado que la solicita. Para aceptar o rechazar una solicitud el administrador debe pulsar sobre el botón con el nombre correspondiente de la solicitud deseada.



Fecha	Motivo	Empleado		
25-11-2022	LICENCIA	empleado1	✓	✗
22-11-2022	LICENCIA	empleado1	✓	✗
01-06-2022 – 15-06-2022	VACACIONES	empleado3	✓	✗
21-12-2022 – 09-01-2023	VACACIONES	empleado3	✓	✗

Ilustración 72. Pantalla ver solicitudes

6.3.2.3.2 Gestionar usuarios y jornadas

Para acceder a la pantalla de gestionar usuarios y jornadas el usuario debe seleccionar el botón con dicho nombre en el menú superior.

En primer lugar, en esta pantalla se podrá buscar por nombre de usuario un empleado en la tabla y añadir un nuevo empleado utilizando el botón “Nuevo usuario”; una vez se selecciona el botón, se abre la ventana emergente de la Ilustración 74 con un formulario para rellenar.

Además, el administrador podrá consultar las jornadas y tareas asignadas a los empleados, a un día concreto o a ambos. En caso de querer añadir una nueva tarea el usuario debe seleccionar un empleado de la lista, un día del calendario y pulsar el botón “Nueva tarea”. Esta acción hará que se abra la ventana emergente de la Ilustración 75 con el formulario a cumplimentar.

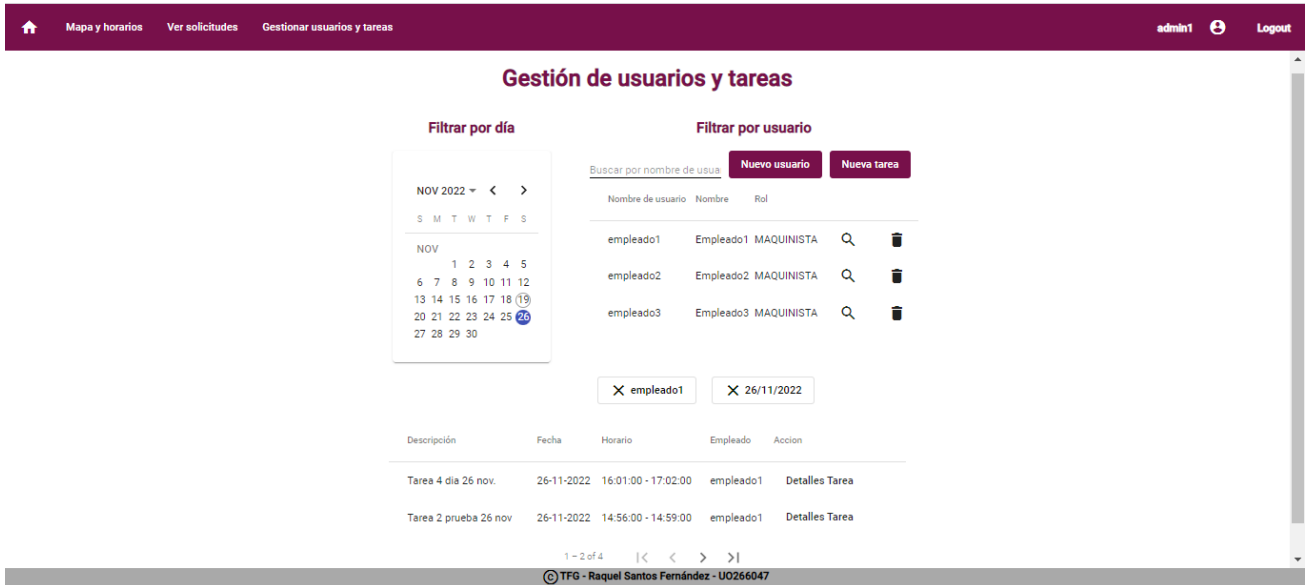


Ilustración 73. Pantalla gestionar usuarios y jornadas

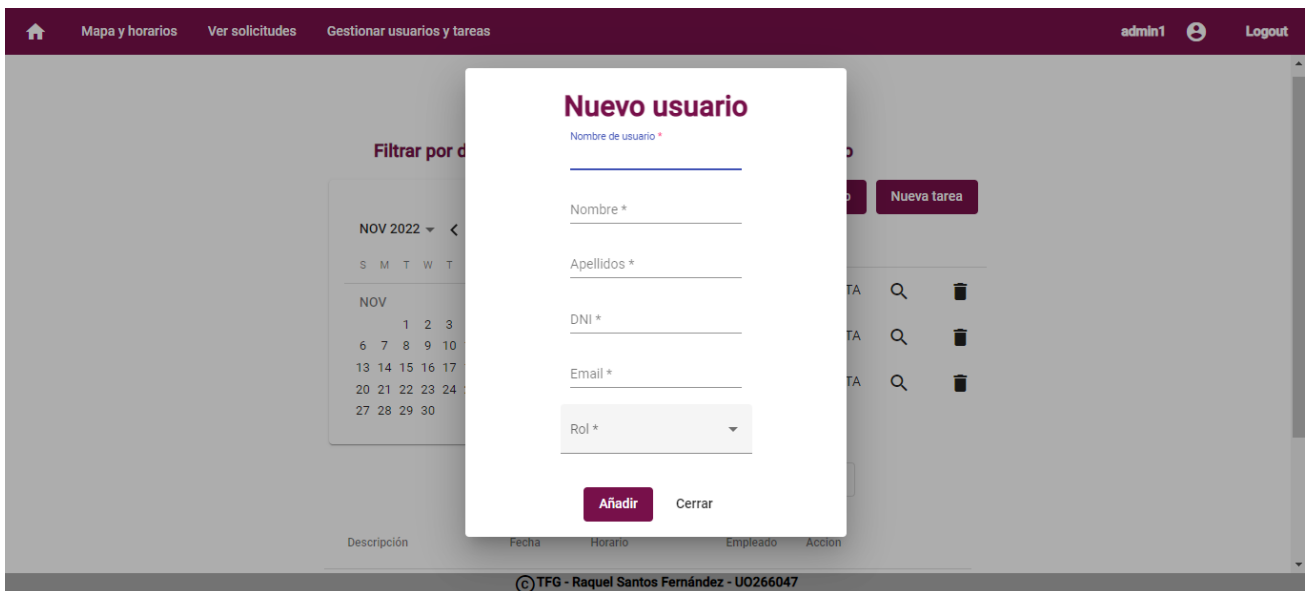


Ilustración 74. Pantalla gestionar usuarios y jornadas - Añadir usuario

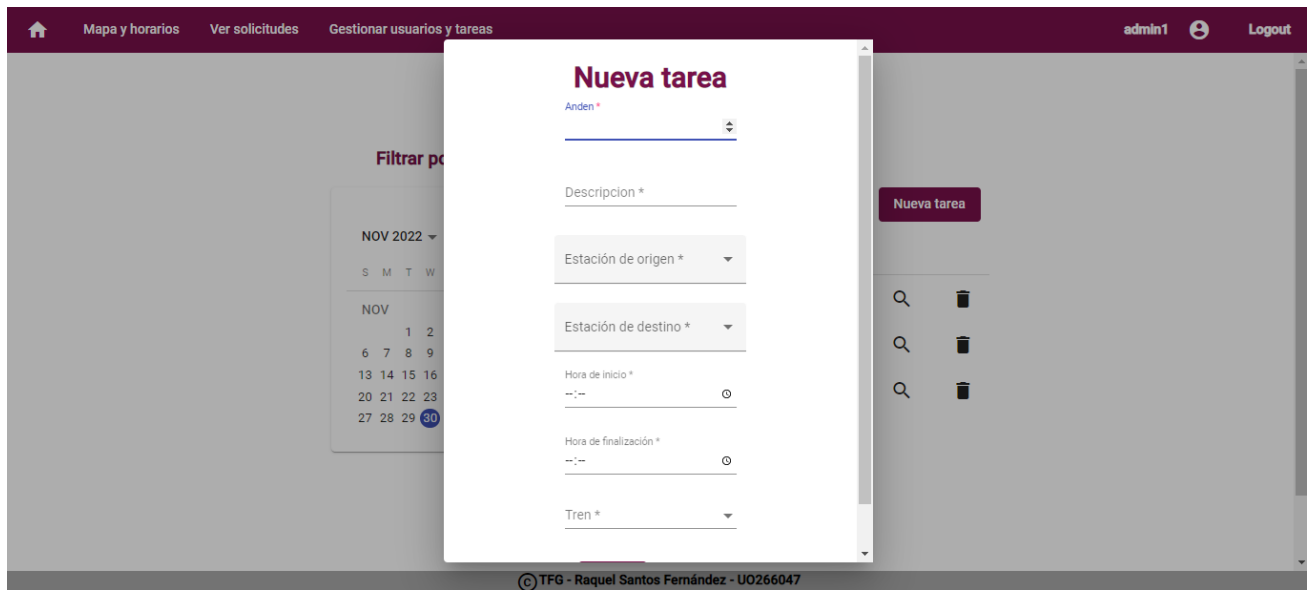


Ilustración 75. Pantalla gestionar usuarios y jornadas - Añadir tarea



6.4 DSI 6: DISEÑO FÍSICO DE DATOS

En este apartado se describe todo lo relativo al SGBD (Sistema de Gestión de Base de Datos) que se usa en la aplicación.

6.4.1 Descripción del SGBD Usado

El SGBD escogido y que se empleará es MySQL, un sistema de gestión de datos relacional muy popular, proporciona robustez, rapidez y alto rendimiento. Debido a las características mencionadas y a la mayor experiencia con el uso de la base de datos relacionales frente a otras, esta ha sido la seleccionada.

La versión utilizada será 10.6.7-MariaDB.

En cuanto a los datos almacenados en la misma cabe decir que requieren de revisión en caso de la creación de nuevas estaciones, modificación de horarios por parte de la compañía Renfe Cercanías o aparición de nuevas rutas o actualización de las mismas.

6.4.2 Integración del SGBD en Nuestro Sistema

La integración del SGBD con la aplicación se realizará a través de un Driver, este permite que la parte del back-end de la aplicación se comuniquen con la BD de manera correcta, gracias a sus protocolos de comunicación internos. Este Driver, al igual que otras configuraciones de la base de datos quedan registradas en el archivo “application.properties”.

Adicionalmente se añadirán las dependencias Maven relativas a MySQL y JPA al proyecto, concretamente al archivo “pom.xml”.

6.4.3 Diagrama E-R

A continuación, se muestra el diagrama entidad relación que muestra las 15 tablas existentes en la base de datos.

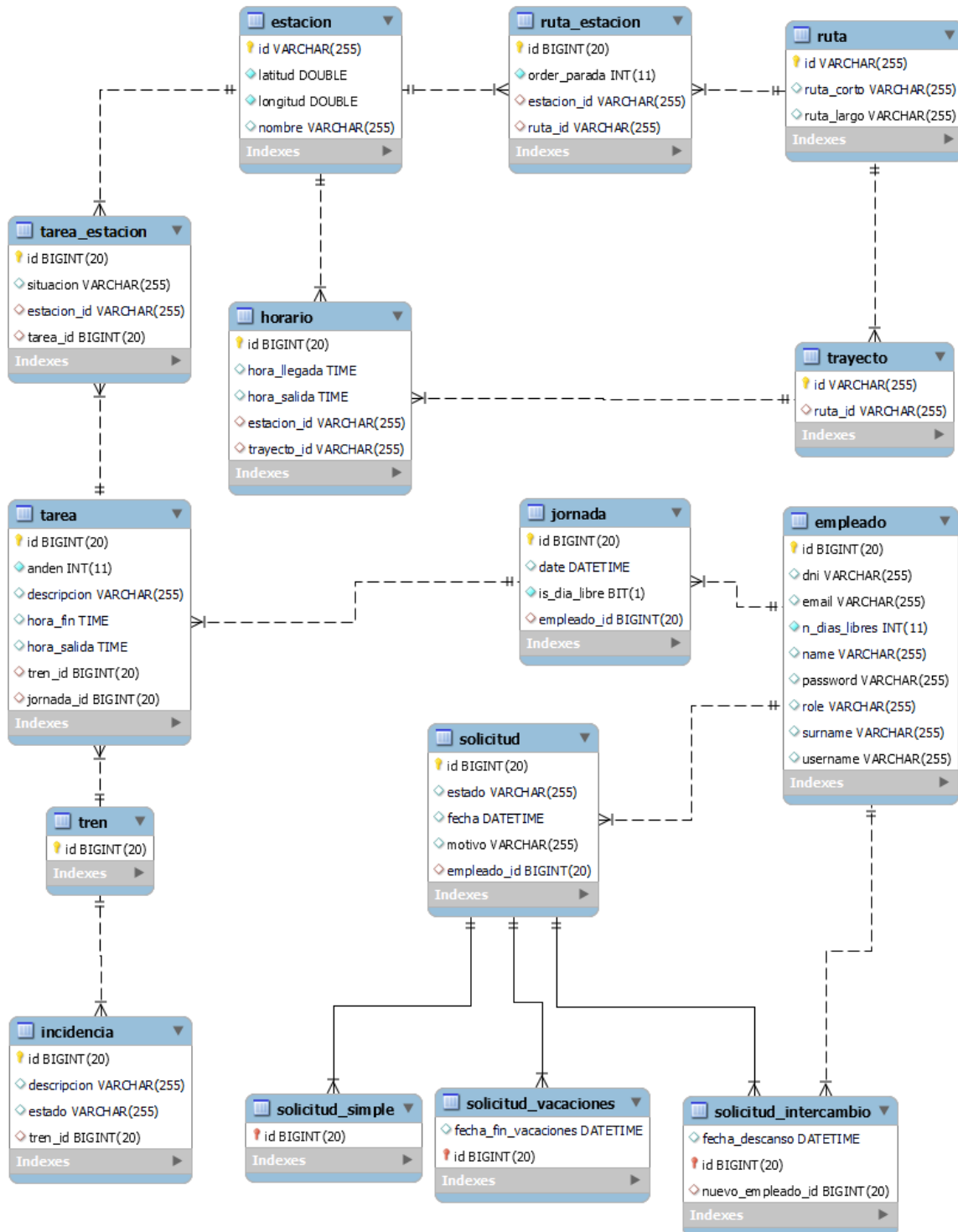


Ilustración 76. Diagrama E-R



6.5 10: ESPECIFICACIÓN TÉCNICA DEL PLAN DE PRUEBAS

Tanto para las pruebas unitarias, como para las pruebas de integración se realizan sobre un portátil HP 15-bs1xx con sistema operativo Windows 11 Home y con procesador Intel(R) Core(TM) i5-8250U CPU @ 1.60GHz, 1801 Mhz y una RAM de 8GB.

6.5.1 Pruebas Unitarias

Este apartado contiene la descripción de las pruebas unitarias que se realizarán sobre la aplicación a medida que esta se vaya construyendo, su objetivo es probar la lógica de negocio de la aplicación. Estas se realizarán empleando la herramienta de automatización JUnit.

6.5.1.1 Empleado Service

EmpleadoService	
Prueba	Resultado esperado
Añadir un nuevo usuario <ul style="list-style-type: none"> - Username: admin1 - Name: Administrador - Surname: Prueba - Email: admin1@gmail.com - DNI: 11111111A - Password: TestPassword1 - Rol: Admin - nDiasLibres: 100 	Se añade el usuario a la base de datos
Añadir un usuario repetido <ul style="list-style-type: none"> - Username: admin1 - Name: Administrador - Surname: Prueba - Email: admin1@gmail.com - DNI: 11111111A - Password: TestPassword1 - Rol: Admin - nDiasLibres: 100 	El usuario no se almacena en la base de datos
Añadir un usuario con datos inválidos <ul style="list-style-type: none"> - Username: admin2 - Name: Administrador - Surname: Prueba - Email: admin2 	El usuario no se almacena en la base de datos

<ul style="list-style-type: none"> - DNI: 11A - Password: admin - Rol: Admin - nDiasLibres: 100 	
Buscar un empleado por nombre de usuario y contraseña <ul style="list-style-type: none"> - Username: admin1 - Password: TestPassword1 	Se obtiene un empleado que coincide.
Listar todos los empleados de la base de datos	Habiendo añadido dos empleados, se obtiene una lista con 1 empleado
Eliminar un empleado existente	El empleado se elimina de la base de datos.
Eliminar un empleado no existente	Se obtiene una excepción y la longitud de la lista de empleados no varia

Tabla 83. Diseño pruebas unitarias – EmpleadoService

6.5.1.2 EstacionService

EstacionService	
Prueba	Resultado esperado
Añadir una estación <ul style="list-style-type: none"> - Id: "1" - Nombre: "Ferreros" - Latitud: 1.1 - Longitud: 1.1 	Se añade una nueva estación y se obtiene una lista con una estación más que inicialmente
Añadir una estación repetida <ul style="list-style-type: none"> - Id: "1" - Nombre: "Ferreros" - Latitud: 1.1 - Longitud: 1.1 	No se añade la estación
Añadir una estación con parámetros inválidos <ul style="list-style-type: none"> - Id: "" - Nombre: "" - Latitud: 1.1 - Longitud: 1.1 	Lanza una excepción que se recoge en el test.
Buscar una estación por id <ul style="list-style-type: none"> - Id: "1" 	Se comprueba que el resultado es distinto de null, por lo tanto la estación existe.
Eliminar una estación sin horarios asociados	Se elimina la estación y se obtiene una lista de longitud inferior en 1 que inicialmente

Tabla 84. Diseño pruebas unitarias – EstacionService

6.5.1.3 IncidenciaService

IncidenciaService	
Prueba	Resultado esperado
Añadir una incidencia válida <ul style="list-style-type: none"> - Descripción: "Incidencia prueba" - Tren: tren almacenado en la base de datos 	Se añade la incidencia, la lista de incidencias pasa a tener longitud 1
Añadir una incidencia no válida <ul style="list-style-type: none"> - Descripción: "" - Tren: null 	Lanza una excepción que se recoge en el test
Buscar todas las incidencias	Tras añadir dos incidencias a la base de datos, la lista devuelta tiene una longitud igual a 2
Buscar las incidencias pendientes	Tras cambiar el estado de una de las dos incidencias de la base de datos, la lista de incidencias pendientes es de dimensión 1
Eliminar una incidencia existente	Se elimina la incidencia y la lista de incidencias decremента su longitud en 1
Eliminar una incidencia no existente	Se lanza una excepción que se recoge en el test.

Tabla 85. Diseño de pruebas unitarias – IncidenciaService

6.5.1.4 JornadaService

JornadaService	
Prueba	Resultado esperado
Añadir una jornada válida <ul style="list-style-type: none"> - Fecha: 2022-10-17 - Empleado: Empleado existente en la BD - isDiaLibre: false 	Se almacena la jornada en la base de datos, la longitud de la lista de las jornadas existentes aumenta en 1.
Añadir una jornada inválida <ul style="list-style-type: none"> - Fecha: null - Empleado: null - isDiaLibre: false 	Lanza una excepción que es recogida en el test
Buscar una jornada por fecha. AFTER añadir dos jornadas con fecha "2022-10-17" <ul style="list-style-type: none"> - Fecha: 2022-10-17 	Devuelve una lista de longitud 2 con las dos jornadas correspondientes añadidas anteriormente.
Buscar una jornada por fecha inexistente en la BD <ul style="list-style-type: none"> - Fecha: 2020-10-17 	Se devuelve una lista vacía.



Buscar una jornada por empleado. AFTER añadir dos jornadas con el mismo empleado asignado <ul style="list-style-type: none">- Empleado: Empleado en la BD	Devuelve una lista de longitud 2 con las dos jornadas correspondientes añadidas anteriormente.
Buscar una jornada por un empleado que no tiene ninguna jornada asignada <ul style="list-style-type: none">- Empleado: Empleado en la BD	Se devuelve una lista vacía
Buscar una jornada por fecha y empleado AFTER añadir una jornada con esa fecha y ese empleado en la BD <ul style="list-style-type: none">- Fecha: 2022-10-17- Empleado: Empleado en la BD	Se devuelve una lista de un elemento cuya fecha y empleado coinciden con los buscados
Marcar como día libre una fecha en la cual existe jornada. AFTER: <ol style="list-style-type: none">1. Añadir una jornada con esa fecha para ese empleado2. Añadir una tarea a la jornada<ul style="list-style-type: none">- Fecha: 2022-10-17- Empleado: Empleado en la BD	Se actualiza el valor del campo isDiaLibre convirtiendo el día en día libre y se comprueba que la lista de tareas para la jornada tiene una longitud igual a 0
Marcar como día libre una fecha sin jornada para un empleado existente	Se crea una nueva jornada para la fecha indicada y se asigna como día libre
Buscar tareas de un empleado no existente en la BD por fecha <ul style="list-style-type: none">- IdEmpleado: 0000- Fecha: 2022-10-17	No se obtiene ninguna Tarea
Buscar tareas de un empleado por fecha. AFTER añadir el empleado en la BD y añadir una jornada con una tarea <ul style="list-style-type: none">- IdEmpleado: Id de un empleado en la BD- Fecha: 2022-10-17	Se devuelve una lista con una tarea.
Buscar tareas de un empleado sin tareas por fecha. AFTER añadir el empleado en la BD <ul style="list-style-type: none">- IdEmpleado: Id de un empleado en la BD- Fecha: 2022-10-17	Se devuelve una lista vacía
Buscar tareas de un empleado por fecha inválida. <ul style="list-style-type: none">- IdEmpleado: null	Se devuelve una lista vacía

- Fecha: null	
Buscar tareas de un empleado sin jornada asignada por fecha AFTER añadir el empleado en la base de datos	Se obtiene una lista de longitud 0
- IdEmpleado: Id del empleado en la BD - Fecha: 2022-10-17	
Buscar tareas de un empleado sin tareas dada una fecha, existiendo tareas en esa fecha para otro empleado distinto. AFTER añadir una nueva jornada en esa fecha a un empleado existente	Se obtiene una lista de longitud 0
- IdEmpleado: Id del empleado en la BD - Fecha: 2022-10-17	
Buscar tareas de un empleado para una fecha en la que no tiene jornada, existiendo jornada para otra fecha. AFTER añadir una jornada con fecha distinta a la pasada por parámetro para ese mismo empleado	Se obtiene una lista de longitud 0
- IdEmpleado: Id del empleado en la BD - Fecha: 2022-10-17	
Reasignar una jornada para un empleado que no tiene jornada asignada en el día indicado	No se reasigna la jornada y el estado de la solicitud permanece "PENDIENTE"
- SolicitudIntercambio: Solicitud en la BD	
Reasignar una jornada, siendo el nuevo empleado null.	No se reasigna la jornada y el estado de la solicitud permanece "PENDIENTE"
- SolicitudIntercambio: Solicitud en la BD	
Reasignar una jornada válida	Se intercambian las jornadas de los empleados implicados.
- SolicitudIntercambio: Solicitud en la BD	Se marca la solicitud como "REASIGNADA"

Tabla 86. Diseño pruebas unitarias – JornadaService

6.5.1.5 RutaService

RutaService	
Prueba	Resultado esperado
Añadir una ruta válida	Se almacena la ruta en la BD
- Id: "1" - Ruta_corto: "p" - Ruta_largo: "prueba"	
Añadir una ruta inválida	Se lanza una excepción que se recoge en el test

