UNIVERSITY OF OVIEDO

SCHOOL OF COMPUTER ENGINEERING

DEGREE PROJECT
BACHELOR OF SOFTWARE ENGINEERING

"Collaborative Cloud-based Shape Expressions Editor"

Seen and approved

ADVISOR:
José Emilio Labra Gayo
([redacted]@uniovi.es)

AUTHOR:
Iván Álvarez López
([redacted]@uniovi.es)

Declaration of Honesty in Academic Work

I declare that I have read and checked that all parts of the piece of work here submitted are original, except for source material explicitly acknowledged.

I declare that the piece of work, or a part of the piece of work has not been submitted for more than one purpose (e.g., to satisfy the requirements in two different courses) without declaration.

I declare that the submitted soft copy with details listed in the submission is identical to the hard copy, if any, which has been submitted. I also acknowledge that I am aware of the University's policy and regulations on honesty in academic work, and of the disciplinary guidelines and procedures applicable to breaches of such policy and regulations, as contained in the University website: https://www.innova.uniovi.es/ceonline/propiedadintelectual.

Signed: Iván Álvarez López

Copyright disclaimer

ShareYASHE. Collaborative Cloud-based Shape Expressions Editor.

Acknowledgements

I would like to extend my sincere appreciation to the following individuals, who have played a pivotal role in the completion of my degree project:

First and foremost, I am deeply grateful to my advisor, J. E. Labra, for granting me the opportunity to undertake this captivating piece of work. His guidance, expertise, and support have been invaluable throughout this journey.

I am indebted to my parents, Isidro, and Pilar, for their love, constant encouragement, and assistance. Without their selfless sacrifices and unconditional belief in my abilities, this project would not have come to fruition. Their support has been the cornerstone of my success.

I would like to express my heartfelt appreciation to my friend Miguel, whose consistent friendship and support have been an integral part of my life since our time together in middle school. His belief in my abilities and continuous encouragement have played a great role in my progress.

To my cousin Raúl, I am grateful for the inspiration you have provided since our childhood. Your enthusiasm, creativity, and courageousness have motivated me to surpass my limitations and strive for excellence in all areas of my work.

To my dear friend Lucía, I am immensely grateful for the profound impact you have had on my life, particularly during the final years of my degree. Your friendship and understanding have been a beacon of light during the darkest moments. Your words of wisdom and guidance have provided clarity and perspective, enabling me to navigate through challenges with renewed strength and determination.

Lastly, I want to pay a heartfelt tribute to my grandfathers, with a special mention of my beloved late grandfather, José. His belief in my abilities and encouragement to pursue higher education have left an indelible mark on my academic journey. His words have fuelled my determination and instilled in me the confidence to overcome challenges. Although he is no longer physically with us, his enduring legacy of inspiration continues to guide me, serving as a constant reminder of the strength and resilience that lies within me. I am forever grateful for his presence in my life and the profound impact he has had on shaping the person I have become.

To all these exceptional individuals, your unwavering support, guidance, and belief in my abilities have formed the foundation upon which I have built my academic achievements. I am profoundly grateful for your presence in my life and the profound impact you have had on my personal and academic growth.

From the depths of my heart, thank you all.

Abstract

It will be developed a System which allows ShEx (Shape Expressions) code editing in a collaborative way. A person editing a file will be able to see changes made by another person in real time. This application will be cloud-based, that is, it will be deployed as a Website. Every user will have ShEx files on their own, and these files could be shared with other users for their collaborative editing.

As a Software developing project, this one will be focused on the analysis, design, and implementation of Software Systems that will solve the problem previously described.

Keywords

ShEx, collaborative, editor, ShareYASHE, YASHE, Web Semantics, HTML, CSS, JavaScript, WebSockets