<ul style="list-style-type: none"> - Id: "" - Ruta_corto: "" - Ruta_largo: "" 	
Añadir una ruta repetida <ul style="list-style-type: none"> - Id: "1" - Ruta_corto: "p" - Ruta_largo: "prueba" 	No se añade la ruta a la BD
Buscar una ruta por id válido <ul style="list-style-type: none"> - Id: "1" 	Se encuentra la ruta buscada
Buscar una ruta por id no existente <ul style="list-style-type: none"> - Id: "0000" 	No se encuentra ninguna ruta en la BD
Buscar una ruta entre dos estaciones no existentes <ul style="list-style-type: none"> - OrigenId: "0000" - Destinoid: "0001" 	No se encuentra ninguna ruta en la BD
Buscar una ruta entre dos estaciones con ruta AFTER añadir dos rutas con ids 1 y 2 respectivamente y una ruta entre ellas <ul style="list-style-type: none"> - OrigenId: "1" - Destinoid: "2" 	Se encuentra una ruta que cumple la condición de incluir ambas estaciones
Buscar una ruta entre dos estaciones sin ruta AFTER añadir dos estaciones con ids 1 y 2 <ul style="list-style-type: none"> - OrigenId: "1" - Destinoid: "2" 	No se encuentra una ruta entre las dos estaciones existentes en la BD

Tabla 87. Diseño de pruebas unitarias - RutaService

6.5.1.6 SolicitudService

SolicitudService	
Prueba	Resultado esperado
Añadir una solicitud null <ul style="list-style-type: none"> - solicitud: null 	No se añade la solicitud a la BD
Añadir una solicitud válida a la BD <ul style="list-style-type: none"> - fecha: "2022-10-17" - motivo: "Licencia" - empleado: Empleado en la BD 	Se añade correctamente a la BD
Rechazar una solicitud AFTER añadir una solicitud simple a la BD <ul style="list-style-type: none"> - id: Id de una solicitud en la BD 	El estado de la solicitud pasa a ser "RECHAZADA"



Buscar todas las solicitudes pendientes sin que haya solicitudes	No se obtiene ninguna solicitud
Buscar todas las solicitudes pendientes AFTER añadir una solicitud simple y una solicitud de vacaciones al sistema.	Se encuentran dos solicitudes con estado "PENDIENTE"
Buscar todas las solicitudes pendientes. AFTER añadir una solicitud simple y una solicitud de vacaciones al sistema y rechazar la solicitud simple	Se encuentra una sola solicitud con estado "PENDIENTE"
Buscar todas las solicitudes pendientes con solamente una solicitud de intercambio en el sistema AFTER añadir una solicitud de intercambio a la BD	No se obtiene ninguna solicitud con estado "PENDIENTE"
Añadir una solicitud de intercambio pasando como parámetro una solicitud simple AFTER añadir un empleado a la BD <ul style="list-style-type: none">- solicitud: solicitudSimple- fecha: "2022-10-17"- motivo: "Licencia"- empleado: Empleado en la BD	No se añade la solicitud a la BD
Añadir una solicitud de intercambio inválida <ul style="list-style-type: none">- solicitud: null	No se añade la solicitud a la BD
Añadir una solicitud de intercambio válida AFTER añadir un empleado a la BD <ul style="list-style-type: none">- fecha: "2022-10-17"- motivo: "Licencia"- empleado: Empleado en la BD- fechaDescanso: "2022-11-17"- nuevoEmpleado: null	Se añade la solicitud de intercambio a la BD
Añadir una solicitud de vacaciones inválida <ul style="list-style-type: none">- solicitud: null	La solicitud no se añade a la BD
Añadir una solicitud de vacaciones válida AFTER añadir un empleado a la BD <ul style="list-style-type: none">- fecha: "2022-10-17"- motivo: "Licencia"- empleado: Empleado en la BD	Se añade la solicitud de vacaciones a la BD
Buscar una solicitud de vacaciones por id de empleado con id inválido <ul style="list-style-type: none">- idEmpleado: null	No se obtiene ninguna solicitud



Buscar una solicitud de vacaciones con id de empleado no existente - idEmpleado: "0000"	No se obtiene ninguna solicitud
Buscar solicitudes de vacaciones por id de empleado existente AFTER añadir un empleado y dos solicitudes de vacaciones a la BD. - idEmpleado: Id de un empleado en la BD	Se obtiene una lista con dos solicitudes de vacaciones
Buscar las solicitudes simples y de intercambio realizadas por el mismo empleado que las busca AFTER añadir dos empleados a la BD. El primero de ellos realiza dos solicitudes y el segundo una - idEmpleado: Empleado en la BD que ha realizado dos solicitudes	Se obtiene una lista con dos solicitudes, una de tipo simple y otra de tipo intercambio
Buscar las solicitudes propias con id de empleado inválido - idEmpleado: null	Se obtiene una lista vacía
Buscar las solicitudes propias para un empleado que no está en la BD - idEmpleado: "0000"	Se obtiene una lista vacía
Buscar las solicitudes por fecha y por empleado con datos inválidos - fecha: "" - idEmpleado: null	Se obtiene una lista vacía
Buscar las solicitudes simples y de intercambio por fecha y empleado AFTER añadir un empleado y dos solicitudes de tipos simples e intercambio a la BD - fecha: "2022-10-17" - idEmpleado: Id de un empleado en la BD	Se obtiene una lista con dos solicitudes
Buscar las solicitudes de intercambio de un empleado inválido con estado "PENDIENTE" - empleado: null	Se obtiene una lista vacía
Buscar las solicitudes de intercambio pendientes de otros empleados, con fecha no coincidente con una jornada existente. AFTER añadir dos empleados y una solicitud del primero en la BD	Se obtiene una lista vacía debido a la no coincidencia con una jornada del empleado



<p>- empleado: Segundo empleado almacenado en la BD</p>	
<p>Buscar las solicitudes de intercambio pendientes de otros empleados, con fecha coincidente con una jornada. AFTER añadir dos empleados, una solicitud del segundo en la BD y una jornada para el primero con la misma fecha que la solicitud</p> <p>- empleado: Primer empleado almacenado en la BD</p>	<p>Se obtiene una lista con una solicitud, la creada por el segundo empleado.</p>
<p>Buscar las solicitudes de intercambio pendientes de otros empleados, con fecha coincidente con una jornada que es día libre. AFTER añadir dos empleados, una solicitud del segundo en la BD y una jornada para el primero con la misma fecha que la solicitud.</p> <p>- empleado: Primer empleado almacenado en la BD</p>	<p>Se obtiene una lista vacía.</p>
<p>Buscar las solicitudes de intercambio pendientes de otros empleados, con fecha coincidente con una jornada. AFTER añadir dos empleados, una solicitud del segundo en la BD y dos jornadas para el primero coincidiendo con la fecha de descanso y la fecha a cubrir</p> <p>- empleado: Primer empleado almacenado en la BD</p>	<p>Se obtiene una lista vacía debido a la existencia de jornada para el día a cubrir y el día libre.</p>
<p>Aceptar una solicitud de tipo no válido (intercambio) AFTER añadir un empleado y una solicitud de intercambio a la BD</p> <p>- solicitud: SolicitudIntercambio en la BD</p>	<p>La solicitud no cambia de estado y sigue en "PENDIENTE" y la jornada del empleado no se ve afectada.</p>
<p>Aceptar una solicitud de tipo simple. AFTER añadir un empleado, una solicitud simple a la BD y una jornada con tareas para la fecha de la solicitud</p> <p>- solicitud: SolicitudSimple en la BD</p>	<p>La solicitud pasa a estar en estado "ACEPTADA". La jornada se marca como libre, todas las tareas asociadas se eliminan</p>
<p>Aceptar una solicitud de tipo vacaciones AFTER añadir un empleado y una solicitud de vacaciones a la BD</p>	<p>La solicitud pasa a estado "ACEPTADA", las jornadas entre las fechas de inicio y fin se</p>

- solicitud: SolicitudVacaciones en la BD	marcan como libres y las tareas asociadas a esos días se eliminan.
--	--

Tabla 88. Diseño de pruebas unitarias - SolicitudService

6.5.1.7 TareaService

TareaService	
Prueba	Resultado esperado
Buscar tarea por id existente AFTER añadir un empleado y una jornada con tareas a la BD <ul style="list-style-type: none"> - idTarea: Id una tarea en la BD 	Se obtiene una tarea con ese id
Buscar tarea por id inexistente <ul style="list-style-type: none"> - idTarea: "0000" 	Se obtiene null
Eliminar una tarea AFTER añadir un empleado y una jornada con tareas a la BD <ul style="list-style-type: none"> - tarea: Tarea almacenada en la BD 	Se comprueba que al buscar la tarea con ese id, no se encuentra ninguna.
Buscar todas las tareas siendo estas inexistentes	Se obtiene una lista vacía
Buscar todas las tareas existentes AFTER añadir un empleado y una jornada con dos tareas a la BD	Se obtiene una lista con dos tareas

Tabla 89. Diseño de pruebas unitarias - TareaService

6.5.1.8 TareaEstacionService

TareaEstacionService	
Prueba	Resultado esperado
Añadir una nueva tarea_estacion AFTER añadir una estación y una tarea a la BD <ul style="list-style-type: none"> - situación: INICIO - estación: Estación en la BD - tarea: Tarea en la BD 	Se añade una nueva tarea_estacion a la BD
Añadir una nueva tarea_estacion con datos inválidos <ul style="list-style-type: none"> - situación: null - estación: null - tarea: null 	Devuelve una excepción de parámetros ilegales que es recogida en el test

Tabla 90. Diseño de pruebas unitarias - TareaEstacionService

6.5.1.9 TrenService

TrenService	
Prueba	Resultado esperado
Añadir un tren - tren: nuevo tren	Se añade el tren a la BD
Añadir un tren null - tren: null	El tren no se añade a la BD
Buscar un tren por id AFTER añadir un tren a la BD - trenId: Id del tren en la BD	Se obtiene un tren con id coincidente al parámetro
Buscar un tren por id inexistente - trenId: "0000"	No se obtienen ningún tren

Tabla 91. Diseño de pruebas unitarias – TrenService

6.5.2 Pruebas de Integración y del Sistema

En este apartado se desarrolla el diseño de las pruebas de integración y del sistema. Su objetivo es garantizar que el sistema funciona correctamente.

Estas pruebas se separan en tren subapartados dependiendo de los actores del sistema

6.5.2.1 Usuario no identificado

A continuación, se indican las pruebas relacionadas con las funcionalidades del usuario no identificado.

Pruebas usuario no identificado	
Prueba	Resultado esperado
Acceder a la página de inicio	Se accede a la página principal y contiene lo esperado
Acceder a la consulta de horarios	Se accede a la página de consulta y se muestra el mapa y los selectores de estaciones para la consulta
Consultar horarios entre estaciones con ruta - origen: PUENTE DE LOS FIERROS - destino: POLA DE LENA	Se redirige a una nueva página donde se muestran los horarios correspondientes
Consultar horarios entre dos estaciones sin ruta - origen: GIJÓN - destino: AVILÉS	Se muestra un mensaje de incompatibilidad de estaciones en ruta

Iniciar sesión <ul style="list-style-type: none"> - Nombre de usuario: admin1 - Contraseña: Password10 	Se autentica al usuario introducido y se redirige a la página principal
Iniciar sesión: campos vacíos	No se autentica el usuario y se muestra un mensaje de error
Iniciar sesión: campos erróneos <ul style="list-style-type: none"> - Nombre de usuario: a - Contraseña: Password10 	No se autentica el usuario y se muestra un mensaje de error

Tabla 92. Diseño de pruebas de integración - Usuario no identificado

6.5.2.2 Usuario trabajador identificado

A continuación se indican las pruebas relacionadas con las funcionalidades del usuario trabajador identificado.

Pruebas usuario administrador	
Prueba	Resultado esperado
Consultar una jornada <ul style="list-style-type: none"> - Fecha: 30 noviembre 2022 	Se muestra una lista de tareas para la jornada y sus detalles en una ventana modal.
Añadir una incidencia <ul style="list-style-type: none"> - Fecha tarea: 30 noviembre 2022 - Descripción: Prueba de integración add incidencia 	Se añade una incidencia a la lista de incidencias existentes
Solicitar un día libre <ul style="list-style-type: none"> - Fecha: 29 noviembre 2022 	La jornada aparece marcada como día libre
Solicitar vacaciones <ul style="list-style-type: none"> - Periodo: 3 	Se elimina la posibilidad de seleccionar vacaciones de nuevo y se muestran los periodos solicitados
Solicitar vacaciones repetidas	No se muestra la lista de periodos de vacaciones a seleccionar
Solicitar intercambio <ul style="list-style-type: none"> - Día a solicitar: 30/11/2022 - Día a cubrir: 25/12/2022 	Se muestra un mensaje indicando que la solicitud ha sido añadida con éxito
Solicitar intercambio sin jornada coincidente <ul style="list-style-type: none"> - Día a solicitar: 20/12/2022 - Día a cubrir: 25/12/2022 	Se muestra un mensaje explicando el problema que impide realizar la solicitud
Consultar mis solicitudes	Se muestra una tabla con las solicitudes propias existentes
Consultar vacaciones	Se muestran los periodos de vacaciones solicitados

Ver perfil	Se redirige al usuario a una página donde se muestran los detalles del empleado
Modificar contraseña - Nueva contraseña: Password5 - Repetir contraseña: Password5	Se muestra un mensaje indicando que la contraseña ha sido modificada con éxito
Modificar contraseña: comprobación incorrecta - Nueva contraseña: Password1 - Repetir contraseña: Password5	Se muestra un mensaje indicando que las contraseñas no coinciden
Aceptar solicitud intercambio - Solicitud: 1	Se elimina la solicitud de la lista de solicitudes

Tabla 93. Diseño de pruebas de integración - Usuario administrador

6.5.2.3 Usuario administrador identificado

Por último, se describen las pruebas relacionadas con las funcionalidades del usuario administrador identificado.

Pruebas usuario administrador	
Prueba	Resultado esperado
Buscar un empleado en el filtro - Nombre de usuario: empleado1	La tabla de empleados queda filtrada por la cadena introducida
Añadir un empleado - Nombre de usuario: empleado2 - Nombre: Empleado2 - Apellido: Prueba - DNI: 33333333C - Email: empleado2@gmail.com - Rol: MAQUINISTA	Se añade el nuevo empleado a la lista de empleados
Eliminar un empleado - Empleado: empleado2	Se elimina el empleado de la lista de empleados
Consultar los detalles de un empleado - Empleado: empleado1	Se muestra la ventana modal con los detalles del empleado seleccionado
Añadir una tarea - Fecha: 30/11/2022 - Empleado: empleado1 - Andén: 1 - Descripción: Prueba integración 30nov - Origen: PUENTE DE LOS FIERROS - Destino: POLA DE LENA - Hora inicio: 10:00	Se añade una nueva tarea a la lista de tareas

- Hora fin: 12:00 - Identificador del tren: 1	
Añadir una tarea sin seleccionar un empleado	Se muestra un mensaje emergente indicando que se debe seleccionar antes un empleado
Consultar los detalles de una tarea - Fecha: 30/11/2022 - Empleado: empleado1	Se muestra la venta modal con los detalles de la tarea
Ver las solicitudes pendientes	Se accede a la página de las solicitudes con una tabla que contiene todas las solicitudes pendientes
Aceptar solicitud simple - Fecha: 29/11/2022	Se elimina la solicitud de la lista de solicitudes
Rechazar solicitud AFTER volver a solicitar día libre fecha 29/11/2022 - Fecha: 29/11/2022	Se elimina la solicitud de la lista de solicitudes
Aceptar solicitud vacaciones - Periodo: 2022-12-01 – 2022-12-20	Se elimina la solicitud de la lista de solicitudes

Tabla 94. Diseño de pruebas de integración - Usuario administrador

6.5.3 Pruebas de Usabilidad

En este apartado se lleva a cabo el diseño de pruebas de usabilidad. Mediante este tipo de pruebas podemos determinar la satisfacción del cliente con el producto final, evaluando así el funcionamiento real de la aplicación.

Las pruebas de usabilidad serán realizadas de manera presencial a 5 usuarios con diferentes conocimientos sobre informática.

6.5.3.1 Diseño de Cuestionarios

Los cuestionarios realizados para las pruebas constan de 4 apartados:

- 1. Preguntas de carácter general:** Estas preguntas permiten determinar los conocimientos informáticos que tiene el usuario.
- 2. Actividades guiadas:** El usuario realizará una serie de actividades guiadas para probar la aplicación.
- 3. Batería de preguntas cortas:** El usuario debe responder una serie de preguntas cortas para evaluar distintos aspectos de la aplicación.
- 4. Observaciones:** El usuario podrá indicar cualquier observación con respecto a la aplicación.



El tiempo de cumplimentación de los mismos no será superior a 15 minutos y deben ser rellenados por los cinco usuarios mencionados anteriormente.

Adicionalmente existe un cuestionario para el responsable de las pruebas donde se podrán anotar distintos aspectos detectados durante la realización de las pruebas.

6.5.3.2 Actividades de las Pruebas de Usabilidad

A continuación se detallan cada uno de los cuatro apartados anteriores:

6.5.3.2.1 Preguntas de carácter general

Preguntas de carácter general
¿Usa un ordenador frecuentemente?
<input type="checkbox"/> Todos los días
<input type="checkbox"/> Varias veces a la semana
<input type="checkbox"/> Ocasionalmente
<input type="checkbox"/> Nunca o casi nunca
¿Qué tipo de actividades realiza con el ordenador?
<input type="checkbox"/> Es parte de mi trabajo o profesión
<input type="checkbox"/> Lo uso básicamente para ocio
<input type="checkbox"/> Solo empleo aplicaciones estilo Office
<input type="checkbox"/> Únicamente leo el correo y navego ocasionalmente
¿Ha usado alguna vez software como el de esta prueba?
<input type="checkbox"/> Sí, he empleado software similar
<input type="checkbox"/> No, aunque sí empleo otros programas que me ayudan a realizar tareas similares
<input type="checkbox"/> No, nunca
¿Qué busca Vd. Principalmente en un programa?
<input type="checkbox"/> Que sea fácil de usar
<input type="checkbox"/> Que sea intuitivo
<input type="checkbox"/> Que sea rápido
<input type="checkbox"/> Que tenga todas las funciones necesarias
¿Qué nivel de conocimientos sobre informática tiene?
<input type="checkbox"/> Nulos
<input type="checkbox"/> Básicos
<input type="checkbox"/> Medios
<input type="checkbox"/> Avanzados
¿Qué edad tiene?
<input type="checkbox"/> Entre 18 y 35 años
<input type="checkbox"/> Entre 35 y 60 años



Mayor de 60 años

Tabla 95. Cuestionario de preguntas generales de usabilidad

6.5.3.2.2 Actividades guiadas

Los usuarios deben realizar la siguiente lista de actividades guiadas indicando si han podido completarlas o no. Adicionalmente podrán señalar cualquier dificultad encontrada o comentario que desean destacar.

1. Consultar el mapa con las estaciones
2. Consultar los horarios entre dos estaciones
3. Iniciar sesión
4. Cerrar sesión
5. Rol administrador:
 - a. Añadir empleado
 - b. Eliminar empleado
 - c. Filtrar empleados
 - d. Consultar los detalles de un empleado
 - e. Añadir una nueva tarea
 - f. Consultar los detalles de una tarea
 - g. Consultar las tareas para una fecha concreta
 - h. Consultar las tareas para un empleado concreto
 - i. Consultar las tareas dado un empleado y una fecha concretos
 - j. Ver solicitudes
 - k. Aceptar solicitudes
 - l. Rechazar solicitudes
6. Rol empleado:
 - a. Ver perfil
 - b. Modificar contraseña
 - c. Consultar jornada
 - d. Solicitar día libre
 - e. Solicitar vacaciones
 - f. Solicitar intercambio
 - g. Aceptar intercambio
 - h. Consultar solicitudes
7. Dificultades encontradas o comentarios

6.5.3.2.3 Preguntas Cortas sobre la Aplicación y Observaciones

A continuación, se expone el cuestionario de preguntas cortas para los usuarios.

Preguntas Cortas de Usabilidad

Funciones de la Aplicación



Facilidad de Uso	Siempre	Frecuentemente	Ocasionalmente	Nunca
<i>¿Sabe dónde está dentro de la aplicación?</i>				
<i>¿Existe ayuda para las funciones en caso de que tenga dudas?</i>				
<i>¿Le resulta sencillo el uso de la aplicación?</i>				
<i>¿Le resulta sencillo relacionar los nombres del menú con la acción a realizar deseada?</i>				
<i>¿Puede manejarse fácilmente por la aplicación?</i>				
Funcionalidad	Siempre	Frecuentemente	Ocasionalmente	Nunca
<i>¿Funciona cada tarea como Vd. Espera?</i>				
<i>¿El tiempo de respuesta de la aplicación es muy grande?</i>				
Calidad del Interfaz				
Aspectos gráficos	Muy Adecuado	Adecuado	Poco Adecuado	Nada Adecuado
<i>El tipo y tamaño de letra es</i>				
<i>Los iconos e imágenes usados son</i>				
<i>Los colores se visualizan correctamente</i>				
<i>Los colores seleccionados van acordes con el proyecto</i>				
Diseño de la Interfaz		Si	No	A veces
<i>¿Le resulta fácil de usar?</i>				
<i>¿El diseño de las pantallas es claro y atractivo?</i>				
<i>¿Cree que el programa está bien estructurado?</i>				
<i>¿Se ven correctamente todos los elementos de la aplicación?</i>				
Observaciones				
Aspectos a mejorar				

Tabla 96. Cuestionario de preguntas cortas de usabilidad

6.5.3.2.4 Cuestionario para el responsable de las Pruebas

En la siguiente tabla se muestra el cuestionario para el responsable de las pruebas, en él se evalúan aspectos del usuario que ha realizado las pruebas de la aplicación.

Aspecto Observado	Notas
El usuario comienza a trabajar de forma rápida por las tareas	
Tiempo en realizar cada tarea	
Errores leves cometidos	
Errores graves cometidos	
Dudas que han surgido	
El usuario necesita mucho seguimiento para completar las tareas	

Tabla 97. Cuestionario para el responsable de las pruebas de usabilidad

Para facilitar la cumplimentación del cuestionario se ha elaborado un formulario mediante la aplicación Microsoft Forms, al cual se puede acceder mediante el siguiente enlace o a través del código QR de la Ilustración 77

<https://forms.office.com/r/yU53BpLsYk>



Ilustración 77. Código QR para cuestionario de usabilidad



Capítulo 7 CONSTRUCCIÓN DEL SISTEMA DE INFORMACIÓN

FASE DE DESARROLLO

CSI

7.1 CSI 1: PREPARACIÓN DEL ENTORNO DE GENERACIÓN Y CONSTRUCCIÓN

En este apartado se detalla todo aquello relativo a la implementación del software desarrollado.

7.1.1 Estándares y normas seguidos

En este apartado se exponen los estándares y las normas seguidas en el desarrollo del software.

7.1.1.1 Javadoc

Javadoc [13] es un estándar para documentación de APIs en formato HTML. Ha sido utilizado para documentar todos los servicios de la aplicación.

7.1.1.2 Java Code Conventions

Las Java Code Conventions [14] son una serie de pautas y recomendaciones que hacen que las aplicaciones sean más fáciles de leer y comprender.

7.1.2 Lenguajes de programación

A continuación, se enumeran y describen los lenguajes de programación que se utilizarán.

7.1.2.1 Java



Ilustración 78. Logo Java

Java [15] es un lenguaje de programación orientado a objetos. Funciona de manera rápida, robusta y es multiplataforma. Java tiene como elementos principales clases que se ordenan en paquetes; cada una de estas clases contiene métodos que encapsulan porciones de código.

Se utilizará para la creación de la parte back-end de la aplicación en su versión 1.8. El jdk usado será la versión 15.0.2.

7.1.2.2 TypeScript



Ilustración 79. Logo TypeScript

TypeScript [7] es un lenguaje de programación tipado, de código abierto basado en JavaScript[4]. TypeScript añade algunas características como son los tipos estáticos y los objetos basados en clases hacen de este un lenguaje limpio, sólido y con menos errores. Es el lenguaje por defecto para desarrollar con Angular.

Se utilizará para gran parte del desarrollo del front-end de la aplicación en la versión 4.6.4.

7.1.2.3 HTML5



Ilustración 80. Logo HTML5

HTML5 [16] es la quinta versión del lenguaje HTML. Este lenguaje es utilizado para desarrollo web pudiendo crear el contenido y estructura de una página web; su elemento principal son las etiquetas utilizadas para contener información.

Se empleará para la creación de las vistas en la parte front-end de la aplicación.

7.1.2.4 CSS



Ilustración 81. Logo CSS

CSS[17] es un lenguaje de diseño gráfico para describir la apariencia de los documentos HTML. Permite crear páginas atractivas, permitiendo diseñar las interfaces de usuario de manera sencilla.

Esta separación entre el formato y el contenido de la página permite que esta esté dotada de varios estilos adecuándose a cada elemento.

Se utilizará para dar formato y color a los elementos de los archivos HTML.

7.1.3 Librerías y frameworks

En este apartado se describen las librerías y los framework que se emplearán.

7.1.3.1 Angular



Ilustración 82. Logo Angular

Angular es un framework de código abierto para desarrollar aplicaciones web de una sola página. El lenguaje de programación que utiliza por defecto es TypeScript. Este framework está basado en componentes, estos son los que harán de páginas de la web, permitiendo cambiar la pantalla en la que se encuentra el usuario.

Angular se utilizará en el desarrollo del front-end de la aplicación.

7.1.3.2 Spring Framework



Ilustración 83. Logo Spring Framework

Spring [5], [18] es un framework de código abierto empleado para el desarrollo de aplicaciones. Es uno de los frameworks más importantes para Java que permite estandarizar, agilizar y proporciona ayuda para los problemas que puedan surgir en el desarrollo.

Spring se utilizará en el desarrollo del back-end de la aplicación.

7.1.3.3 Junit



Ilustración 84. Logo Junit

JUnit [19] es un framework de código abierto empleado para la realización de pruebas unitarias de aplicaciones Java.

7.1.4 Herramientas y programas usados

A continuación, se exponen las herramientas y los programas usados en el desarrollo, pruebas y documentación de la aplicación.

7.1.4.1 IntelliJ IDEA



Ilustración 85. IntelliJ IDEA logo

Tanto el back-end de la aplicación como el front-end de la misma serán desarrollados haciendo uso del programa IntelliJ IDEA 2020.3.2 (Ultimate Edition) [20], [21]. Este es un entorno de desarrollo integrado para el desarrollo de programas informáticos.

7.1.4.2 MySQL



Ilustración 86. Logo MySQL

Descrito en el apartado 6.4.1 Descripción del SGBD Usado

7.1.4.3 MySQL Workbench



Ilustración 87. Logo MySQL Workbench

MySQL Workbench [22] es una herramienta visual de base de datos que integra desarrollo de software, administración, diseño, gestión y mantenimiento de bases de datos MySQL. La versión utilizada será la 8.0

7.1.4.4 GitHub



Ilustración 88. Logo GitHub

GitHub [23] es una herramienta que permite almacenar proyectos en la nube usando el sistema de control de versiones Git. Gracias al uso de Github se conseguirá llevar un control de los cambios realizados en el proyecto, además de un almacenamiento seguro del mismo.

7.1.4.5 Firefox



Ilustración 89. Logo Mozilla Firefox

Mozilla Firefox [24] es un navegador web libre de código abierto. Será el navegador empleado para las pruebas de integración. La versión usada es la 97.0

7.1.4.6 Selenium



Ilustración 90. Logo Selenium

Selenium [25] es un entorno de pruebas de software que se utiliza para probar aplicaciones web. Gracias a una de sus herramientas se puede grabar/reproducir casos de prueba para posteriormente reproducirlos de forma automática.

Selenium será utilizado para la realización de las pruebas de integración.

7.1.4.7 Word



Ilustración 91. Logo Word

Microsoft Word [26] es un software de procesamiento de textos. Ha sido empleado para la realización de toda la documentación.

7.1.4.8 Excel



Ilustración 92. Logo Excel

Microsoft Excel [27] es un programa de hojas de cálculo, es un software avanzado de análisis y visualización de datos. Ha sido utilizado para el Plan de gestión de riesgos.

7.1.4.9 Project



Ilustración 93. Logo Project

Microsoft Project [10] es un software de administración de proyectos, permite desarrollar planes, asignar recursos a áreas, hacer un seguimiento del progreso, administrar el presupuesto, etc.

Project ha sido utilizado para realizar todas las planificaciones de la documentación, las tareas asignadas a la misma y cada uno de los recursos que están involucrados en las mismas. Además, ha permitido realizar el presupuesto del proyecto y un seguimiento del mismo.

7.1.4.10 PowerPoint



Ilustración 94. Logo PowerPoint

Microsoft PowerPoint [28] permite realizar presentaciones en diapositivas y con texto esquematizado, además de animaciones de texto e imágenes.

7.1.4.11 Forms



Ilustración 95. Logo Forms

El último programa del paquete Microsoft Office empleado es Microsoft Forms [29] permite la creación y cumplimentación de cuestionarios. Será empleado para llevar a cabo las pruebas de usabilidad de la aplicación.

7.1.4.12 *Draw.io*



Ilustración 96. Logo Draw.io

Draw.io es un programa online que permite realizar diagramas de múltiples tipos. Ha sido empleado para realizar la mayoría de los diagramas de la documentación.

7.1.4.13 *Balsamiq*



Ilustración 97. Logo Balsamiq

Balsamiq [30] es una herramienta para la creación de prototipos de interfaz de webs y aplicaciones móviles.

Ha sido utilizado para realizar el prototipado de las pantallas de la aplicación.

7.2 CSI 2: GENERACIÓN DEL CÓDIGO DE LOS COMPONENTES Y PROCEDIMIENTOS

En este apartado se describen las clases más importantes de la aplicación, detallando sus atributos y métodos junto con su descripción.

Se ha decidido optar por mostrar únicamente los servicios `JornadaService` y `SolicitudService`, ya que son las clases que más lógica de negocio contienen. El resto de los servicios se encuentran documentados en la carpeta `./doc/con/uniovi/services`

7.2.1 JornadaService

Class `JornadaService`

```
java.lang.Object  
com.uniovi.services.JornadaService
```

```
@Service  
public class JornadaService  
extends java.lang.Object
```

Servicios encargado de la gestión de las jornadas

Author:
l'0266047

Constructor Summary

Constructors

Constructor	Description
<code>JornadaService()</code>	

Method Summary

All Methods Instance Methods Concrete Methods

Modifier and Type	Method	Description
<code>com.uniovi.entities.Jornada</code>	<code>addJornada(com.uniovi.entities.Jornada jornada)</code>	Añade una nueva jornada
<code>void</code>	<code>eliminarTodos()</code>	Elimina todas las jornadas
<code>java.util.List<com.uniovi.entities.Jornada></code>	<code>findJornadaByEmpleado(java.lang.Long id)</code>	Obtiene la jornada de un empleado dado su id
<code>java.util.List<com.uniovi.entities.Jornada></code>	<code>findJornadaByFecha(java.util.Date date)</code>	Busca las jornadas existentes en una fecha
<code>java.util.List<com.uniovi.entities.Jornada></code>	<code>findJornadaByFechaEmpleado(java.util.Date date, java.lang.Long id)</code>	Busca las jornadas de un empleado dada una fecha
<code>java.util.List<com.uniovi.entities.Tarea></code>	<code>getTareasByFechaEmpleado(java.lang.Long id, java.util.Date date)</code>	Obtiene las tareas de un empleado dado su id y una fecha
<code>void</code>	<code>marcarDialibre(java.util.Date fecha, com.uniovi.entities.Empleado empleado)</code>	Marca un día como libre
<code>void</code>	<code>reassignar(com.uniovi.entities.SolicitudIntercambio solicitud)</code>	Reasigna una solicitud de intercambio

Ilustración 98. *JavaDoc JornadaService*



7.2.2 SolicitudService

Package com.uniovi.services

Class SolicitudService

java.lang.Object
com.uniovi.services.SolicitudService

```
@Service
public class SolicitudService
extends java.lang.Object
```

Servicios encargado de la gestión de las solicitudes

Author:
UO266047

Constructor Summary

Constructors	Description
SolicitudService()	

Method Summary

All Methods	Instance Methods	Concrete Methods	Method	Description	
			Modifier and Type	Description	
			void	acceptarSolicitud(com.uniovi.entities.Solicitud solicitud)	Acepta una solicitud simple o de vacaciones
			void	addSolicitudIntercambio(com.uniovi.entities.Solicitud solicitud)	Añade una solicitud de intercambio
			void	deleteAllSolicitudes()	Elimina todas las solicitudes
			boolean	existeSolicitud(java.lang.String fecha, java.lang.Long idEmpleado)	Comprueba si existen solicitudes para la fecha y empleado señalados
			java.util.List<com.uniovi.entities.SolicitudIntercambio>	findOthersSolicitudesPendientes(com.uniovi.entities.Empleado empleado)	Busca las solicitudes de intercambio enviadas por otros compañeros
			java.util.List<com.uniovi.entities.Solicitud>	findOwnSolicitudes(java.lang.Long idEmpleado)	Busca las solicitudes de intercambio y simples de un empleado dado su id
			java.util.List<com.uniovi.entities.SolicitudVacaciones>	findSolicitudesVacaciones(java.lang.Long idEmpleado)	Busca las solicitudes de vacaciones de un empleado
			java.util.List<com.uniovi.entities.SolicitudVacaciones>	findSolicitudesVacacionesPendientes(java.lang.Long idEmpleado)	Busca las solicitudes de vacaciones de un empleado con estado pendiente
			java.util.List<com.uniovi.entities.Solicitud>	getAllSolicitudesPendientes()	
			void	rechazarSolicitud(java.lang.Long idSolicitud)	Rechaza una solicitud dado su id
			void	setSolicitud(com.uniovi.entities.SolicitudSimple solicitud)	Añade una solicitud simple
			void	solicitarVacaciones(com.uniovi.entities.SolicitudVacaciones solicitud)	Añade una solicitud de vacaciones

Methods inherited from class java.lang.Object
clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Ilustración 99. JavaDoc SolicitudService

7.3 CSI 3: EJECUCIÓN DE LAS PRUEBAS UNITARIAS

En este apartado se describe el resultado de las pruebas unitarias diseñadas en el apartado 6.5.1 Pruebas Unitarias.

7.3.1 EmpleadoService

EmpleadoService	
Prueba	Resultado
Añadir un nuevo usuario. <ul style="list-style-type: none">- Username: admin1- Name: Administrador- Surname: Prueba- Email: admin1@gmail.com- DNI: 11111111A- Password: TestPassword1- Rol: Admin- nDiasLibres: 100	Esperado: Se añade el usuario a la base de datos Obtenido: CORRECTO
Añadir un usuario repetido <ul style="list-style-type: none">- Username: admin1- Name: Administrador- Surname: Prueba- Email: admin1@gmail.com- DNI: 11111111A- Password: TestPassword1- Rol: Admin- nDiasLibres: 100	Esperado: El usuario no se almacena en la base de datos Obtenido: CORRECTO
Añadir un usuario con datos inválidos <ul style="list-style-type: none">- Username: admin2- Name: Administrador- Surname: Prueba- Email: admin2- DNI: 11A- Password: admin- Rol: Admin- nDiasLibres: 100	Esperado: El usuario no se almacena en la base de datos Obtenido: INCORRECTO Solución: Añadir validaciones a los campos
Buscar un empleado por nombre de usuario y contraseña	Esperado: Se obtiene un empleado que coincide. Obtenido: INCORRECTO

- Username: admin1 - Password: TestPassword1	Solución: Añadir validación de pareja username-password única
Listar todos los empleados de la base de datos	Esperado: Habiendo añadido dos empleados, se obtiene una lista con 1 empleado Obtenido: CORRECTO
Eliminar un empleado existente	Esperado: El empleado se elimina de la base de datos. Obtenido: CORRECTO
Eliminar un empleado no existente	Esperado: Se obtiene una excepción y la longitud de la lista de empleados no varía Obtenido: CORRECTO

Tabla 98. Ejecución de pruebas unitarias – EmpleadoService

7.3.2 EstacionService

EstacionService	
Prueba	Resultado
Añadir una estación - Id: "1" - Nombre: "Ferreros" - Latitud: 1.1 - Longitud: 1.1	Esperado: Se añade una nueva estación y se obtiene una lista con una estación más que inicialmente Obtenido: CORRECTO
Añadir una estación repetida - Id: "1" - Nombre: "Ferreros" - Latitud: 1.1 - Longitud: 1.1	Esperado: No se añade la estación Obtenido: CORRECTO
Añadir una estación con parámetros inválidos - Id: "" - Nombre: "" - Latitud: 1.1 - Longitud: 1.1	Esperado: Lanza una excepción que se recoge en el test. Obtenido: CORRECTO
Buscar una estación por id - Id: "1"	Esperado: Se comprueba que el resultado es distinto de null, por lo tanto la estación existe. Obtenido: CORRECTO
Eliminar una estación sin horarios asociados	Esperado: Se elimina la estación y se obtiene una lista de longitud inferior en 1 que inicialmente Obtenido: CORRECTO

Tabla 99. Ejecución de pruebas unitarias – EstacionService

7.3.3 IncidenciaService

IncidenciaService	
Prueba	Resultado esperado
Añadir una incidencia válida <ul style="list-style-type: none"> - Descripción: "Incidencia prueba" - Tren: tren almacenado en la base de datos 	Esperado: Se añade la incidencia, la lista de incidencias pasa a tener longitud 1 Obtenido: CORRECTO
Añadir una incidencia no válida <ul style="list-style-type: none"> - Descripción: "" - Tren: null 	Esperado: Lanza una excepción que se recoge en el test Obtenido: INCORRECTO Solución: Añadir comprobación not null para el tren
Buscar todas las incidencias	Esperado: Tras añadir dos incidencias a la base de datos, la lista devuelta tiene una longitud igual a 2 Obtenido: CORRECTO
Buscar las incidencias pendientes	Esperado: Tras cambiar el estado de una de las dos incidencias de la base de datos, la lista de incidencias pendientes es de dimensión 1 Obtenido: CORRECTO
Eliminar una incidencia existente	Esperado: Se elimina la incidencia y la lista de incidencias decrementa su longitud en 1 Obtenido: CORRECTO
Eliminar una incidencia no existente	Esperado: Se lanza una excepción que se recoge en el test. Obtenido: CORRECTO

Tabla 100. Ejecución de pruebas unitarias – IncidenciaService

7.3.4 JornadaService

JornadaService	
Prueba	Resultado esperado
Añadir una jornada válida <ul style="list-style-type: none"> - Fecha: 2022-10-17 - Empleado: Empleado existente en la BD - isDiaLibre: false 	Se almacena la jornada en la base de datos, la longitud de la lista de las jornadas existentes aumenta en 1. Obtenido: CORRECTO
Añadir una jornada inválida <ul style="list-style-type: none"> - Fecha: null 	Lanza una excepción que es recogida en el test



<ul style="list-style-type: none"> - Empleado: null - isDiaLibre: false 	Obtenido: CORRECTO
<p>Buscar una jornada por fecha. AFTER añadir dos jornadas con fecha "2022-10-17"</p> <ul style="list-style-type: none"> - Fecha: 2022-10-17 	<p>Devuelve una lista de longitud 2 con las dos jornadas correspondientes añadidas anteriormente.</p> <p style="background-color: #d9ead3;">Obtenido: CORRECTO</p>
<p>Buscar una jornada por fecha inexistente en la BD</p> <ul style="list-style-type: none"> - Fecha: 2020-10-17 	<p>Se devuelve una lista vacía.</p> <p style="background-color: #d9ead3;">Obtenido: CORRECTO</p>
<p>Buscar una jornada por empleado. AFTER añadir dos jornadas con el mismo empleado asignado</p> <ul style="list-style-type: none"> - Empleado: Empleado en la BD 	<p>Devuelve una lista de longitud 2 con las dos jornadas correspondientes añadidas anteriormente.</p> <p style="background-color: #d9ead3;">Obtenido: CORRECTO</p>
<p>Buscar una jornada por un empleado que no tiene ninguna jornada asignada</p> <ul style="list-style-type: none"> - Empleado: Empleado en la BD 	<p>Se devuelve una lista vacía</p> <p style="background-color: #d9ead3;">Obtenido: CORRECTO</p>
<p>Buscar una jornada por fecha y empleado AFTER añadir una jornada con esa fecha y ese empleado en la BD</p> <ul style="list-style-type: none"> - Fecha: 2022-10-17 - Empleado: Empleado en la BD 	<p>Se devuelve una lista de un elemento cuya fecha y empleado coinciden con los buscados</p> <p style="background-color: #d9ead3;">Obtenido: CORRECTO</p>
<p>Marcar como día libre una fecha en la cual existe jornada. AFTER:</p> <ol style="list-style-type: none"> 1. Añadir una jornada con esa fecha para ese empleado 2. Añadir una tarea a la jornada <ul style="list-style-type: none"> - Fecha: 2022-10-17 - Empleado: Empleado en la BD 	<p>Se actualiza el valor del campo isDiaLibre convirtiendo el día en día libre y se comprueba que la lista de tareas para la jornada tiene una longitud igual a 0</p> <p style="background-color: #d9ead3;">Obtenido: CORRECTO</p>
<p>Marcar como día libre una fecha sin jornada para un empleado existente</p>	<p>Se crea una nueva jornada para la fecha indicada y se asigna como día libre</p> <p style="background-color: #d9ead3;">Obtenido: CORRECTO</p>
<p>Buscar tareas de un empleado no existente en la BD por fecha</p> <ul style="list-style-type: none"> - IdEmpleado: 0000 - Fecha: 2022-10-17 	<p>No se obtiene ninguna Tarea</p> <p style="background-color: #d9ead3;">Obtenido: CORRECTO</p>
<p>Buscar tareas de un empleado por fecha.</p>	<p>Se devuelve una lista con una tarea.</p>



<p>AFTER añadir el empleado en la BD y añadir una jornada con una tarea</p> <ul style="list-style-type: none"> - IdEmpleado: Id de un empleado en la BD - Fecha: 2022-10-17 	<p>Obtenido: CORRECTO</p>
<p>Buscar tareas de un empleado sin tareas por fecha.</p> <p>AFTER añadir el empleado en la BD</p> <ul style="list-style-type: none"> - IdEmpleado: Id de un empleado en la BD - Fecha: 2022-10-17 	<p>Se devuelve una lista vacía</p> <p>Obtenido: CORRECTO</p>
<p>Buscar tareas de un empleado por fecha inválida.</p> <ul style="list-style-type: none"> - IdEmpleado: null - Fecha: null 	<p>Se devuelve una lista vacía</p> <p>Obtenido: CORRECTO</p>
<p>Buscar tareas de un empleado sin jornada asignada por fecha</p> <p>AFTER añadir el empleado en la base de datos</p> <ul style="list-style-type: none"> - IdEmpleado: Id del empleado en la BD - Fecha: 2022-10-17 	<p>Se obtiene una lista de longitud 0</p> <p>Obtenido: CORRECTO</p>
<p>Buscar tareas de un empleado sin tareas dada una fecha, existiendo tareas en esa fecha para otro empleado distinto.</p> <p>AFTER añadir una nueva jornada en esa fecha a un empleado existente</p> <ul style="list-style-type: none"> - IdEmpleado: Id del empleado en la BD - Fecha: 2022-10-17 	<p>Se obtiene una lista de longitud 0</p> <p>Obtenido: CORRECTO</p>
<p>Buscar tareas de un empleado para una fecha en la que no tiene jornada, existiendo jornada para otra fecha.</p> <p>AFTER añadir una jornada con fecha distinta a la pasada por parámetro para ese mismo empleado</p> <ul style="list-style-type: none"> - IdEmpleado: Id del empleado en la BD - Fecha: 2022-10-17 	<p>Se obtiene una lista de longitud 0</p> <p>Obtenido: CORRECTO</p>
<p>Reasignar una jornada para un empleado que no tiene jornada asignada en el día indicado</p> <ul style="list-style-type: none"> - SolicitudIntercambio: Solicitud en la BD 	<p>No se reasigna la jornada y el estado de la solicitud permanece "PENDIENTE"</p> <p>Obtenido: CORRECTO</p>
<p>Reasignar una jornada, siendo el nuevo empleado null.</p> <ul style="list-style-type: none"> - SolicitudIntercambio: Solicitud en la BD 	<p>No se reasigna la jornada y el estado de la solicitud permanece "PENDIENTE"</p> <p>Obtenido: CORRECTO</p>

<p>Reasignar una jornada válida</p> <ul style="list-style-type: none"> - SolicitudIntercambio: Solicitud en la BD 	<p>Se intercambian las jornadas de los empleados implicados. Se marca la solicitud como “REASIGNADA”</p> <p>Obtenido: CORRECTO</p>
---	---

Tabla 101. Ejecución de pruebas unitarias- JornadaService

7.3.5 RutaService

RutaService	
Prueba	Resultado esperado
<p>Añadir una ruta válida</p> <ul style="list-style-type: none"> - Id: “1” - Ruta_corto: “p” - Ruta_largo: “prueba” 	<p>Se almacena la ruta en la BD</p> <p>Obtenido: CORRECTO</p>
<p>Añadir una ruta inválida</p> <ul style="list-style-type: none"> - Id: “” - Ruta_corto: “” - Ruta_largo: “” 	<p>Se lanza una excepción que se recoge en el test</p> <p>Obtenido: INCORRECTO</p> <p>Solución: Se añaden las validaciones de los argumentos al constructor de la ruta</p>
<p>Añadir una ruta repetida</p> <ul style="list-style-type: none"> - Id: “1” - Ruta_corto: “p” - Ruta_largo: “prueba” 	<p>No se añade la ruta a la BD</p> <p>Obtenido: CORRECTO</p>
<p>Buscar una ruta por id válido</p> <ul style="list-style-type: none"> - Id: “1” 	<p>Se encuentra la ruta buscada</p> <p>Obtenido: CORRECTO</p>
<p>Buscar una ruta por id no existente</p> <ul style="list-style-type: none"> - Id: “0000” 	<p>No se encuentra ninguna ruta en la BD</p> <p>Obtenido: CORRECTO</p>
<p>Buscar una ruta entre dos estaciones no existentes</p> <ul style="list-style-type: none"> - OrigenId: “0000” - Destinold: “0001” 	<p>No se encuentra ninguna ruta en la BD</p> <p>Obtenido: CORRECTO</p>
<p>Buscar una ruta entre dos estaciones con ruta AFTER añadir dos rutas con ids 1 y 2 respectivamente y una ruta entre ellas</p> <ul style="list-style-type: none"> - OrigenId: “1” - Destinold: “2” 	<p>Se encuentra una ruta que cumple la condición de incluir ambas estaciones</p> <p>Obtenido: CORRECTO</p>
<p>Buscar una ruta entre dos estaciones sin ruta AFTER añadir dos estaciones con ids 1 y 2</p> <ul style="list-style-type: none"> - OrigenId: “1” - Destinold: “2” 	<p>No se encuentra una ruta entre las dos estaciones existentes en la BD</p> <p>Obtenido: CORRECTO</p>

Tabla 102. Ejecución de pruebas unitarias - RutaService

7.3.6 SolicitudService

SolicitudService	
Prueba	Resultado esperado
Añadir una solicitud null - solicitud: null	No se añade la solicitud a la BD Obtenido: CORRECTO
Añadir una solicitud válida a la BD - fecha: "2022-10-17" - motivo: "Licencia" - empleado: Empleado en la BD	Se añade correctamente a la BD Obtenido: CORRECTO
Rechazar una solicitud AFTER añadir una solicitud simple a la BD - id: Id de una solicitud en la BD	El estado de la solicitud pasa a ser "RECHAZADA" Obtenido: CORRECTO
Buscar todas las solicitudes pendientes sin que haya solicitudes	No se obtiene ninguna solicitud Obtenido: CORRECTO
Buscar todas las solicitudes pendientes AFTER añadir una solicitud simple y una solicitud de vacaciones al sistema.	Se encuentran dos solicitudes con estado "PENDIENTE" Obtenido: CORRECTO
Buscar todas las solicitudes pendientes. AFTER añadir una solicitud simple y una solicitud de vacaciones al sistema y rechazar la solicitud simple	Se encuentra una sola solicitud con estado "PENDIENTE" Obtenido: CORRECTO
Buscar todas las solicitudes pendientes con solamente una solicitud de intercambio en el sistema AFTER añadir una solicitud de intercambio a la BD	No se obtiene ninguna solicitud con estado "PENDIENTE" Obtenido: CORRECTO
Añadir una solicitud de intercambio pasando como parámetro una solicitud simple AFTER añadir un empleado a la BD - solicitud: solicitudSimple - fecha: "2022-10-17" - motivo: "Licencia" - empleado: Empleado en la BD	No se añade la solicitud a la BD Obtenido: CORRECTO
Añadir una solicitud de intercambio inválida - solicitud: null	No se añade la solicitud a la BD Obtenido: CORRECTO



<p>Añadir una solicitud de intercambio válida AFTER añadir un empleado a la BD</p> <ul style="list-style-type: none"> - fecha: "2022-10-17" - motivo: "Licencia" - empleado: Empleado en la BD - fechaDescanso: "2022-11-17" - nuevoEmpleado: null 	<p>Se añade la solicitud de intercambio a la BD</p> <p>Obtenido: CORRECTO</p>
<p>Añadir una solicitud de vacaciones inválida</p> <ul style="list-style-type: none"> - solicitud: null 	<p>La solicitud no se añade a la BD</p> <p>Obtenido: CORRECTO</p>
<p>Añadir una solicitud de vacaciones válida AFTER añadir un empleado a la BD</p> <ul style="list-style-type: none"> - fecha: "2022-10-17" - motivo: "Licencia" - empleado: Empleado en la BD 	<p>Se añade la solicitud de vacaciones a la BD</p> <p>Obtenido: CORRECTO</p>
<p>Buscar una solicitud de vacaciones por id de empleado con id inválido</p> <ul style="list-style-type: none"> - idEmpleado: null 	<p>No se obtiene ninguna solicitud</p> <p>Obtenido: CORRECTO</p>
<p>Buscar una solicitud de vacaciones con id de empleado no existente</p> <ul style="list-style-type: none"> - idEmpleado: "0000" 	<p>No se obtiene ninguna solicitud</p> <p>Obtenido: CORRECTO</p>
<p>Buscar solicitudes de vacaciones por id de empleado existente AFTER añadir un empleado y dos solicitudes de vacaciones a la BD.</p> <ul style="list-style-type: none"> - idEmpleado: Id de un empleado en la BD 	<p>Se obtiene una lista con dos solicitudes de vacaciones</p> <p>Obtenido: CORRECTO</p>
<p>Buscar las solicitudes simples y de intercambio realizadas por el mismo empleado que las busca AFTER añadir dos empleados a la BD. El primero de ellos realiza dos solicitudes y el segundo una</p> <ul style="list-style-type: none"> - idEmpleado: Empleado en la BD que ha realizado dos solicitudes 	<p>Se obtiene una lista con dos solicitudes, una de tipo simple y otra de tipo intercambio</p> <p>Obtenido: CORRECTO</p>
<p>Buscar las solicitudes propias con id de empleado inválido</p> <ul style="list-style-type: none"> - idEmpleado: null 	<p>Se obtiene una lista vacía</p> <p>Obtenido: CORRECTO</p>
<p>Buscar las solicitudes propias para un empleado que no está en la BD</p> <ul style="list-style-type: none"> - idEmpleado: "0000" 	<p>Se obtiene una lista vacía</p> <p>Obtenido: CORRECTO</p>
	<p>Se obtiene una lista vacía</p>



<p>Buscar las solicitudes por fecha y por empleado con datos inválidos</p> <ul style="list-style-type: none"> - fecha: "" - idEmpleado: null 	<p>Obtenido: CORRECTO</p>
<p>Buscar las solicitudes simples y de intercambio por fecha y empleado</p> <p>AFTER añadir un empleado y dos solicitudes de tipos simples e intercambio a la BD</p> <ul style="list-style-type: none"> - fecha: "2022-10-17" - idEmpleado: Id de un empleado en la BD 	<p>Se obtiene una lista con dos solicitudes</p> <p>Obtenido: CORRECTO</p>
<p>Buscar las solicitudes de intercambio de un empleado inválido con estado "PENDIENTE"</p> <ul style="list-style-type: none"> - empleado: null 	<p>Se obtiene una lista vacía</p> <p>Obtenido: INCORRECTO</p> <p>Solución: Añadir comprobación de que el empleado no sea null</p>
<p>Buscar las solicitudes de intercambio pendientes de otros empleados, con fecha no coincidente con una jornada existente.</p> <p>AFTER añadir dos empleados y una solicitud del primero en la BD</p> <ul style="list-style-type: none"> - empleado: Segundo empleado almacenado en la BD 	<p>Se obtiene una lista vacía debido a la no coincidencia con una jornada del empleado</p> <p>Obtenido: CORRECTO</p>
<p>Buscar las solicitudes de intercambio pendientes de otros empleados, con fecha coincidente con una jornada.</p> <p>AFTER añadir dos empleados, una solicitud del segundo en la BD y una jornada para el primero con la misma fecha que la solicitud</p> <ul style="list-style-type: none"> - empleado: Primer empleado almacenado en la BD 	<p>Se obtiene una lista con una solicitud, la creada por el segundo empleado.</p> <p>Obtenido: CORRECTO</p>
<p>Buscar las solicitudes de intercambio pendientes de otros empleados, con fecha coincidente con una jornada que es día libre.</p> <p>AFTER añadir dos empleados, una solicitud del segundo en la BD y una jornada para el primero con la misma fecha que la solicitud.</p> <ul style="list-style-type: none"> - empleado: Primer empleado almacenado en la BD 	<p>Se obtiene una lista vacía.</p> <p>Obtenido: CORRECTO</p>

<p>Buscar las solicitudes de intercambio pendientes de otros empleados, con fecha coincidente con una jornada.</p> <p>AFTER añadir dos empleados, una solicitud del segundo en la BD y dos jornadas para el primero coincidiendo con la fecha de descanso y la fecha a cubrir</p> <ul style="list-style-type: none"> - empleado: Primer empleado almacenado en la BD 	<p>Se obtiene una lista vacía debido a la existencia de jornada para el día a cubrir y el día libre.</p> <p>Obtenido: CORRECTO</p>
<p>Aceptar una solicitud de tipo no válido (intercambio)</p> <p>AFTER añadir un empleado y una solicitud de intercambio a la BD</p> <ul style="list-style-type: none"> - solicitud: SolicitudIntercambio en la BD 	<p>La solicitud no cambia de estado y sigue en "PENDIENTE" y la jornada del empleado no se ve afectada.</p> <p>Obtenido: CORRECTO</p>
<p>Aceptar una solicitud de tipo simple.</p> <p>AFTER añadir un empleado, una solicitud simple a la BD y una jornada con tareas para la fecha de la solicitud</p> <ul style="list-style-type: none"> - solicitud: SolicitudSimple en la BD 	<p>La solicitud pasa a estar en estado "ACEPTADA". La jornada se marca como libre, todas las tareas asociadas se eliminan</p> <p>Obtenido: INCORRECTO</p> <p>Solución: Modificar la forma en la que se cambia el estado de la solicitud</p>
<p>Aceptar una solicitud de tipo vacaciones</p> <p>AFTER añadir un empleado y una solicitud de vacaciones a la BD</p> <ul style="list-style-type: none"> - solicitud: SolicitudVacaciones en la BD 	<p>La solicitud pasa a estado "ACEPTADA", las jornadas entre las fechas de inicio y fin se marcan como libres y las tareas asociadas a esos días se eliminan.</p> <p>Obtenido: CORRECTO</p>

Tabla 103. Ejecución de pruebas unitarias – SolicitudService

7.3.7 TareaService

TareaService	
Prueba	Resultado esperado
<p>Buscar tarea por id existente</p> <p>AFTER añadir un empleado y una jornada con tareas a la BD</p> <ul style="list-style-type: none"> - idTarea: Id una tarea en la BD 	<p>Se obtiene una tarea con ese id</p> <p>Obtenido: CORRECTO</p>
<p>Buscar tarea por id inexistente</p> <ul style="list-style-type: none"> - idTarea: "0000" 	<p>Se obtiene null</p> <p>Obtenido: CORRECTO</p>
<p>Eliminar una tarea</p>	<p>Se comprueba que al buscar la tarea con ese id, no se encuentra ninguna.</p>

AFTER añadir un empleado y una jornada con tareas a la BD - tarea: Tarea almacenada en la BD	Obtenido: CORRECTO
Buscar todas las tareas siendo estas inexistentes	Se obtiene una lista vacía Obtenido: CORRECTO
Buscar todas las tareas existentes AFTER añadir un empleado y una jornada con dos tareas a la BD	Se obtiene una lista con dos tareas Obtenido: CORRECTO

Tabla 104. Ejecución de pruebas unitarias – TareaService

7.3.8 TareaEstacionService

TareaEstacionService	
Prueba	Resultado esperado
Añadir una nueva tarea_ estacion AFTER añadir una estación y una tarea a la BD - situación: INICIO - estación: Estación en la BD - tarea: Tarea en la BD	Se añade una nueva tarea_ estacion a la BD Obtenido: CORRECTO
Añadir una nueva tarea_ estacion con datos inválidos - situación: null - estación: null - tarea: null	Devuelve una excepción de parámetros ilegales que es recogida en el test Obtenido: CORRECTO

Tabla 105. Ejecución de pruebas unitarias - TareaEstacionService

7.3.9 TrenService

TrenService	
Prueba	Resultado esperado
Añadir un tren - tren: nuevo tren	Se añade el tren a la BD Obtenido: CORRECTO
Añadir un tren null - tren: null	El tren no se añade a la BD Obtenido: CORRECTO
Buscar un tren por id AFTER añadir un tren a la BD - trenId: Id del tren en la BD	Se obtiene un tren con id coincidente al parámetro Obtenido: CORRECTO
Buscar un tren por id inexistente - trenId: "0000"	No se obtienen ningún tren Obtenido: CORRECTO

Tabla 106. Ejecución de pruebas unitarias - TrenService



Adicionalmente a los problemas descritos en cada una de las pruebas de los servicios con resultado “INCORRECTO” se han tenido otras dificultades:

- **Problema 1:** Al inicio de las pruebas se trabajaba con una versión posterior de Junit 4, lo que provocaba que alguna de las anotaciones no funcionase correctamente. Tras cambiar la versión a la 4.12 y utilizar todas las notaciones del paquete “org.junit” en lugar de las de “jupiter” el problema se solucionó.
- **Problema 2:** Por otra parte, surgió la necesidad de crear los métodos con notaciones “@After” y “@Before” para limpiar la base de datos de los nuevos elementos que se habían añadido y modificado en la ejecución de las pruebas.

7.4 CSI 4: EJECUCIÓN DE LAS PRUEBAS DE INTEGRACIÓN

En este apartado se describe el resultado de las pruebas de integración diseñadas en el apartado 6.5.2 Pruebas de Integración y del Sistema. Para la ejecución de las pruebas de integración se debe ejecutar primero el archivo *scriptCarga.sql* adjuntado como anexo de la entrega para cargar los datos necesarios en la base de datos.

7.4.1 Usuario no identificado

Pruebas usuario no identificado	
Prueba	Resultado esperado
Acceder a la página de inicio	Se accede a la página principal y contiene lo esperado Obtenido: CORRECTO
Acceder a la consulta de horarios	Se accede a la página de consulta y se muestra el mapa y los selectores de estaciones para la consulta Obtenido: CORRECTO
Consultar horarios entre estaciones con ruta <ul style="list-style-type: none">- origen: PUENTE DE LOS FIERROS- destino: POLA DE LENA	Se redirige a una nueva página donde se muestran los horarios correspondientes Obtenido: CORRECTO
Consultar horarios entre dos estaciones sin ruta <ul style="list-style-type: none">- origen: GIJÓN- destino: AVILÉS	Se muestra un mensaje de incompatibilidad de estaciones en ruta Obtenido: CORRECTO
Iniciar sesión <ul style="list-style-type: none">- Nombre de usuario: admin1- Contraseña: Password10	Se autentica al usuario introducido y se redirige a la página principal Obtenido: CORRECTO
Iniciar sesión: campos vacíos	No se autentica el usuario y se muestra un mensaje de error Obtenido: CORRECTO
Iniciar sesión: campos erróneos <ul style="list-style-type: none">- Nombre de usuario: a- Contraseña: Password10	No se autentica el usuario y se muestra un mensaje de error Obtenido: CORRECTO

Tabla 107. Ejecución de pruebas de integración - Usuario no identificado

7.4.2 Usuario trabajador identificado

Pruebas usuario administrador	
Prueba	Resultado esperado
Consultar una jornada - Fecha: 30 noviembre 2022	Se muestra una lista de tareas para la jornada y sus detalles en una ventana modal. Obtenido: INCORRECTO Solución: Modificar el html de la ventana modal para englobarlo todo en un div con etiqueta class
Añadir una incidencia - Fecha tarea: 30 noviembre 2022 - Descripción: Prueba de integración add incidencia	Se añade una incidencia a la lista de incidencias existentes Obtenido: CORRECTO
Solicitar un día libre - Fecha: 29 noviembre 2022	La jornada aparece marcada como día libre Obtenido: CORRECTO
Solicitar vacaciones - Periodo: 3	Se elimina la posibilidad de seleccionar vacaciones de nuevo y se muestran los periodos solicitados Obtenido: CORRECTO
Solicitar vacaciones repetidas	No se muestra la lista de periodos de vacaciones a seleccionar Obtenido: CORRECTO
Solicitar intercambio - Día a solicitar: 30/11/2022 - Día a cubrir: 25/12/2022	Se muestra un mensaje indicando que la solicitud ha sido añadida con éxito Obtenido: CORRECTO
Solicitar intercambio sin jornada coincidente - Día a solicitar: 20/12/2022 - Día a cubrir: 25/12/2022	Se muestra un mensaje explicando el problema que impide realizar la solicitud Obtenido: CORRECTO
Consultar mis solicitudes	Se muestra una tabla con las solicitudes propias existentes Obtenido: CORRECTO
Consultar vacaciones	Se muestran los periodos de vacaciones solicitados Obtenido: CORRECTO
Ver perfil	Se redirige al usuario a una página donde se muestran los detalles del empleado Obtenido: CORRECTO
Modificar contraseña - Nueva contraseña: Password5	Se muestra un mensaje indicando que la contraseña ha sido modificada con éxito

- Repetir contraseña: Password5	Obtenido: INCORRECTO Solución: Se intentaba acceder al contenido del input como si se tratase de un div, al usar la propiedad "getAttribute" el problema se soluciona
Modificar contraseña: comprobación incorrecta	Se muestra un mensaje indicando que las contraseñas no coinciden
- Nueva contraseña: Password1	Obtenido: CORRECTO
- Repetir contraseña: Password5	
Aceptar solicitud intercambio	Se elimina la solicitud de la lista de solicitudes
Solicitud: 1	Obtenido: CORRECTO

Tabla 108. Ejecución de pruebas de integración - Usuario trabajador identificado

7.4.3 Usuario administrador identificado

Pruebas usuario administrador	
Prueba	Resultado esperado
Buscar un empleado en el filtro	La tabla de empleados queda filtrada por la cadena introducida
- Nombre de usuario: empleado1	Obtenido: CORRECTO
Añadir un empleado	Se añade el nuevo empleado a la lista de empleados
- Nombre de usuario: empleado2	Obtenido: CORRECTO
- Nombre: Empleado2	
- Apellido: Prueba	
- DNI: 33333333C	
- Email: empleado2@gmail.com	
- Rol: MAQUINISTA	
Eliminar un empleado	Se elimina el empleado de la lista de empleados
- Empleado: empleado2	Obtenido: CORRECTO
Consultar los detalles de un empleado	Se muestra la ventana modal con los detalles del empleado seleccionado
- Empleado: empleado1	Obtenido: CORRECTO
Añadir una tarea	Se añade una nueva tarea a la lista de tareas
- Fecha: 30/11/2022	Obtenido: INCORRECTO
- Empleado: empleado1	Solución: El footer de la página tapaba el botón de añadir la tarea, se modificaron alguna de las propiedades del css para solucionarlo
- Andén: 1	
- Descripción: Prueba integración 30nov	
- Origen: PUENTE DE LOS FIERROS	



<ul style="list-style-type: none">- Destino: POLA DE LENA- Hora inicio: 10:00- Hora fin: 12:00- Identificador del tren: 1	
Añadir una tarea sin seleccionar un empleado	Se muestra un mensaje emergente indicando que se debe seleccionar antes un empleado Obtenido: CORRECTO
Consultar los detalles de una tarea <ul style="list-style-type: none">- Fecha: 30/11/2022- Empleado: empleado1	Se muestra la venta modal con los detalles de la tarea Obtenido: CORRECTO
Ver las solicitudes pendientes	Se accede a la página de las solicitudes con una tabla que contiene todas las solicitudes pendientes Obtenido: CORRECTO
Aceptar solicitud simple <ul style="list-style-type: none">- Fecha: 29/11/2022	Se elimina la solicitud de la lista de solicitudes Obtenido: CORRECTO
Rechazar solicitud AFTER volver a solicitar día libre fecha 29/11/2022 <ul style="list-style-type: none">- Fecha: 29/11/2022	Se elimina la solicitud de la lista de solicitudes Obtenido: CORRECTO
Aceptar solicitud vacaciones <ul style="list-style-type: none">- Periodo: 2022-12-01 – 2022-12-20	Se elimina la solicitud de la lista de solicitudes Obtenido: CORRECTO

Tabla 109. Ejecución de pruebas de integración - Usuario administrador identificado

7.5 CSI 5: EJECUCIÓN DE LAS PRUEBAS DEL SISTEMA

7.5.1 Pruebas de Usabilidad

En este apartado se describe el resultado de las pruebas de usabilidad diseñadas en el apartado 6.5.3 Pruebas de Usabilidad. Los resultados han sido extraídos de los formularios creados en ese apartado y cumplimentados por los 5 usuarios.

7.5.1.1 Preguntas de carácter personal

1. ¿Usa un ordenador frecuentemente? (0 punto)

[Más detalles](#)

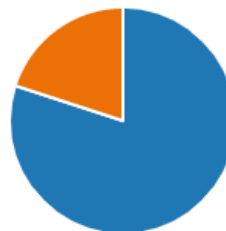
● Todos los días	2
● Varias veces a la semana	2
● Ocasionalmente	1
● Nunca o casi nunca	0



2. ¿Qué tipo de actividades realiza con el ordenador? (0 punto)

[Más detalles](#)

● Es parte de mi trabajo o profesión	4
● Lo uso básicamente para ocio	1
● Solo empleo aplicaciones estilo ...	0
● Únicamente leo el correo y nave...	0



3. ¿Ha usado usted algún software como el de esta prueba? (0 punto)

[Más detalles](#)

● Sí, he empleado software similar	1
● No, aunque sí empleo otros pro...	2
● No, nunca	2



4. ¿Qué busca Vd. principalmente en un programa? (0 punto)

[Más detalles](#)

● Que sea fácil de usar	0
● Que sea intuitivo	2
● Que sea rápido	0
● Que tenga todas las funciones n...	3



5. ¿Qué nivel de conocimientos sobre informática tiene? (0 punto)

[Más detalles](#)

● Nulos	0
● Básicos	2
● Medios	1
● Avanzados	2



6. ¿Qué edad tiene? (0 punto)

[Más detalles](#)

● Entre 18 y 35	3
● Entre 35 y 60	2
● Mayor de 60	0
● Otras	0

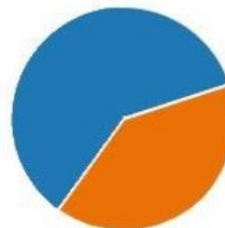


Ilustración 100. Ejecución pruebas de usabilidad - preguntas de carácter personal

7.5.1.2 Actividades guiadas



7. Consultar el mapa con las estaciones (0 punto)

[Más detalles](#)

● Logrado	4
● No logrado	1



10. Cerrar sesión (0 punto)

[Más detalles](#)

● Logrado	5
● No logrado	0



8. Consultar los horarios entre dos estaciones (0 punto)

[Más detalles](#)

● Logrado	5
● No logrado	0



11. Añadir empleado (0 punto)

[Más detalles](#)

● Logrado	5
● No logrado	0



9. Iniciar sesión (0 punto)

[Más detalles](#)

● Logrado	5
● No logrado	0



12. Eliminar empleado (0 punto)

[Más detalles](#)

● Logrado	5
● No logrado	0



13. Filtrar empleados (0 punto)

[Más detalles](#)

● Logrado	5
● No logrado	0



16. Consultar las tareas para una fecha y un empleado concretos (0 punto)

[Más detalles](#)

● Logrado	5
● No logrado	0



14. Consultar detalles de un empleado (0 punto)

[Más detalles](#)

● Logrado	5
● No logrado	0



17. Consultar los detalles de una tarea (0 punto)

[Más detalles](#)

● Logrado	5
● No logrado	0



15. Añadir una nueva tarea (0 punto)

[Más detalles](#)

● Logrado	5
● No logrado	0



18. Ver solicitudes (0 punto)

[Más detalles](#)

● Logrado	5
● No logrado	0

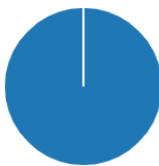




19. Aceptar solicitudes (0 punto)

[Más detalles](#)

● Logrado 5
● No logrado 0



22. Modificar contraseña (0 punto)

[Más detalles](#)

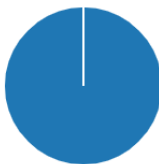
● Logrado 5
● No logrado 0



20. Rechazar solicitudes (0 punto)

[Más detalles](#)

● Logrado 5
● No logrado 0



23. Consultar jornada (0 punto)

[Más detalles](#)

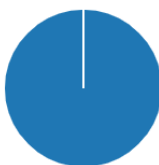
● Logrado 5
● No logrado 0



21. Ver perfil (0 punto)

[Más detalles](#)

● Logrado 5
● No logrado 0



24. Solicitar día libre (0 punto)

[Más detalles](#)

● Logrado 5
● No logrado 0



25. Solicitar vacaciones (0 punto)

[Más detalles](#)

● Logrado 5
● No logrado 0



28. Consultar solicitudes (0 punto)

[Más detalles](#)

● Logrado 5
● No logrado 0



26. Solicitar intercambio (0 punto)

[Más detalles](#)

● Logrado 5
● No logrado 0



27. Aceptar intercambio (0 punto)

[Más detalles](#)

● Logrado 5
● No logrado 0



29. Dificultades encontradas o comentarios

5 Respuestas

ID ↑	Nombre	Respuestas
1	anonymous	Añadiría un mensaje informativo al aceptar y rechazar las solicitudes simples en modo administrador.
2	anonymous	Podría ser conveniente mejorar la visión de la consulta del mapa de estaciones
3	anonymous	En la pregunta 12, encontré el botón de eliminar, pero el funcionamiento no era el esperado.
4	anonymous	Ninguna dificultad, la interfaz es muy intuitiva. Posible cambio: La fecha al solicitar un intercambio aparece en formato inglés, puede que sea mejor que aparezca en formato español.
5	anonymous	Me gustaría que resaltase la opción de poder seleccionar los empleados en la tabla a la hora de gestionar las tareas

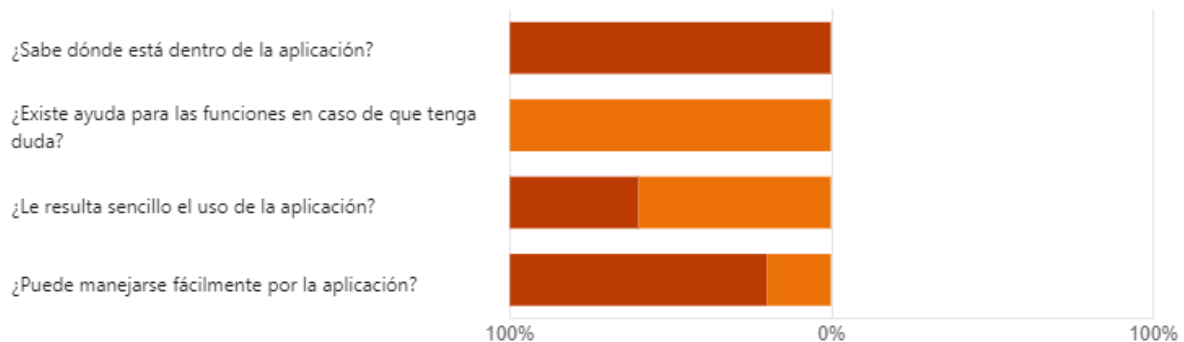
Ilustración 101. Ejecución de pruebas de usabilidad - actividades guiadas

7.5.1.3 Preguntas cortas sobre la aplicación y observaciones

30. Facilidad de uso (0 punto)

[Más detalles](#)

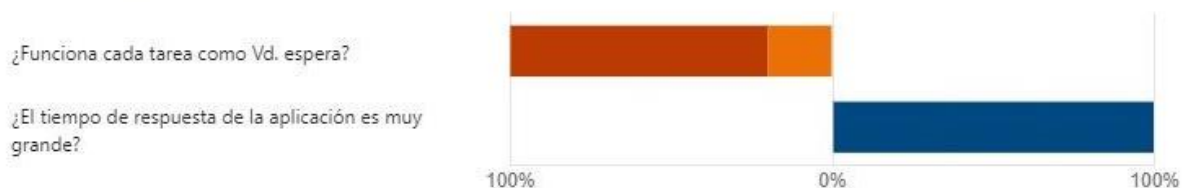
■ Siempre ■ Frecuentemente ■ Ocasionalmente ■ Nunca



31. Funcionalidad (0 punto)

[Más detalles](#)

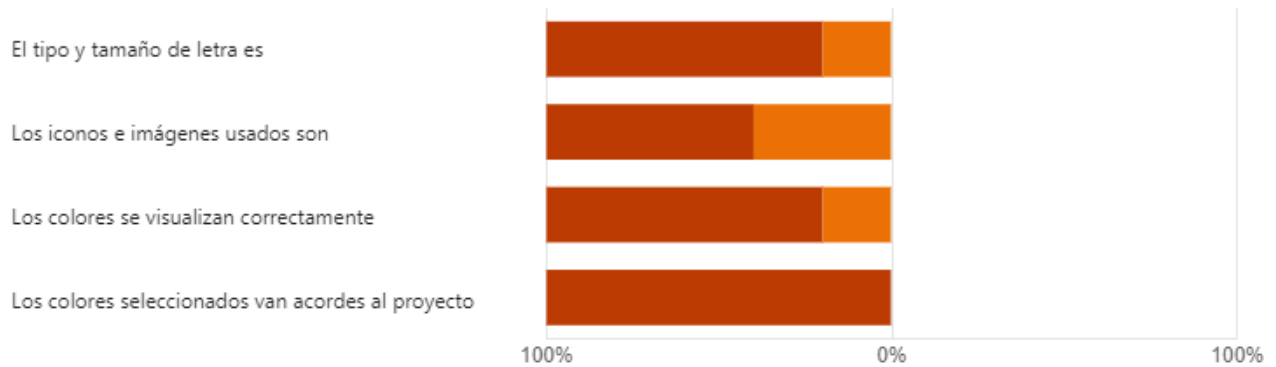
■ Siempre ■ Frecuentemente ■ Ocasionalmente ■ Nunca



32. Aspectos gráficos (0 punto)

[Más detalles](#)

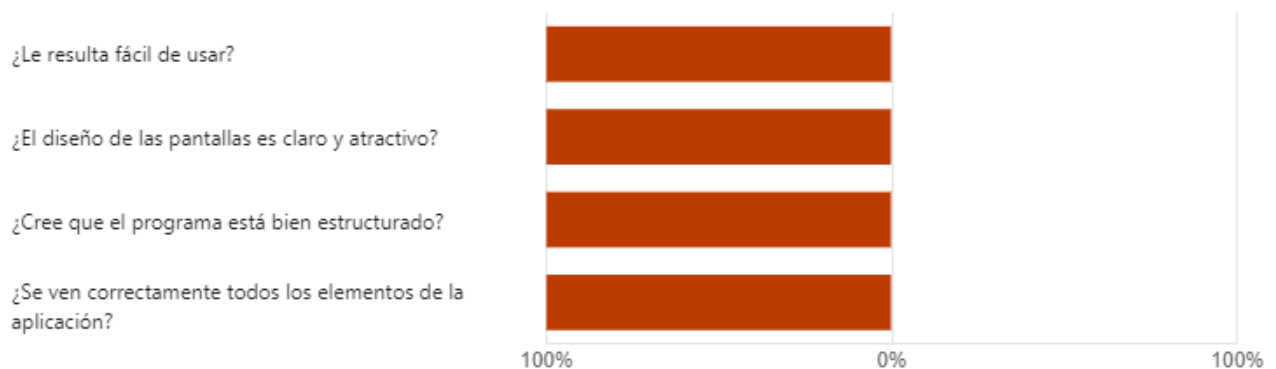
■ Muy adecuado ■ Adecuado ■ Poco adecuado ■ Nada adecuado



33. Aspectos gráficos (0 punto)

[Más detalles](#)

■ Sí ■ No ■ A veces



34. Observaciones

2 Respuestas

ID ↑	Nombre	Respuestas
1	anonymous	El botón de solicitar el día libre sería más fácil de localizar con un color más llamativo.
2	anonymous	Los colores de la aplicación acorde con el color oficial de renfe ayudan a confiar en la aplicación y saber para que sirve

35. Aspectos a mejorar

1 Respuestas

ID ↑	Nombre	Respuestas
1	anonymous	Añadiría algo que me indicara que puedo hacer click en el empleado en la gestión de solicitudes como administrador.

Ilustración 102. Ejecución de pruebas de usabilidad - preguntas cortas

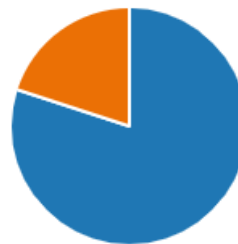
7.5.1.4 Cuestionario para el responsable de las pruebas

En cada una de las cinco ejecuciones de prueba realizada por los usuarios el responsable de pruebas ha realizado simultáneamente un cuestionario cuyos resultados se exponen a continuación.

1. El usuario comienza a trabajar de manera rápida por las tareas (0 punto)

[Más detalles](#)

- El usuario comienza rápido 4
- El usuario tarda en comenzar 1



2. El tiempo en realizar cada tarea

Respuestas

El usuario realiza a excepción de la primera, las tareas de manera rápida
El tiempo en realizar cada tarea es rápido
El usuario realiza las tareas de manera rápida a excepción de la tarea número 24
El usuario realiza todas las tareas de manera ágil
El tiempo en realizar cada tarea es mínimo



3. Errores leves cometidos

Respuestas

El usuario no consigue encontrar el mapa de las estaciones sin ayuda

No se ha cometido ningún error

No se comete ningún error

El usuario confunde el botón para solicitar un intercambio con el botón para solicitar un día libre. Se recalca al usuario el título que tiene esa pantalla, y corrige la acción.

No se comete ningún error

4. Errores graves cometidos

Respuestas

No se ha cometido ningún error

No se comete ningún error

El usuario no comete ningún error grave

No se comete ningún error

5. Dudas que han surgido

Respuestas

El usuario agradecería confirmación de que ha aceptado una solicitud

No surge ninguna duda significativa

El usuario no comenta ninguna duda durante la prueba

No surge ninguna duda significativ

6. El usuario necesita mucho seguimiento para completar las tareas

Respuestas

El usuario necesita ayuda para completar la primera tarea
No se necesita mucho seguimiento. Se proporciona la información de que se pueden seleccionar elementos en alguna de las tablas
El usuario necesita ayuda para localizar el componente para solicitar un día libre (tarea 24), una vez es localizado, el usuario realiza la tarea sin problema alguno
El usuario necesita ayuda con la diferencia entre los botones de solicitar un intercambio y solicitar un día libre.
El usuario completa las tareas sin ningún tipo de ayuda

Ilustración 103. Ejecución pruebas de usabilidad - cuestionario responsable de pruebas

Tras la ejecución de las pruebas de usabilidad se han realizado los siguientes cambios:

- **Cambio 1:** Se detecta que uno de los campos tiene la palabra “Tren” mal escrita y se corrige.
- **Cambio 2:** Se modifica la opción de menú para acceder al mapa y los horarios y pasa de llamarse “Consultar horarios” a “Mapa y horarios”.
- **Cambio 3:** Se resaltan las filas de las tablas que se pueden pulsar y se cambia la forma del ratón al pasarlo por encima.
- **Cambio 4:** Se resalta el botón de solicitar un día libre
- **Cambio 5:** Se añade un aviso cuando el administrador acepta una solicitud, indicando que la acción se ha llevado a cabo correctamente.

7.6 CSI 6: ELABORACIÓN DE LOS MANUALES

En esta sección se elaboran los manuales de usuario principales para la instalación ejecución y manejo de la aplicación.

7.6.1 Manual de Instalación

El manual de instalación contempla todos los pasos necesarios para instalar el sistema.

7.6.1.1 Instalar Java

Para instalar Java [15] en su versión 8 en Windows se deben de seguir los siguientes pasos:

1. Acceder a la página oficial de [Java](#)
2. Descargar la versión correspondiente
3. Una vez aparezca la ventana del instalador pulsar “Ejecutar”
4. Una vez finalice el proceso de instalación pulsar “Instalar”

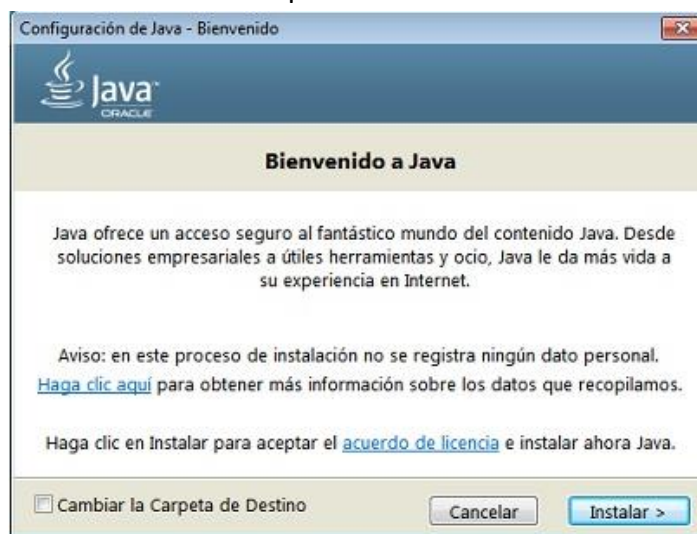


Ilustración 104. Configuración de Java

5. En caso de que el instalador solicite la instalación de programas adicionales, seleccione los deseados y pulse “Siguiente”
6. Cuando aparezca el cuadro final indicando que se ha completado la instalación haga click en “Cerrar”

Se puede encontrar un manual más detallado a continuación: [Manual Oficial de instalación Java](#)

7.6.1.2 Instalar MySQL

Para instalar MySQL se deben de seguir los siguientes pasos:

1. Acceder a la [página oficial](#) para descargar MySQL.
2. Descargar el primer zip pulsando “Download” para Windows

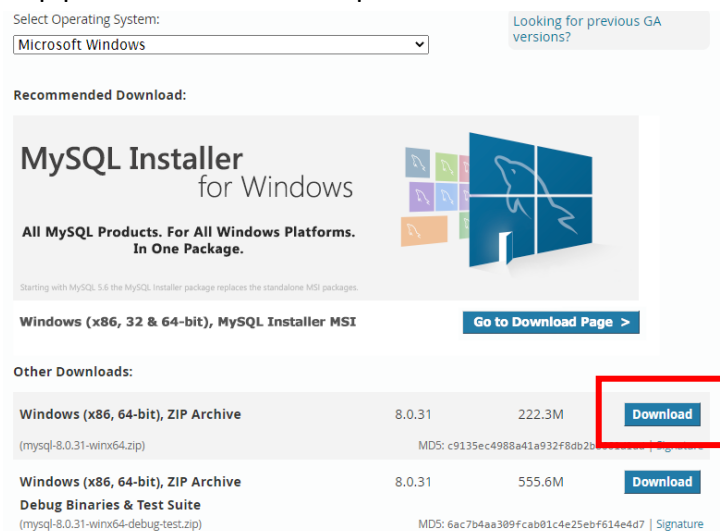


Ilustración 105. Descargar MySQL

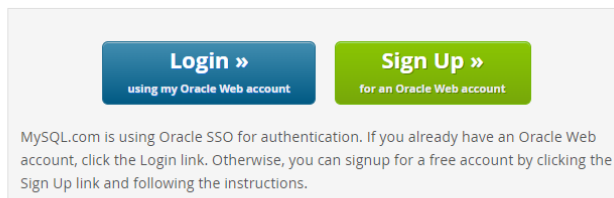
3. No es necesario iniciar sesión ni registrarse, para continuar pulsar sobre el enlace señalado en la Ilustración 106.

MySQL Community Downloads

Login Now or Sign Up for a free account.

An Oracle Web Account provides you with the following advantages:

- Fast access to MySQL software downloads
- Download technical White Papers and Presentations
- Post messages in the MySQL Discussion Forums
- Report and track bugs in the MySQL bug system



No thanks, just start my download.

Ilustración 106. Confirmación descarga MySQL

4. Una vez se haya completado la descarga, se procede a realizar la configuración deseada.
5. En el último paso, para configurar la conexión introducimos el nombre de conexión, el hostname y el port a libre elección.
6. El nombre de usuario y contraseña son los proporcionados a continuación, que a la vez son los mismos que los localizados en el archivo “application.properties” de la aplicación.
 - Nombre de usuario: root
 - Contraseña: TFG266047
7. Se finaliza la ejecución pulsando el botón “Ok”.

7.6.2 Manual de Ejecución

En este manual se incluyen todos los pasos necesarios para el arranque y detención de nuestro sistema.

Los valores establecidos por defecto en las configuraciones del back-end y del front-end son los citados a continuación, cabe destacar que todos ellos son modificables a necesidad del usuario.

- **Puerto back-end:** localhost: 8095
- **Puerto front-end:** localhost:4200
- **Puerto base de datos:** localhost:3307
- **Credenciales BD:**
 - o Nombre de usuario: root
 - o Contraseña: TFG266047

Todas las configuraciones de la parte back-end pueden verse y configurarse en el archivo *application.properties*. En el front-end el archivo *launch.json* indica que el puerto a utilizar es el 4200.

7.6.2.1 Creación de los materiales de ejecución

7.6.2.1.1 Jar back-end

Para la creación del JAR que permite ejecutar la parte back end del proyecto, se ha importado el mismo como proyecto Maven en Spring Tool Suite. Una vez el archivo se ha cargado correctamente, se ejecuta con “Maven install” como se puede ver en la Ilustración 107. Creación jar -maven install

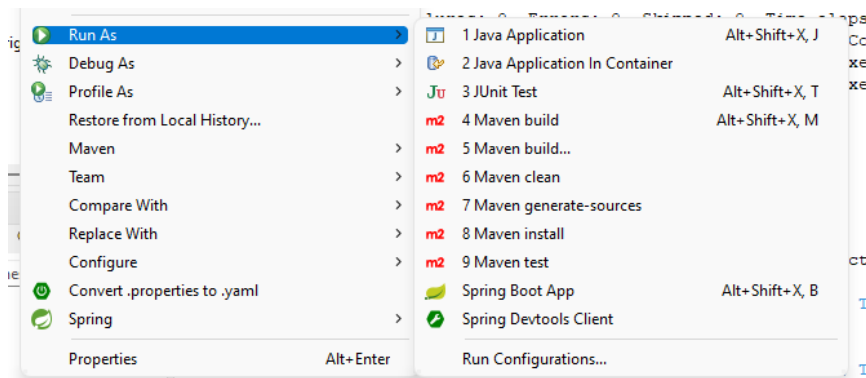


Ilustración 107. Creación jar -maven install

El archivo jar generado se almacena en la carpeta *target*.

7.6.2.1.2 Archivo comprimido front-end

Para la parte back-end de la aplicación se ha generado un zip conteniendo todos los archivos necesarios para la ejecución. El comando ejecutado para la creación de este zip es **ng build [nombre proyecto]**

El resultado de esta ejecución crea un directorio *dist* que contienen los archivos necesarios.

7.6.2.2 Ejecución de los materiales

Para ejecutar el proyecto se debe hacer uso de los archivos JAR y zip mencionados en el apartado anterior.

7.6.2.2.1 Back-end

Una vez generado el JAR, u obtenido del material entregado en los anexos (*UO266047BackEnd.jar*), se debe ejecutar el siguiente comando desde la carpeta que contiene el archivo:

```
java -jar UO266047BackEnd.jar
```

La aplicación ya está funcionando en el puerto 8095. Para realizar la carga de datos en la base de datos se debe ejecutar desde la misma el script de carga de datos proporcionado en los anexos con nombre *scriptCarga.sql*; para ello:

Desde MySQL Workbench seleccionamos Server → Data import en la barra de menú superior.

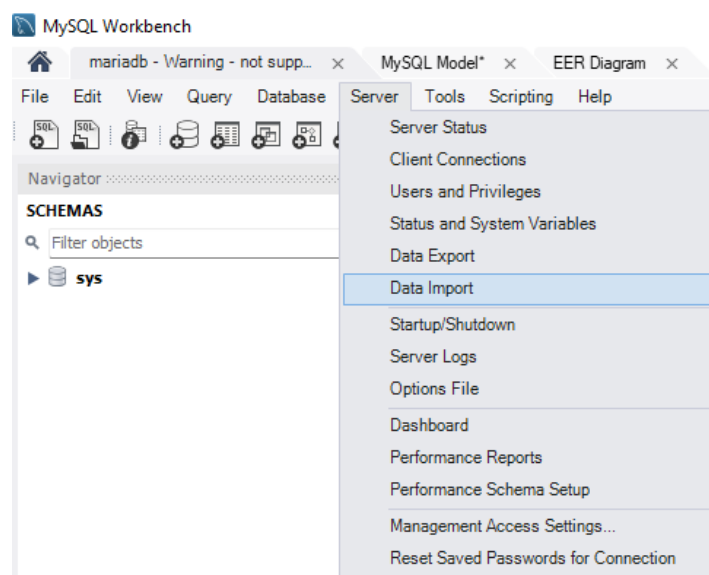


Ilustración 108. Carga script – menú import

Nos aparecerá el siguiente menú el en que debemos de seleccionar la ruta y script a importar y generar un nuevo esquema con el nombre **"bd_uo266047"**

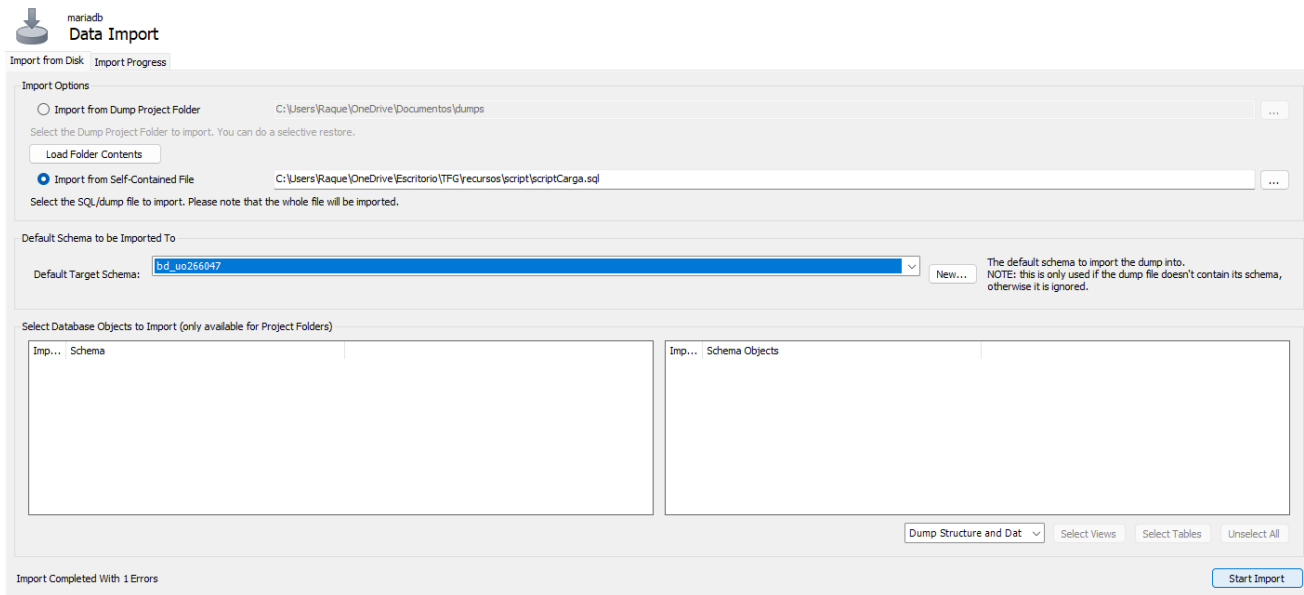


Ilustración 109. Carga script - configuración import

Finalizamos pulsando el botón “Start import”.

7.6.2.2.2 Front-end

Una vez tenemos los archivos necesarios para la ejecución, o bien hemos descomprimido el archivo *UO266047FrontEnd.zip* entregado en los anexos, nos situamos en esa carpeta. Ejecutamos los comandos

```
npm install http-server -g
```

```
http-server --port 4200
```

Una vez realizada esta acción, la parte front-end de la aplicación estará lista para usar y será accesible a través de la dirección **http://localhost:4200/**

7.6.3 Manual de Usuario

El manual de usuario de la aplicación, redactado en este apartado, ayuda a los usuarios finales a comprender cómo funciona la aplicación.

El manual de usuario se ha dividido en varios subapartados dependiendo del rol del usuario en la aplicación.

7.6.3.1 Usuario no identificado

7.6.3.1.1 Pantalla principal

Cuando el usuario accede a la aplicación la pantalla principal que observará será la ilustración 110; esta consta de una barra de navegación fija, la zona central cambiante dependiendo de la pantalla en la que se encuentre y el pie de página fijo también.

Gracias a la barra de navegación podrá acceder a cada una de las pantallas de la aplicación.

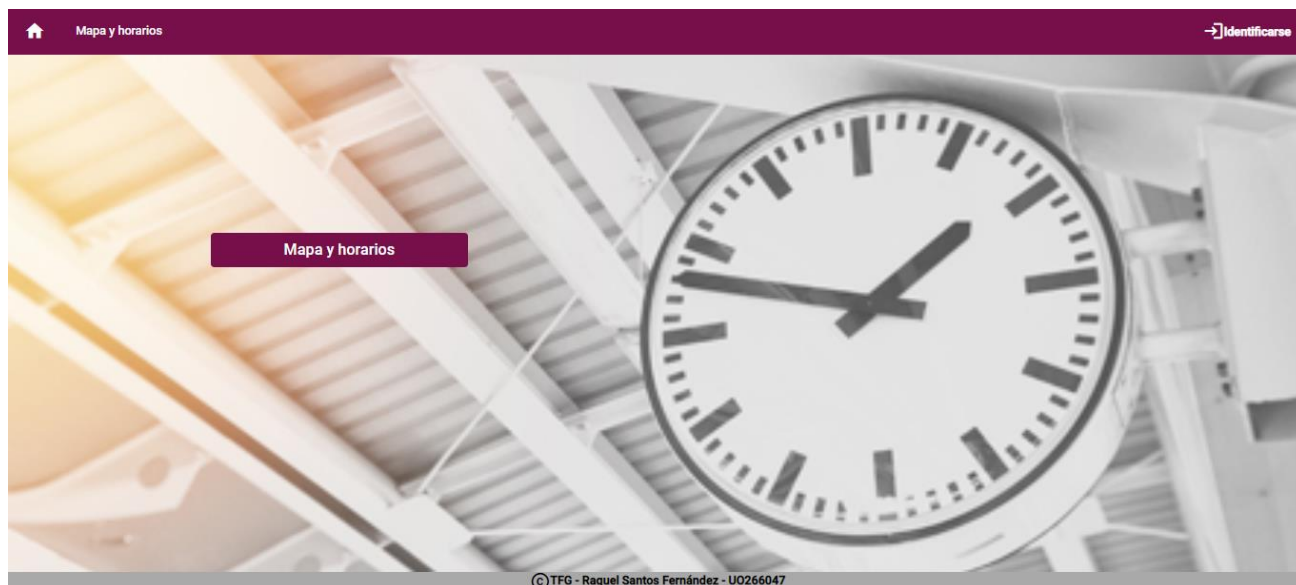


Ilustración 110. Pantalla principal

7.6.3.1.2 Pantalla consulta de horarios

A través del botón “Mapa y horarios” el usuario podrá visualizar el mapa de las estaciones y realizar la consulta de horarios entre dos estaciones.

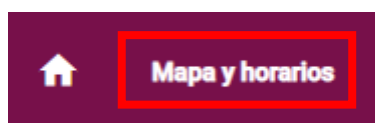


Ilustración 111. Botón mapa y horarios

Una vez el usuario está en la página, puede comprobar las estaciones existentes examinando el mapa.

Para acceder a los horarios entre dos estaciones se debe seleccionar la estación de inicio pulsando en el desplegable y eligiendo la estación deseada; lo mismo para la estación de destino. Una vez las dos estaciones están seleccionadas, se debe pulsar el botón “Consultar”.

Origen ▼ Destino ▼ Consultar



Ilustración 112. Mapa y horarios

Los horarios entre las dos estaciones aparecerán por pantalla.

7.6.3.1.3 Pantalla inicio de sesión

Para el inicio de sesión el usuario debe pulsar el botón “Identificarse” del menú superior.




Ilustración 113. Botón identificarse

Este le permitirá acceder a la pantalla de inicio de sesión. Para poder iniciar sesión debe introducir los datos de nombre de usuario y contraseña correctos en el formulario y pulsar el botón de “Acceder”.

Introduzca sus credenciales

Usuario *

Contraseña * 

Acceder

Ilustración 114. Inicio de sesión

7.6.3.2 Usuario trabajador identificado

Una vez el usuario inicie sesión, aparecerá de nuevo en la pantalla principal.

7.6.3.2.1 Pantalla consultar jornada

Para acceder a la pantalla de consultar jornada el usuario debe seleccionar la opción “Gestión de jornada” de la barra de navegación desplegando el menú, para así poder pulsar el botón de “Consultar jornada” y pasar a la pantalla correspondiente.

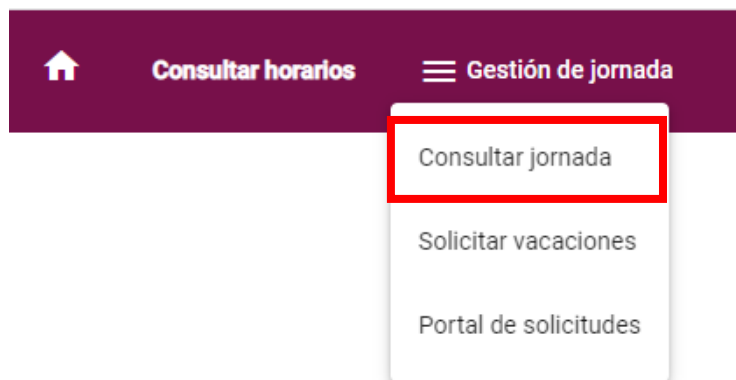


Ilustración 115. Menú consultar jornada

Para consultar la jornada de una fecha específica, el usuario trabajador solamente debe seleccionar el día que desee en el calendario, la jornada asignada a ese día aparecerá en pantalla.

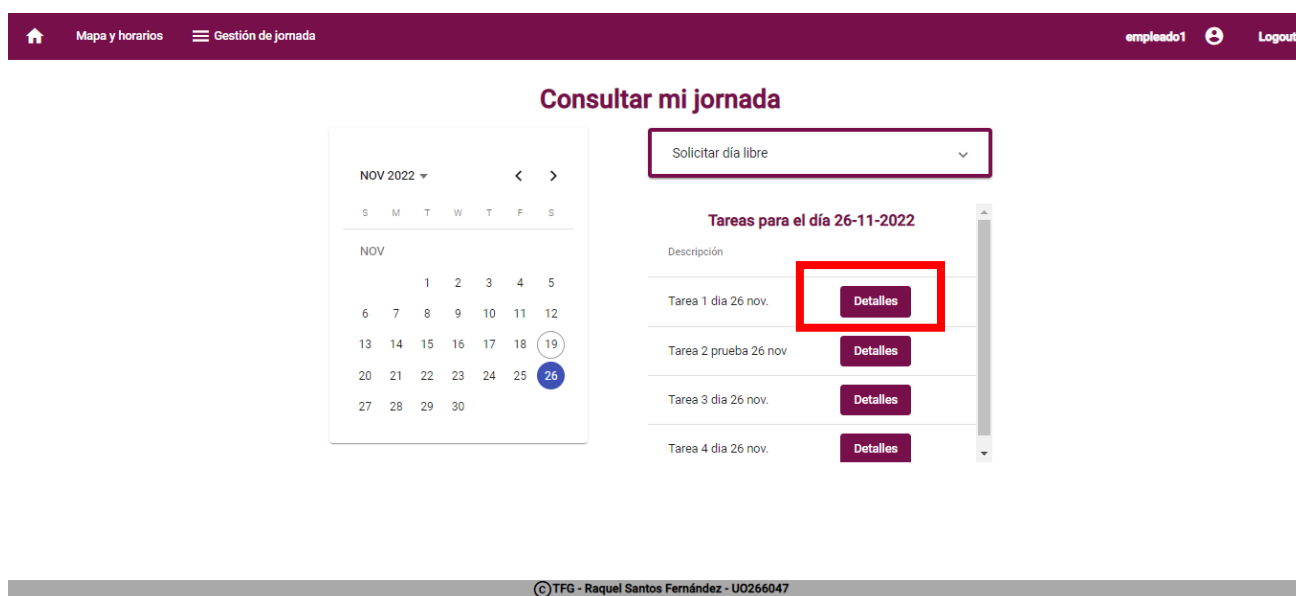


Ilustración 116. Consultar jornada

En caso de querer seleccionar un día libre, el usuario debe seleccionar el día que desea en el calendario y pulsar en “Solicitar día libre”; esto hará que se despliegue un formulario para indicar las razones de la ausencia. Una vez el usuario rellene el formulario debe pulsar el botón “Solicitar”.

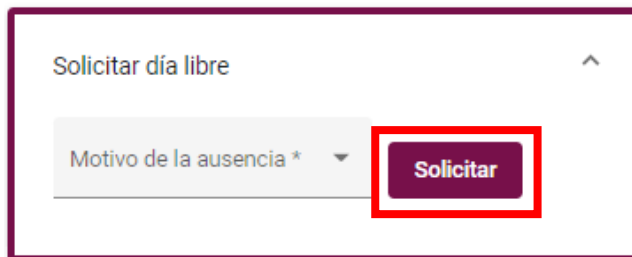


Ilustración 117. Consultar jornada - Solicitar día libre

En caso de que el usuario quiera ver los detalles de una tarea concreta asignada a una jornada deberá hacer click en el botón “Detalles” (tras haber realizado el procedimiento para la consulta de la jornada anteriormente mencionado) y esto hará que aparezca una ventana emergente con la información.

Si el usuario desea añadir una incidencia al tren que ha sido asignado a esa tarea, debe rellenar el campo de “Nueva incidencia” y pulsar el botón “+”

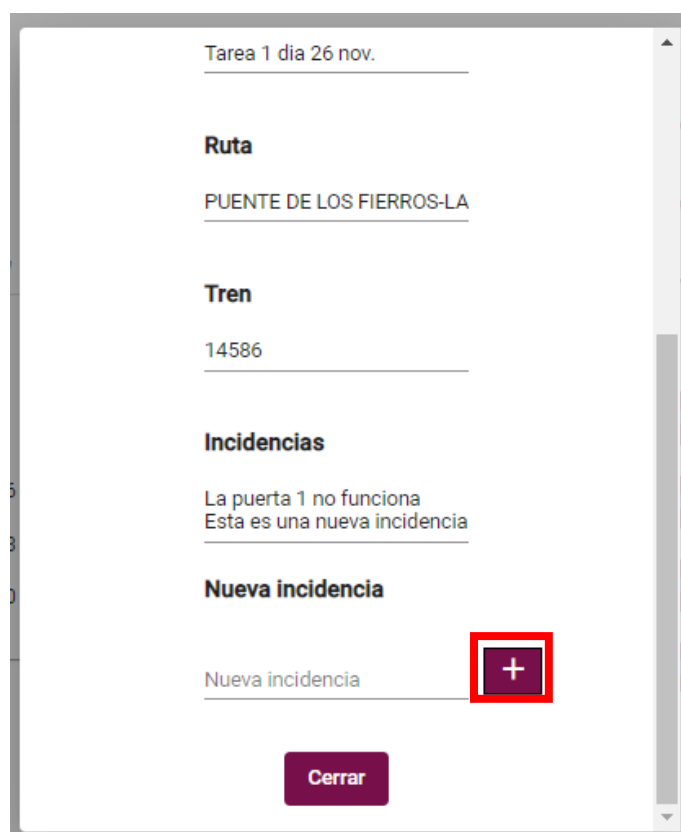


Ilustración 118. Detalles de tarea

7.6.3.2.2 Pantalla solicitar vacaciones

Para acceder a la pantalla de consultar jornada el usuario debe seleccionar la opción “Gestión de jornada” de la barra de navegación desplegando el menú, para así poder pulsar el botón de “Solicitar vacaciones” y pasar a la pantalla correspondiente.

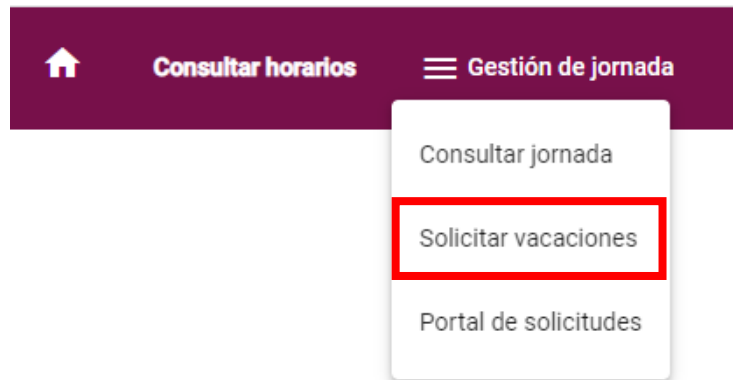


Ilustración 119. Menú solicitar vacaciones

Para solicitar un periodo de vacaciones, el empleado debe seleccionar uno de los disponibles en la lista y pulsar el botón “Solicitar”

Solicitud de vacaciones

Selecciona un periodo de vacaciones

Periodo de invierno	Periodo de verano
21/12/2022 - 09/01/2023	01/06/2022 - 15/06/2022
01/10/2022 - 20/10/2022	01/07/2022 - 15/07/2022
01/12/2022 - 20/12/2022	16/06/2022 - 30/06/2022
21/10/2022 - 08/11/2022	16/07/2022 - 31/07/2022
01/09/2022 - 15/09/2022	12/05/2022 - 31/05/2022
01/08/2022 - 15/08/2022	12/03/2022 - 31/03/2022
16/09/2022 - 30/09/2022	02/04/2022 - 21/04/2022
16/08/2022 - 31/08/2022	23/04/2022 - 11/05/2022

Solicitar

Ilustración 120. Solicitar vacaciones

En caso de que el empleado ya haya realizado una solicitud de vacaciones, la pantalla que se mostrará será la de la Ilustración 121. En caso de que todas las solicitudes tengan un estado “RECHAZADA” el usuario podrá volver a solicitar un nuevo periodo pulsando el botón “Solicitar vacaciones”.

Solicitud de vacaciones

Las solicitudes de vacaciones existentes son:

Periodo	Estado
01-06-2022 – 15-06-2022	RECHAZADA
21-10-2022 – 08-11-2022	RECHAZADA
16-07-2022 – 31-07-2022	RECHAZADA
21-12-2022 – 09-01-2023	RECHAZADA

[Solicitar vacaciones](#)

Ilustración 121. Solicitud de vacaciones - existente

7.6.3.2.3 Pantalla portal de solicitudes

Al igual que para las dos pantallas anteriores se debe acceder a la opción “Gestión de jornada” y al submenú “Portal de solicitudes” para pasar a la página correspondiente.



Ilustración 122. Menú portal de solicitudes

El usuario puede navegar por las pestañas del componente principal consultando las solicitudes de intercambio propuestas por otros empleados, sus propias solicitudes y las solicitudes de vacaciones.

Portal de solicitudes

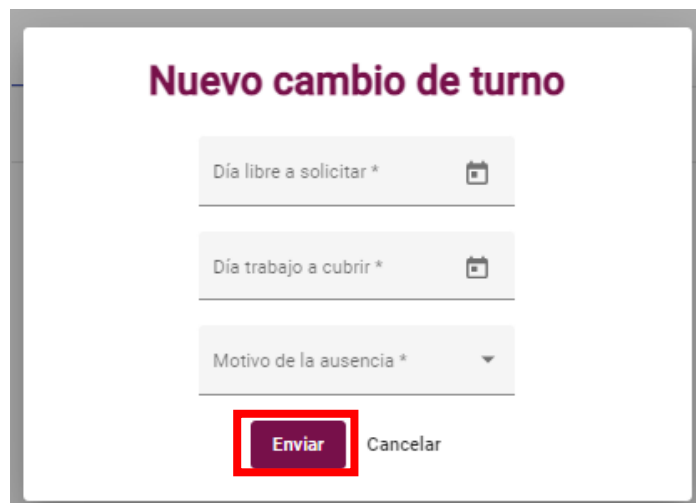
Solicitar intercambio

Solicitudes intercambio		Mis solicitudes	Vacaciones	
Día a cubrir	Día de descanso	Empleado	Acción	
21-11-2022	23-11-2022	empleado1	<input checked="" type="checkbox"/>	🔍
26-11-2022	30-11-2022	empleado1	<input checked="" type="checkbox"/>	🔍

Ilustración 123. Portal de solicitudes

En caso de querer aceptar una solicitud de intercambio de las existentes, debe pulsar el botón con el icono de un tic de la correspondiente solicitud.

En caso de querer enviar una nueva solicitud de intercambio el usuario debe pulsar el botón “Solicitar intercambio”. Esto le mostrará una ventana emergente con un formulario que debe rellenar. Una vez el formulario esté debidamente cumplimentado deberá pulsar el botón “Enviar” para realizar la solicitud.



Nuevo cambio de turno

Día libre a solicitar *

Día trabajo a cubrir *

Motivo de la ausencia *

Ilustración 124. Solicitar intercambio

7.6.3.2.4 Pantalla ver perfil

Para acceder a la pantalla del perfil, se debe pulsar en el icono del menú que representa un usuario; si el usuario se pone encima de dicho icono le aparecerá un mensaje explicativo o tooltip que indica esta acción (ver perfil), como se puede ver en la Ilustración 125

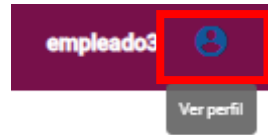


Ilustración 125. Menú ver perfil

El único campo modificable del formulario es la contraseña, para poder modificarla el usuario debe hacer clic sobre el botón “Modificar contraseña”.

Identificado como empleado3

Nombre	Empleado3
Apellidos	Prueba
Nombre de usuario	empleado3
Rol	MAQUINISTA
Días libres disponibles	100
Modificar contraseña	

Ilustración 126. Ver y modificar contraseña

Al hacer clic sobre el icono de editar aparecerá una ventana emergente con un formulario de verificación de nueva contraseña, una vez el usuario rellene los campos y pulse “Modificar” la contraseña será modificada.

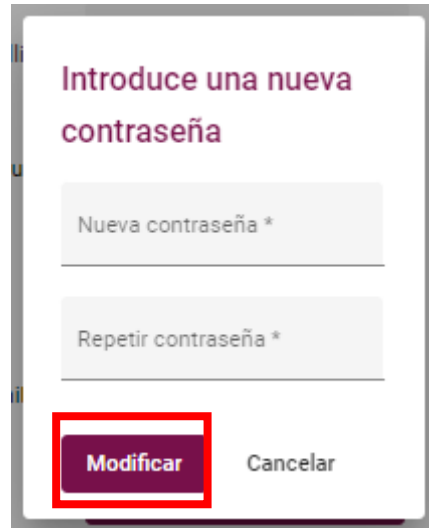


Ilustración 127. Nueva contraseña

7.6.3.3 Usuario administrador identificado

7.6.3.3.1 Pantalla ver solicitudes

Para acceder a esta pantalla se debe de seleccionar “Ver solicitudes” en el menú superior.



Ilustración 128. Menú ver solicitudes

Una vez en la pantalla se pueden ver todas las solicitudes pendientes. El administrador podrá aceptar o rechazar una solicitud pulsando sobre los respectivos botones.

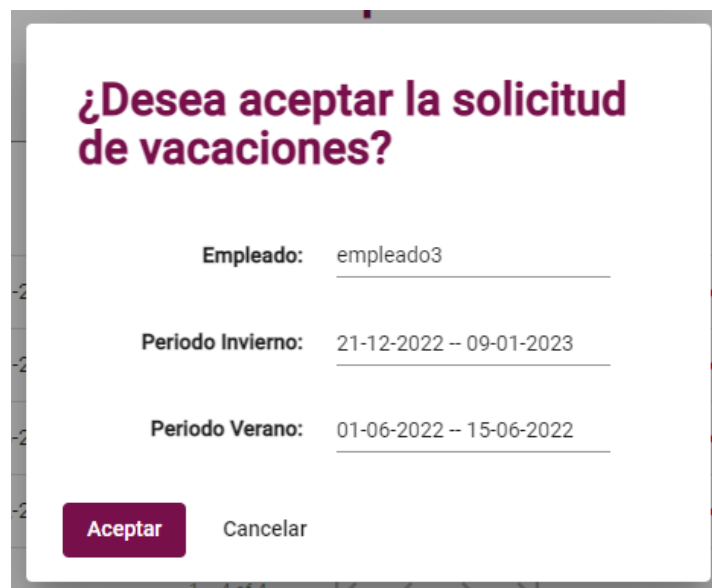
Solicitudes pendientes

Fecha	Motivo	Empleado		
25-11-2022	LICENCIA	empleado1	✓	✗
22-11-2022	LICENCIA	empleado1	✓	✗

1 - 2 of 2 |< < > >|

Ilustración 129. Ver solicitudes

En caso de que la solicitud sea de tipo vacaciones aparecerá una ventana emergente como la de la Ilustración 130 para confirmar la aceptación o rechazo del periodo.



¿Desea aceptar la solicitud de vacaciones?

Empleado: empleado3

Periodo Invierno: 21-12-2022 -- 09-01-2023

Periodo Verano: 01-06-2022 -- 15-06-2022

Aceptar Cancelar

Ilustración 130. Mensaje confirmación solicitud

7.6.3.3.2 Pantalla gestionar usuarios y jornadas

Para acceder a esta pantalla se debe de seleccionar “Gestionar usuarios y jornadas” en el menú superior.



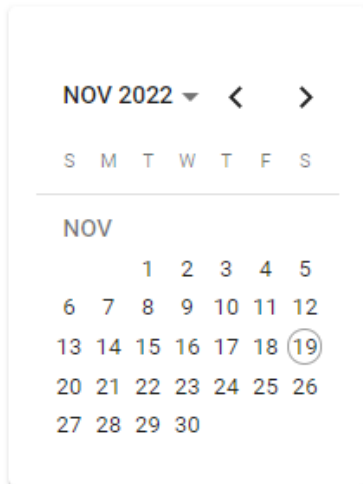
Ilustración 131. Menú gestionar usuarios y jornadas

En esta última pantalla se puede filtrar los usuarios por nombre de usuario, introduciendo este en el campo indicado para ello.

En segundo lugar, se puede añadir un nuevo usuario pulsando el botón “Nuevo usuario”; por el contrario, para eliminar un empleado se debe seleccionar el botón con icono de papelera. Para ver sus detalles pulsando el botón con icono de lupa.

Gestión de usuarios y tareas

Filtrar por día



Filtrar por usuario

Buscar por nombre de usuario

Nuevo usuario

Nueva tarea

Nombre de usuario	Nombre	Rol		
empleado1	Empleado1	MAQUINISTA		
empleado2	Empleado2	MAQUINISTA		
empleado3	Empleado3	MAQUINISTA		

Ilustración 132. Gestión usuarios

Nuevo usuario

Nombre de usuario *

Nombre *

Apellidos *

DNI *

Email *

Rol *

Añadir Cerrar

Ilustración 133. Añadir nuevo usuario

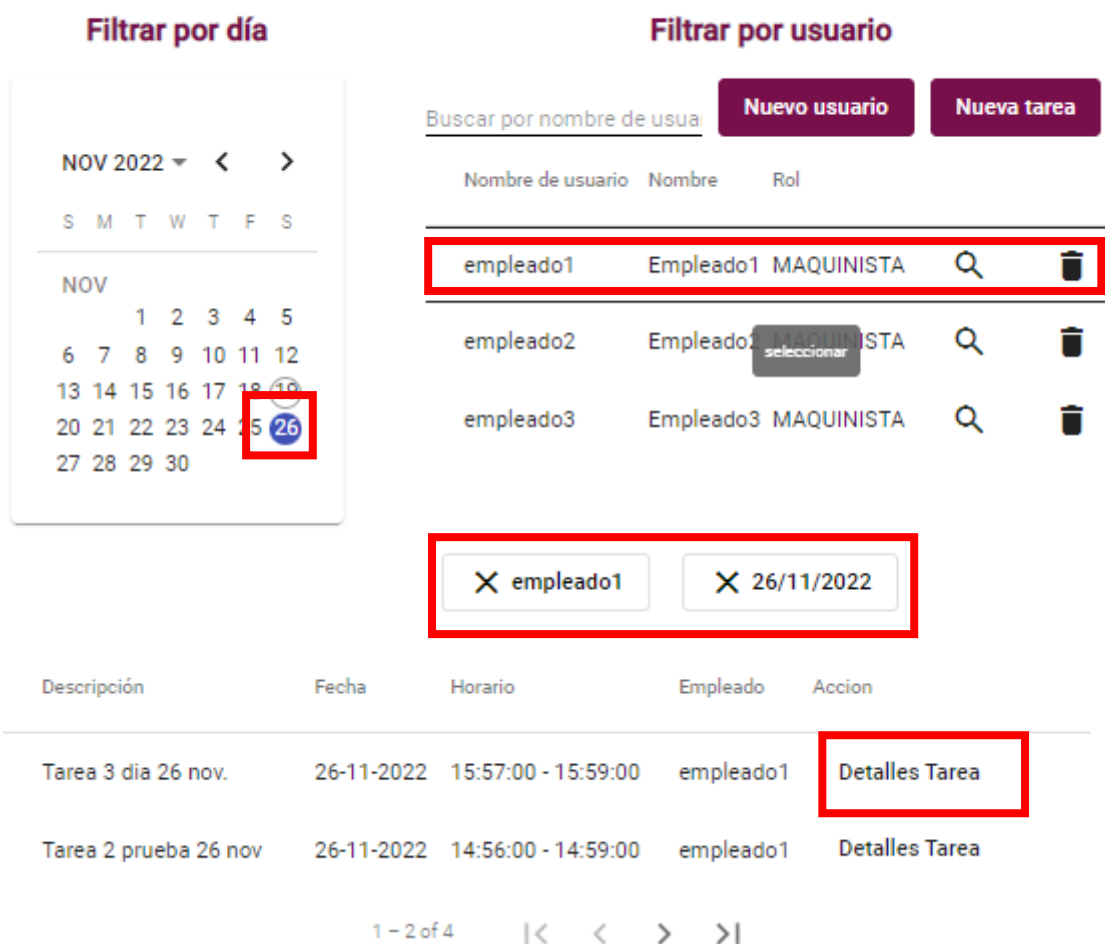
Para consultar la jornada de un día concreto, se debe seleccionar este en el calendario. Una vez seleccionado el día en la parte inferior de la pantalla se podrán ver todas las tareas de todos los empleados con jornada ese día.

En caso de querer ver las tareas de un empleado para una jornada concreta, además de seleccionar el día en el calendario se debe de seleccionar el empleado deseado en la lista.

En caso de querer eliminar la selección debemos de pulsar en el botón que representa el filtro del día o del empleado.

Adicionalmente, se podrá ver las tareas para todas las jornadas de un empleado, si se selecciona este y no se tiene ningún filtro de día aplicado.

Gestión de usuarios y tareas



Filtrar por día

NOV 2022 < >

S M T W T F S

NOV

1 2 3 4 5

6 7 8 9 10 11 12

13 14 15 16 17 18 19

20 21 22 23 24 25 26

27 28 29 30

Filtrar por usuario

Buscar por nombre de usua **Nuevo usuario** **Nueva tarea**

Nombre de usuario	Nombre	Rol		
empleado1	Empleado1	MAQUINISTA	🔍	🗑️
empleado2	Empleado2	MAQUINISTA	🔍	🗑️
empleado3	Empleado3	MAQUINISTA	🔍	🗑️

🗑️ empleado1 🗑️ 26/11/2022

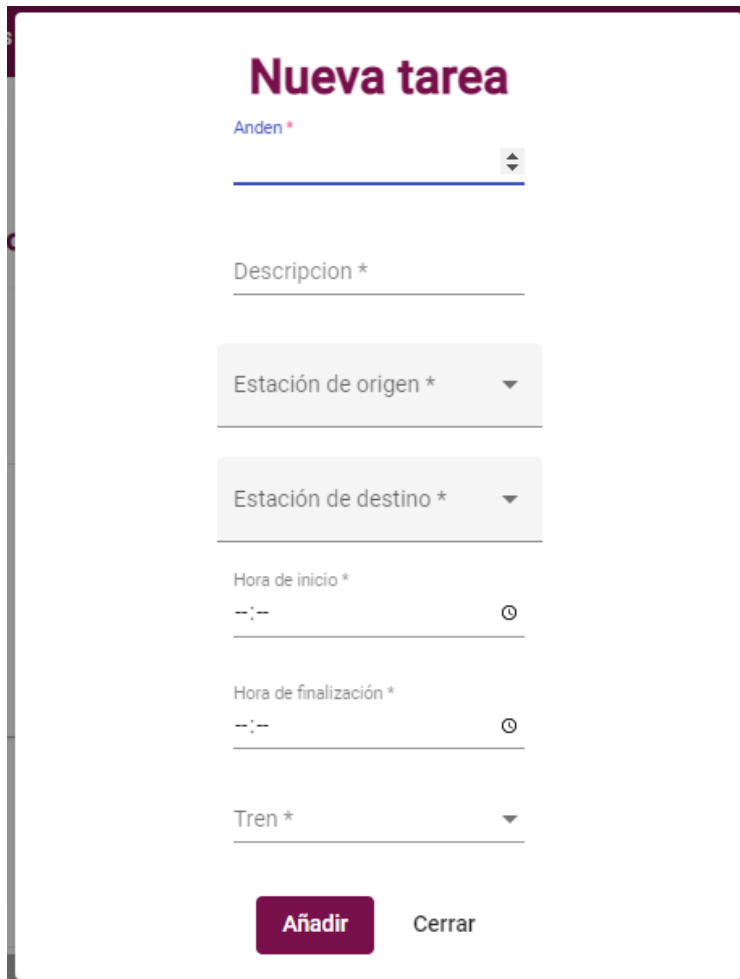
Descripción	Fecha	Horario	Empleado	Accion
Tarea 3 día 26 nov.	26-11-2022	15:57:00 - 15:59:00	empleado1	Detalles Tarea
Tarea 2 prueba 26 nov	26-11-2022	14:56:00 - 14:59:00	empleado1	Detalles Tarea

1 - 2 of 4 |< < > >|

Ilustración 134. Gestión de usuarios y jornadas

Al igual que para el empleado trabajador identificado, se puede ver los detalles de una tarea, pulsando el botón “Detalles” de la tarea deseada.

Para añadir una nueva tarea se debe seleccionar un día en el calendario, un empleado en la lista y pulsar sobre el botón “Nueva tarea”. Aparecerá una ventana emergente con el formulario para la creación de la nueva tarea.



The image shows a modal window titled "Nueva tarea" (New task). The form contains the following fields and controls:

- Anden ***: A dropdown menu for selecting a station.
- Descripcion ***: A text input field for the task description.
- Estación de origen ***: A dropdown menu for selecting the starting station.
- Estación de destino ***: A dropdown menu for selecting the destination station.
- Hora de inicio ***: A time selection field with a "--:--" format and a clear button.
- Hora de finalización ***: A time selection field with a "--:--" format and a clear button.
- Tren ***: A dropdown menu for selecting a train.
- Añadir**: A purple button to save the task.
- Cerrar**: A text button to close the modal.

Ilustración 135. Nueva tarea



Capítulo 8 APÉNDICES

8.1 PLAN DE GESTIÓN DE RIESGOS

Según el PMBOK [31] la Gestión de Riesgos incluye los procesos para llevar a cabo la planificación de la gestión de riesgos, así como la identificación, análisis, planificación de respuesta y el control de los riesgos de un proyecto. Con ello se pretende minimizar los riesgos negativos y aumentar la probabilidad e impacto de los positivos.

8.1.1 Planificar la Gestión de Riesgos

Como metodología para planificar la Gestión de Riesgos se ha seleccionado una que combina la propuesta por Boehm [32], pero que además añade parte de la metodología propuesta por el PMBOK incluyendo el paso de Planificar la Gestión de Riesgos, lo que permite incluir los fundamentos para las futuras decisiones de los riesgos.

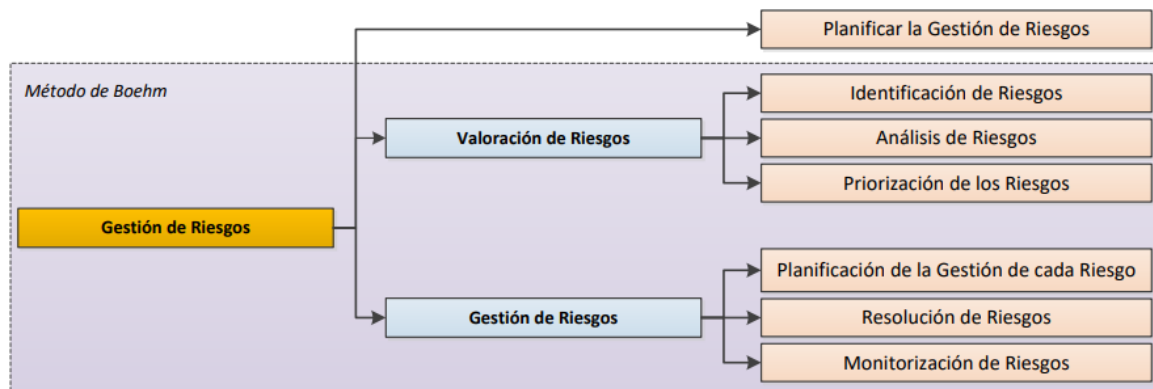


Ilustración 136. Diagrama gestión de riesgos

Los pasos principales son:

1. **Planificar la Gestión de Riesgos:** En esta etapa se deciden cuáles van a ser los fundamentos y políticas que se van a aplicar a lo largo de toda la Gestión de Riesgos.
2. **Valoración de Riesgos:** Etapa compuesta por la fase de Identificación de Riesgos, Análisis y Priorización de Riesgos. Se lleva a cabo antes de comenzar la fase de desarrollo del proyecto para identificar, priorizar y estudiar el impacto de los riesgos. También puede ser modificada a lo largo de la vida del proyecto
3. **Gestión de Riesgos:** Esta última etapa está formada por la Planificación de la Gestión de cada Riesgo, la Resolución de Riesgos y la Monitorización de riesgos, esta fase comienza también antes de comenzar con el desarrollo del proyecto, pero cobra mayor sentido cuando se empieza con el mismo.

Para la clasificación de los riesgos se ha empleado la estructura de desglose de riesgos RBS del PMBOK que incluye las categorías y subcategorías a partir de las cuales los riesgos pueden surgir en un proyecto.

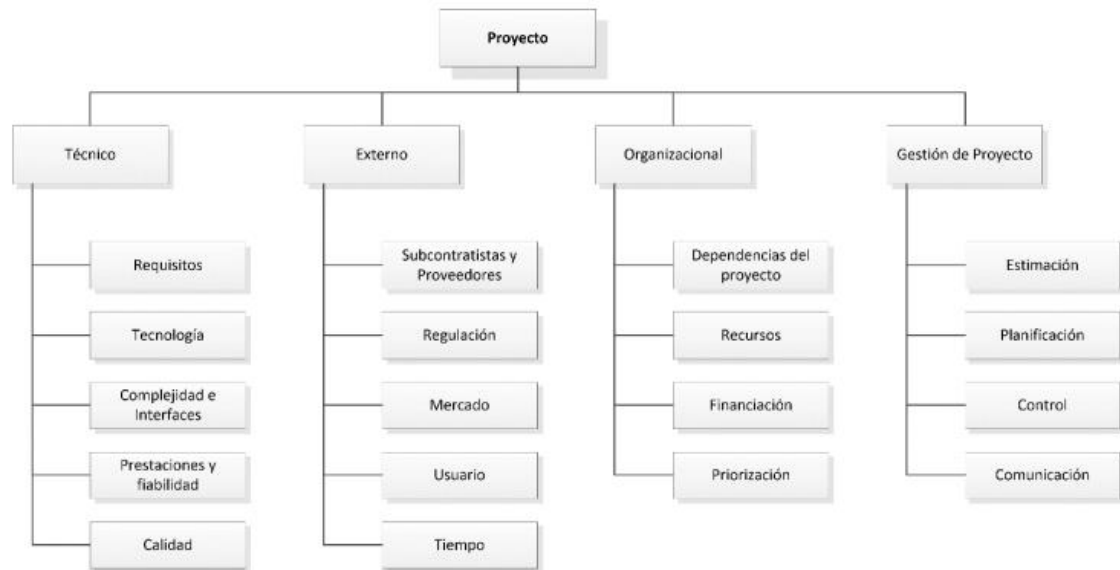


Ilustración 137. Estructura de desglose de riesgos

8.1.2 Identificación de Riesgos

La identificación de riesgos está desarrollada en el apartado *Identificación de Riesgos* de esta misma documentación. Generando una lista priorizada de los riesgos detectados e incluyendo una breve descripción de cada uno de ellos.

8.1.3 Registro de riesgos

En este apartado se establece una tabla indicando el identificador del riesgo, su nombre, su categoría, su probabilidad y su impacto.

La evaluación del impacto se realiza estudiando por una parte la probabilidad (porcentaje de probabilidad de que el riesgo ocurra) y por otra parte el impacto (efecto que ocasionaría en caso de producirse). Se usan como referencia las tablas mostradas a continuación:

Etiqueta	Valor de probabilidad
Muy bajo	10%
Bajo	30%
Medio	50%
Alto	70%
Muy alto	90%

Tabla 110 Valor de las probabilidades

Se hace uso de la escala de impacto para el coste, tiempo, alcance y calidad del proyecto.

Etiqueta	Valor de probabilidad
Inapreciable	5%
Bajo	10%
Medio	20%
Alto	40%
Crítico	80%

Tabla 111. Escala de impacto

Dado cada uno de los riesgos identificados en el apartado anterior se asigna un valor de prioridad y un valor de impacto para los cuatro objetivos principales. Para la asignación de valores numéricos se hace uso de la matriz de probabilidad e impacto. El resultado final nos permitirá priorizar los riesgos.

Probabilidad	Muy Alta	0,90	0,05	0,14	0,27	0,50	0,81
	Alta	0,70	0,04	0,11	0,21	0,39	0,63
	Media	0,50	0,03	0,08	0,15	0,28	0,45
	Baja	0,30	0,02	0,05	0,09	0,17	0,27
	Muy Baja	0,10	0,01	0,02	0,03	0,06	0,09
			0,05	0,15	0,30	0,55	0,90
			Inapreciable	Bajo	Medio	Alto	Crítico
			0,05	0,15	0,30	0,55	0,90
			Impacto				

Tabla 112. Matriz de probabilidad e impacto

ID	Nombre riesgo	Categoría	Probabilidad	Impacto				Impacto
				Presup.	Planific.	Alcance	Calidad	
1	Fallos en la estimación temporal	Gestión de proyecto (Estimación)	Media	Alto	Crítico	Bajo	Inapreciable	0,45
2	Fallos en las estimaciones de presupuestos	Gestión de proyecto (Estimación)	Media	Crítico	Medio	Bajo	Inapreciable	0,45
3	Se realizan funciones software innecesarias	Organización al (Recursos)	Alta	Medio	Alto	Medio	Medio	0,385
4	Falta de usabilidad	Externo (Usuario)	Media	Bajo	Medio	Alto	Alto	0,275
5	Requisitos mal definidos	Técnico (Requisitos)	Media	Medio	Alto	Alto	Alto	0,275
6	Inconsistencia a la hora de integrar front-end y back-end	Técnico (Tecnología)	Media	Medio	Alto	Alto	Alto	0,275
7	Cambios en el alcance del proyecto	Técnico (Requisitos)	Baja	Alto	Alto	Crítico	Alto	0,27

8	Formación escasa del personal	Técnico (Calidad)	Muy alta	Bajo	Bajo	Bajo	Medio	0,27
9	Falta de conocimiento del dominio o de la tecnología	Técnico (Complejidad)	Alta	Inapreciable	Inapreciable	Inapreciable	Medio	0,21
10	Pérdida de datos	Técnico (Prestaciones y fiabilidad)	Baja	Medio	Medio	Medio	Alto	0,165
11	Los empleados no se adaptan al nuevo sistema	Externo (Usuario)	Baja	Alto	Medio	Alto	Bajo	0,165
12	Fallos en la migración de datos	Técnico (Tecnología)	Media	Bajo	Medio	Medio	Medio	0,15
13	Cambios en las leyes	Externo (Regulación)	Baja	Bajo	Medio	Medio	Medio	0,09

Tabla 113. Registro de riesgos

8.1.4 Resolución de Riesgos

Una vez priorizados los riesgos tras haber estudiado su impacto se procede a proporcionar respuestas a los riesgos y a plantear estrategias de mitigación.

Se proponen cuatro estrategias de mitigación:

- **Eliminación del riesgo:** Eliminar los factores que pueden provocar el riesgo y con ello eliminar la posibilidad de aparición.
- **Transferencias del riesgo:** Transferir el riesgo a terceros.
- **Mitigación del riesgo:** Identificar los factores que desencadenan el riesgo y mantenerlos controlados.
- **Asumir el riesgo:** Se acepta que el riesgo existe y se convive con él.

ID	Nombre riesgo	Respuesta al riesgo	Estrategia
1	Fallos en la estimación temporal	Los fallos en la estimación temporal pueden ocurrir por razones ajenas al estudiante, por lo que se debe estimar con un poco de margen para tener en cuenta posibles contratiempos	Mitigar el riesgo
2	Fallos en las estimaciones de presupuestos	Los fallos en el presupuesto suelen ser algo común, por lo que se debe guardar cierta parte del presupuesto para posibles imprevistos	Mitigar el riesgo
3	Se realizan funciones software innecesarias	Es muy posible que el programador decida realizar funciones software que crea que ayuden al proyecto, pero en realidad no aporten ninguna mejora o cumplimenten los objetivos principales del proyecto, el riesgo se debe asumir	Asumir el riesgo

4	Falta de usabilidad	Realización de numerosas pruebas de usabilidad para mitigar en la medida de lo posible este riesgo	Mitigar el riesgo
5	Requisitos mal definidos	Para reducir al mínimo la definición errónea de requisitos se debe realizar un estudio intensivo del dominio, plantear las dudas existentes al tutor y en caso de que surgiesen nuevos a lo largo de la vida del proyecto estudiar su impacto en el sistema	Mitigar el riesgo
6	Inconsistencia a la hora de integrar front-end y back-end	A la hora de integrar los subsistemas y la parte de front-end y de back-end pueden producirse fallos de integración, esto se debe tener en cuenta y se debe diseñar y programar de cara a esta integración	Mitigar el riesgo
7	Cambios en el alcance del proyecto	Debemos tener en cuenta que el alcance del proyecto puede variar, pero con una buena planificación y definición inicial de requisitos este riesgo se debe mitigar en la medida de lo posible	Mitigar el riesgo
8	Formación escasa del personal	Se debe asumir que el personal de Renfe Cercanías Asturias no va a tener una formación concreta acerca de la tecnología usada, es por ello que debemos realizar un manual detallado.	Asumir el riesgo
9	Falta de conocimiento del dominio o de la tecnología	Se debe realizar un estudio intensivo previo para mitigar en la medida de lo posible este riesgo.	Mitigar el riesgo
10	Pérdida de datos	Se deben realizar varias copias de seguridad periódicas, o cuando se realicen cambios significativos en el proyecto. Las copias deben realizarse en varios lugares, para que en caso de error el trabajo se encuentre almacenado en varias plataformas	Eliminar el riesgo
11	Los empleados no se adaptan al nuevo sistema	Empleando un plan de formación apropiado podríamos hacer que los empleados se adaptasen lo más rápido posible al nuevo sistema	Mitigar el riesgo
12	Fallos en la migración de datos	La migración de datos a la nueva aplicación se debe realizar de manera cuidadosa y bajo supervisión de varios miembros de equipo, para eliminar el riesgo de la pérdida de datos o posibles incoherencias	Eliminar el riesgo
13	Cambios en las leyes	Este riesgo no es algo que podamos controlar, por lo que al no poder evitarlo debemos asumirlo y adecuarnos de la mejor manera posible en caso de que suceda	Asumir el riesgo

Tabla 114. Resolución de riesgos



8.2 CONCLUSIONES

El proyecto realizado cumple con los principales objetivos establecidos; la creación de una aplicación de gestión de personal para la compañía Renfe Cercanías Asturias; en cuanto a la visualización de información para pasajeros, se ha conseguido mostrar los horarios entre dos estaciones, y visualizar las mismas en el mapa una vez se realiza la consulta. Se pretendía realizar cinco subsistemas (Gestión de empleado, Gestión de jornada laboral, Gestión de vacaciones y días libres, Portal de solicitudes entre empleados y Gestión de información para viajeros), y una vez se ha concluido el proyecto se afirma que se han creado y su funcionalidad es la esperada.

En cuanto a las tecnologías empleadas cabe resaltar el poco conocimiento que se tenía de Angular al inicio del proyecto, ya que es una tecnología con la que no he trabajado en el grado, y las dificultades que se encontraron por ello a lo largo del mismo. Aun así, creo que la elección de las tecnologías ha sido la correcta porque me ha permitido aprender mucho sobre este framework. Por otra parte, el lenguaje y framework empleado para desarrollar la parte back-end de la aplicación se conocían de asignaturas cursadas en el grado, por lo que un repaso de las prácticas de asignaturas como *Sistemas Distribuidos e Internet* y *Repositorios de Información* bastaron para refrescar los conocimientos.

Documentar todo el proceso de creación de la aplicación web ha permitido poner en práctica y repasar todos los conocimientos aprendidos durante los años de carrera y crear un archivo con todas las especificaciones y explicaciones necesarias para retomar el proyecto en un futuro.

Concluyo diciendo que a pesar de que la planificación inicial no se ha conseguido cumplir, estoy contenta con el trabajo realizado ya que opino que he dado lo mejor de mí para el tiempo del que disponía. Soy consciente de que existen muchas mejoras posibles y uno de mis objetivos para el futuro es llevarlas a cabo junto con las ampliaciones propuestas en el siguiente apartado.



8.3 AMPLIACIONES

En este apartado se comentan algunas ampliaciones que se han pensado para realizar en un futuro.

8.3.1 Aplicación móvil

Uno de los principales motivos de la elección de tecnologías de desarrollo realizada fue poder convertir lo que actualmente es una aplicación web en una aplicación móvil de manera sencilla.

Dado que la aplicación está pensada para empleados de Renfe, convertir la misma en una aplicación móvil permitiría a los usuarios utilizarla desde su propio dispositivo móvil durante la jornada laboral. De esta manera podrían consultar las tareas asignadas para un día y notificar de incidencias en los trenes de una manera mucho más rápida y sencilla.

8.3.2 Centro de notificaciones

Actualmente, una vez que un empleado realiza una nueva solicitud o un administrador acepta/rechaza una solicitud existente, ninguno de los dos usuarios es notificado. Para comprobar el estado de las solicitudes el usuario debe acceder a la pantalla correspondiente de visualización de solicitudes.

Crear un centro de notificaciones para los usuarios registrados sería una buena idea para avisar a los mismos de resoluciones en las solicitudes que han realizado o de nuevas solicitudes pendientes.

8.3.3 Ampliar funcionalidad de los usuarios pasajeros

Otra de las ampliaciones que se pretenden realizar en un futuro es ampliar la funcionalidad de los usuarios pasajeros.

Debido al limitado tiempo para realizar el proyecto las funcionalidades disponibles para los usuarios pasajeros se acotan a consultar la localización de estaciones en el mapa y a consultar los horarios entre dos estaciones.

Se pretende que próximamente los pasajeros puedan consultar los horarios en tiempo real y puedan disponer de su propio perfil dentro de la aplicación. Alguno de los datos que almacenaría el usuario en la aplicación sería su tarjeta de viajes (en forma de código QR) para facilitar los desplazamientos.



8.4 REFERENCIAS BIBLIOGRÁFICAS

- [1] “(PDF) Documentos-modelo para Trabajos de Fin de Grado/Master de la Escuela de Informática de Oviedo.”
https://www.researchgate.net/publication/327882831_Documentos-modelo_para_Trabajos_de_Fin_de_GradoMaster_de_la_Escuela_de_Informatica_de_Oviedo (accessed May 29, 2021).
- [2] “Núcleo de Cercanías de Asturias - Renfe.”
<https://www.renfe.com/es/es/cercanias/cercanias-asturias> (accessed Nov. 17, 2022).
- [3] “Acerca | Node.js.” <https://nodejs.org/es/about/> (accessed May 29, 2021).
- [4] “JavaScript | MDN.” <https://developer.mozilla.org/es/docs/Web/JavaScript> (accessed Nov. 17, 2022).
- [5] “Spring Framework.” <https://spring.io/projects/spring-framework> (accessed Nov. 17, 2022).
- [6] “Diferencias entre Spring y Spring Boot - Java desde 0.” <https://javadesde0.com/diferencias-entre-spring-y-spring-boot/> (accessed May 29, 2021).
- [7] “TypeScript: JavaScript With Syntax For Types.” <https://www.typescriptlang.org/> (accessed Nov. 17, 2022).
- [8] “The most popular database for modern apps | MongoDB.” <https://www.mongodb.com/> (accessed May 29, 2021).
- [9] “MySQL.” <https://www.mysql.com/> (accessed May 31, 2021).
- [10] “Software de administración de proyectos | Microsoft Project.”
<https://www.microsoft.com/es-es/microsoft-365/project/project-management-software> (accessed Nov. 17, 2022).
- [11] “Estándar IEEE 830-1998 | Estándares de TI.”
<https://estandaresti.wordpress.com/2016/12/17/estandar-ieee-830-1998/> (accessed Nov. 17, 2022).
- [12] “WCAG 2 Overview | Web Accessibility Initiative (WAI) | W3C.”
<https://www.w3.org/WAI/standards-guidelines/wcag/> (accessed Nov. 17, 2022).
- [13] “Javadoc.” <https://docs.oracle.com/javase/8/docs/technotes/tools/windows/javadoc.html> (accessed Nov. 17, 2022).
- [14] “Java Code Conventions,” 1997.
- [15] “Java | Oracle.” <https://www.java.com/es/> (accessed Nov. 17, 2022).



- [16] “Lenguaje HTML5 - HTML en español - Lenguaje HTML.” <https://lenguajehtml.com/html/> (accessed Nov. 17, 2022).
- [17] “CSS | MDN.” <https://developer.mozilla.org/es/docs/Web/CSS> (accessed Nov. 17, 2022).
- [18] “Spring | Home.” <https://spring.io/> (accessed May 29, 2021).
- [19] “JUnit 5 .” <https://junit.org/junit5/docs/snapshot/user-guide/> (accessed Nov. 17, 2022).
- [20] “The IntelliJ IDEA Blog : IntelliJ IDEA – the Leading Java and Kotlin IDE, by JetBrains | The JetBrains Blog.” <https://blog.jetbrains.com/idea/> (accessed Nov. 17, 2022).
- [21] “IntelliJ IDEA: el IDE de Java eficaz y ergonómico de JetBrains.” <https://www.jetbrains.com/es-es/idea/> (accessed Jul. 02, 2021).
- [22] “MySQL :: MySQL Workbench.” <https://www.mysql.com/products/workbench/> (accessed Nov. 17, 2022).
- [23] “GitHub.” <https://github.com/> (accessed May 29, 2021).
- [24] “Firefox - Protege tu vida en línea con productos que priorizan la privacidad — Mozilla.” <https://www.mozilla.org/es-ES/firefox/> (accessed Nov. 17, 2022).
- [25] “Selenium.” <https://www.selenium.dev/> (accessed Nov. 17, 2022).
- [26] “Microsoft Word: software de procesamiento de texto | Microsoft 365.” <https://www.microsoft.com/es-es/microsoft-365/word?activetab=tabs%3afaqheaderregion3> (accessed Nov. 17, 2022).
- [27] “Software de hojas de cálculo Microsoft Excel | Microsoft 365.” <https://www.microsoft.com/es-es/microsoft-365/excel> (accessed Nov. 17, 2022).
- [28] “Software de presentación de diapositivas Microsoft PowerPoint | Microsoft 365.” <https://www.microsoft.com/es-es/microsoft-365/powerpoint> (accessed Nov. 17, 2022).
- [29] “Microsoft Forms | Encuestas, sondeos y cuestionarios.” <https://www.microsoft.com/es-es/microsoft-365/online-surveys-polls-quizzes> (accessed Nov. 17, 2022).
- [30] “Balsamiq. Rapid, Effective and Fun Wireframing Software | Balsamiq.” <https://balsamiq.com/> (accessed Jul. 02, 2021).
- [31] “PMBOK Guide | Project Management Institute.” https://www.pmi.org/pmbok-guide-standards/foundational/pmbok?sc_campaign=D750AAC10C2F4378CE6D51F8D987F49D (accessed Nov. 17, 2022).
- [32] B. W. Boehm, “Software Risk Management: Principles and Practices,” *IEEE Softw*, vol. 8, no. 1, pp. 32–41, 1991, doi: 10.1109/52.62930.

8.5 CONTENIDO ENTREGADO EN LOS ANEXOS

8.5.1 Contenidos

En la entrega del proyecto se incluyen el documento principal y un archivo comprimido con una serie de contenidos adicionales que se detallan a continuación.

8.5.1.1 Estructura del archivo entregado

Directorio	Contenido
./	Contiene el documento PDF y un fichero README.TXT explicando toda esta estructura
./anexos	Contiene los ficheros complementarios a la documentación y otros archivos de importancia para el proyecto.
./TFG_UO266047_BackEnd	Contiene un zip con la parte back-end del proyecto de desarrollo. Una vez sea descomprimido se podrá ver la estructura de la Tabla 116. Estructura de directorios de desarrollo - Back-end
./TFG_UO266047_FrontEnd	Contiene un zip con la parte front-end del proyecto de desarrollo. Una vez sea descomprimido se podrá ver la estructura de la Tabla 117.

Tabla 115. Estructura del archivo entregado

8.5.1.2 Estructura de directorios de desarrollo – Back-end

Directorio	Contenido
./	Contiene las carpetas y ficheros del proyecto. - <i>pom.xml</i> : Contiene las dependencias del proyecto
./doc	Contiene los ficheros generados con <i>Javadoc</i>
./lib	Contiene bibliotecas externas necesarias para compilar y distribuir, de las que depende este proyecto.
./src/main/java	Contiene los ficheros de código fuente agrupados por paquetes según la estructura del proyecto.

./src/main/resources	Contiene el archivo de configuración <i>application.properties</i>
./src/main/resources/test	Contiene el archivo ejecutable necesario para los test de integración
./src/test/java/com/uniovi/integracion	Contiene todos los ficheros para las pruebas de integración
./src/test/java/com/uniovi/unit	Contiene todos los ficheros para las pruebas unitarias
./src/test/java/com/uniovi/utills	Contiene un fichero complementario para las pruebas.

Tabla 116. Estructura de directorios de desarrollo - Back-end

8.5.1.3 Estructura de directorios de desarrollo – Front-end

Directorio	Contenido
./	Contiene las carpetas principales de la aplicación y archivos JSON de configuración.
./src	Contiene las carpetas principales de la aplicación y la página principal de la misma
./src/app	Contiene los ficheros de código fuente agrupados por paquetes según la estructura del proyecto
./src/assets	Contiene las imágenes del proyecto
./src/enviroments	Contiene archivos que permiten la conexión con el back-end.

Tabla 117. Estructura de directorios de desarrollo - Front-end

8.5.1.4 Contenidos de los anexos

Directorio	Contenido
./UO266047BackEnd.jar	Archivo jar del proyecto
./UO266047FrontEnd.zip	Contiene los archivos de ejecución del front-end
./riesgos_TFG_UO266047	Registro de riesgos
./presupuesto_TFG_UO266047	Presupuestos del proyecto
./scriptCarga.sql	Fichero de carga de datos iniciales en la BD.

Tabla 118. Contenidos de los anexos



8.6 GNU FREE DOCUMENTATION LICENSE

Version 1.3, 3 November 2008

Copyright © 2000, 2001, 2002, 2007, 2008 Free Software Foundation, Inc. <<https://fsf.org/>>

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

8.6.1.1 0. PREAMBLE

The purpose of this License is to make a manual, textbook, or other functional and useful document "free" in the sense of freedom: to assure everyone the effective freedom to copy and redistribute it, with or without modifying it, either commercially or noncommercially. Secondly, this License preserves for the author and publisher a way to get credit for their work, while not being considered responsible for modifications made by others.

This License is a kind of "copyleft", which means that derivative works of the document must themselves be free in the same sense. It complements the GNU General Public License, which is a copyleft license designed for free software.

We have designed this License in order to use it for manuals for free software, because free software needs free documentation: a free program should come with manuals providing the same freedoms that the software does. But this License is not limited to software manuals; it can be used for any textual work, regardless of subject matter or whether it is published as a printed book. We recommend this License principally for works whose purpose is instruction or reference.

8.6.1.2 1. APPLICABILITY AND DEFINITIONS

This License applies to any manual or other work, in any medium, that contains a notice placed by the copyright holder saying it can be distributed under the terms of this License. Such a notice grants a world-wide, royalty-free license, unlimited in duration, to use that work under the conditions stated herein. The "Document", below, refers to any such manual or work. Any member of the public is a licensee, and is addressed as "you". You accept the license if you copy, modify or distribute the work in a way requiring permission under copyright law.

A "Modified Version" of the Document means any work containing the Document or a portion of it, either copied verbatim, or with modifications and/or translated into another language.

A "Secondary Section" is a named appendix or a front-matter section of the Document that deals exclusively with the relationship of the publishers or authors of the Document to the Document's overall subject (or to related matters) and contains nothing that could fall directly within that overall subject. (Thus, if the Document is in part a textbook of mathematics, a Secondary Section may not



explain any mathematics.) The relationship could be a matter of historical connection with the subject or with related matters, or of legal, commercial, philosophical, ethical or political position regarding them.

The "Invariant Sections" are certain Secondary Sections whose titles are designated, as being those of Invariant Sections, in the notice that says that the Document is released under this License. If a section does not fit the above definition of Secondary then it is not allowed to be designated as Invariant. The Document may contain zero Invariant Sections. If the Document does not identify any Invariant Sections then there are none.

The "Cover Texts" are certain short passages of text that are listed, as Front-Cover Texts or Back-Cover Texts, in the notice that says that the Document is released under this License. A Front-Cover Text may be at most 5 words, and a Back-Cover Text may be at most 25 words.

A "Transparent" copy of the Document means a machine-readable copy, represented in a format whose specification is available to the general public, that is suitable for revising the document straightforwardly with generic text editors or (for images composed of pixels) generic paint programs or (for drawings) some widely available drawing editor, and that is suitable for input to text formatters or for automatic translation to a variety of formats suitable for input to text formatters. A copy made in an otherwise Transparent file format whose markup, or absence of markup, has been arranged to thwart or discourage subsequent modification by readers is not Transparent. An image format is not Transparent if used for any substantial amount of text. A copy that is not "Transparent" is called "Opaque".

Examples of suitable formats for Transparent copies include plain ASCII without markup, Texinfo input format, LaTeX input format, SGML or XML using a publicly available DTD, and standard-conforming simple HTML, PostScript or PDF designed for human modification. Examples of transparent image formats include PNG, XCF and JPG. Opaque formats include proprietary formats that can be read and edited only by proprietary word processors, SGML or XML for which the DTD and/or processing tools are not generally available, and the machine-generated HTML, PostScript or PDF produced by some word processors for output purposes only.

The "Title Page" means, for a printed book, the title page itself, plus such following pages as are needed to hold, legibly, the material this License requires to appear in the title page. For works in formats which do not have any title page as such, "Title Page" means the text near the most prominent appearance of the work's title, preceding the beginning of the body of the text.

The "publisher" means any person or entity that distributes copies of the Document to the public.

A section "Entitled XYZ" means a named subunit of the Document whose title either is precisely XYZ or contains XYZ in parentheses following text that translates XYZ in another language. (Here XYZ stands for a specific section name mentioned below, such as "Acknowledgements", "Dedications", "Endorsements", or "History".) To "Preserve the Title" of such a section when you modify the Document means that it remains a section "Entitled XYZ" according to this definition.



The Document may include Warranty Disclaimers next to the notice which states that this License applies to the Document. These Warranty Disclaimers are considered to be included by reference in this License, but only as regards disclaiming warranties: any other implication that these Warranty Disclaimers may have is void and has no effect on the meaning of this License.

8.6.1.3 2. VERBATIM COPYING

You may copy and distribute the Document in any medium, either commercially or noncommercially, provided that this License, the copyright notices, and the license notice saying this License applies to the Document are reproduced in all copies, and that you add no other conditions whatsoever to those of this License. You may not use technical measures to obstruct or control the reading or further copying of the copies you make or distribute. However, you may accept compensation in exchange for copies. If you distribute a large enough number of copies you must also follow the conditions in section 3.

You may also lend copies, under the same conditions stated above, and you may publicly display copies.

8.6.1.4 3. COPYING IN QUANTITY

If you publish printed copies (or copies in media that commonly have printed covers) of the Document, numbering more than 100, and the Document's license notice requires Cover Texts, you must enclose the copies in covers that carry, clearly and legibly, all these Cover Texts: Front-Cover Texts on the front cover, and Back-Cover Texts on the back cover. Both covers must also clearly and legibly identify you as the publisher of these copies. The front cover must present the full title with all words of the title equally prominent and visible. You may add other material on the covers in addition. Copying with changes limited to the covers, as long as they preserve the title of the Document and satisfy these conditions, can be treated as verbatim copying in other respects.

If the required texts for either cover are too voluminous to fit legibly, you should put the first ones listed (as many as fit reasonably) on the actual cover, and continue the rest onto adjacent pages.

If you publish or distribute Opaque copies of the Document numbering more than 100, you must either include a machine-readable Transparent copy along with each Opaque copy, or state in or with each Opaque copy a computer-network location from which the general network-using public has access to download using public-standard network protocols a complete Transparent copy of the Document, free of added material. If you use the latter option, you must take reasonably prudent steps, when you begin distribution of Opaque copies in quantity, to ensure that this Transparent copy will remain thus accessible at the stated location until at least one year after the last time you distribute an Opaque copy (directly or through your agents or retailers) of that edition to the public.



It is requested, but not required, that you contact the authors of the Document well before redistributing any large number of copies, to give them a chance to provide you with an updated version of the Document.

8.6.1.5 4. MODIFICATIONS

You may copy and distribute a Modified Version of the Document under the conditions of sections 2 and 3 above, provided that you release the Modified Version under precisely this License, with the Modified Version filling the role of the Document, thus licensing distribution and modification of the Modified Version to whoever possesses a copy of it. In addition, you must do these things in the Modified Version:

- A. Use in the Title Page (and on the covers, if any) a title distinct from that of the Document, and from those of previous versions (which should, if there were any, be listed in the History section of the Document). You may use the same title as a previous version if the original publisher of that version gives permission.*
- B. List on the Title Page, as authors, one or more persons or entities responsible for authorship of the modifications in the Modified Version, together with at least five of the principal authors of the Document (all of its principal authors, if it has fewer than five), unless they release you from this requirement.*
- C. State on the Title page the name of the publisher of the Modified Version, as the publisher.*
- D. Preserve all the copyright notices of the Document.*
- E. Add an appropriate copyright notice for your modifications adjacent to the other copyright notices.*
- F. Include, immediately after the copyright notices, a license notice giving the public permission to use the Modified Version under the terms of this License, in the form shown in the Addendum below.*
- G. Preserve in that license notice the full lists of Invariant Sections and required Cover Texts given in the Document's license notice.*
- H. Include an unaltered copy of this License.*
- I. Preserve the section Entitled "History", Preserve its Title, and add to it an item stating at least the title, year, new authors, and publisher of the Modified Version as given on the Title Page. If there is no section Entitled "History" in the Document, create one stating the title, year, authors, and publisher of the Document as given on its Title Page, then add an item describing the Modified Version as stated in the previous sentence.*
- J. Preserve the network location, if any, given in the Document for public access to a Transparent copy of the Document, and likewise the network locations given in the Document for previous versions it was based on. These may be placed in the "History" section. You may omit a network location for a work that was published at least four years before the Document itself, or if the original publisher of the version it refers to gives permission.*



- K. For any section Entitled "Acknowledgements" or "Dedications", Preserve the Title of the section, and preserve in the section all the substance and tone of each of the contributor acknowledgements and/or dedications given therein.*
- L. Preserve all the Invariant Sections of the Document, unaltered in their text and in their titles. Section numbers or the equivalent are not considered part of the section titles.*
- M. Delete any section Entitled "Endorsements". Such a section may not be included in the Modified Version.*
- N. Do not retitle any existing section to be Entitled "Endorsements" or to conflict in title with any Invariant Section.*
- O. Preserve any Warranty Disclaimers.*

If the Modified Version includes new front-matter sections or appendices that qualify as Secondary Sections and contain no material copied from the Document, you may at your option designate some or all of these sections as invariant. To do this, add their titles to the list of Invariant Sections in the Modified Version's license notice. These titles must be distinct from any other section titles.

You may add a section Entitled "Endorsements", provided it contains nothing but endorsements of your Modified Version by various parties—for example, statements of peer review or that the text has been approved by an organization as the authoritative definition of a standard.

You may add a passage of up to five words as a Front-Cover Text, and a passage of up to 25 words as a Back-Cover Text, to the end of the list of Cover Texts in the Modified Version. Only one passage of Front-Cover Text and one of Back-Cover Text may be added by (or through arrangements made by) any one entity. If the Document already includes a cover text for the same cover, previously added by you or by arrangement made by the same entity you are acting on behalf of, you may not add another; but you may replace the old one, on explicit permission from the previous publisher that added the old one.

The author(s) and publisher(s) of the Document do not by this License give permission to use their names for publicity for or to assert or imply endorsement of any Modified Version.

8.6.1.6 5. COMBINING DOCUMENTS

You may combine the Document with other documents released under this License, under the terms defined in section 4 above for modified versions, provided that you include in the combination all of the Invariant Sections of all of the original documents, unmodified, and list them all as Invariant Sections of your combined work in its license notice, and that you preserve all their Warranty Disclaimers.

The combined work need only contain one copy of this License, and multiple identical Invariant Sections may be replaced with a single copy. If there are multiple Invariant Sections with the same name but different contents, make the title of each such section unique by adding at the end of it, in parentheses, the name of the original author or publisher of that section if known, or else a unique



number. Make the same adjustment to the section titles in the list of Invariant Sections in the license notice of the combined work.

In the combination, you must combine any sections Entitled "History" in the various original documents, forming one section Entitled "History"; likewise combine any sections Entitled "Acknowledgements", and any sections Entitled "Dedications". You must delete all sections Entitled "Endorsements".

8.6.1.7 6. COLLECTIONS OF DOCUMENTS

You may make a collection consisting of the Document and other documents released under this License, and replace the individual copies of this License in the various documents with a single copy that is included in the collection, provided that you follow the rules of this License for verbatim copying of each of the documents in all other respects.

You may extract a single document from such a collection, and distribute it individually under this License, provided you insert a copy of this License into the extracted document, and follow this License in all other respects regarding verbatim copying of that document.

8.6.1.8 7. AGGREGATION WITH INDEPENDENT WORKS

A compilation of the Document or its derivatives with other separate and independent documents or works, in or on a volume of a storage or distribution medium, is called an "aggregate" if the copyright resulting from the compilation is not used to limit the legal rights of the compilation's users beyond what the individual works permit. When the Document is included in an aggregate, this License does not apply to the other works in the aggregate which are not themselves derivative works of the Document.

If the Cover Text requirement of section 3 is applicable to these copies of the Document, then if the Document is less than one half of the entire aggregate, the Document's Cover Texts may be placed on covers that bracket the Document within the aggregate, or the electronic equivalent of covers if the Document is in electronic form. Otherwise they must appear on printed covers that bracket the whole aggregate.

8.6.1.9 8. TRANSLATION

Translation is considered a kind of modification, so you may distribute translations of the Document under the terms of section 4. Replacing Invariant Sections with translations requires special permission from their copyright holders, but you may include translations of some or all Invariant Sections in addition to the original versions of these Invariant Sections. You may include a translation of this License, and all the license notices in the Document, and any Warranty Disclaimers, provided that you also include the original English version of this License and the original versions of those



notices and disclaimers. In case of a disagreement between the translation and the original version of this License or a notice or disclaimer, the original version will prevail.

If a section in the Document is Entitled "Acknowledgements", "Dedications", or "History", the requirement (section 4) to Preserve its Title (section 1) will typically require changing the actual title.

8.6.1.10 9. TERMINATION

You may not copy, modify, sublicense, or distribute the Document except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, or distribute it is void, and will automatically terminate your rights under this License.

However, if you cease all violation of this License, then your license from a particular copyright holder is reinstated (a) provisionally, unless and until the copyright holder explicitly and finally terminates your license, and (b) permanently, if the copyright holder fails to notify you of the violation by some reasonable means prior to 60 days after the cessation.

Moreover, your license from a particular copyright holder is reinstated permanently if the copyright holder notifies you of the violation by some reasonable means, this is the first time you have received notice of violation of this License (for any work) from that copyright holder, and you cure the violation prior to 30 days after your receipt of the notice.

Termination of your rights under this section does not terminate the licenses of parties who have received copies or rights from you under this License. If your rights have been terminated and not permanently reinstated, receipt of a copy of some or all of the same material does not give you any rights to use it.

8.6.1.11 10. FUTURE REVISIONS OF THIS LICENSE

The Free Software Foundation may publish new, revised versions of the GNU Free Documentation License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns. See <https://www.gnu.org/licenses/>.

Each version of the License is given a distinguishing version number. If the Document specifies that a particular numbered version of this License "or any later version" applies to it, you have the option of following the terms and conditions either of that specified version or of any later version that has been published (not as a draft) by the Free Software Foundation. If the Document does not specify a version number of this License, you may choose any version ever published (not as a draft) by the Free Software Foundation. If the Document specifies that a proxy can decide which future versions of this License can be used, that proxy's public statement of acceptance of a version permanently authorizes you to choose that version for the Document.



8.6.1.12 11. RELICENSING

"Massive Multiauthor Collaboration Site" (or "MMC Site") means any World Wide Web server that publishes copyrightable works and also provides prominent facilities for anybody to edit those works. A public wiki that anybody can edit is an example of such a server. A "Massive Multiauthor Collaboration" (or "MMC") contained in the site means any set of copyrightable works thus published on the MMC site.

"CC-BY-SA" means the Creative Commons Attribution-Share Alike 3.0 license published by Creative Commons Corporation, a not-for-profit corporation with a principal place of business in San Francisco, California, as well as future copyleft versions of that license published by that same organization.

"Incorporate" means to publish or republish a Document, in whole or in part, as part of another Document.

An MMC is "eligible for relicensing" if it is licensed under this License, and if all works that were first published under this License somewhere other than this MMC, and subsequently incorporated in whole or in part into the MMC, (1) had no cover texts or invariant sections, and (2) were thus incorporated prior to November 1, 2008.

The operator of an MMC Site may republish an MMC contained in the site under CC-BY-SA on the same site at any time before August 1, 2009, provided the MMC is eligible for relicensing.

8.6.2 ADDENDUM: How to use this License for your documents

To use this License in a document you have written, include a copy of the License in the document and put the following copyright and license notices just after the title page:

```
Copyright (C) YEAR YOUR NAME.  
Permission is granted to copy, distribute and/or modify this document  
under the terms of the GNU Free Documentation License, Version 1.3  
or any later version published by the Free Software Foundation;  
with no Invariant Sections, no Front-Cover Texts, and no Back-Cover Texts.  
A copy of the license is included in the section entitled "GNU  
Free Documentation License".
```

If you have Invariant Sections, Front-Cover Texts and Back-Cover Texts, replace the "with ... Texts." line with this:

```
with the Invariant Sections being LIST THEIR TITLES, with the  
Front-Cover Texts being LIST, and with the Back-Cover Texts being LIST.
```

If you have Invariant Sections without Cover Texts, or some other combination of the three, merge those two alternatives to suit the situation.



If your document contains nontrivial examples of program code, we recommend releasing these examples in parallel under your choice of free software license, such as the GNU General Public License, to permit their use in free software.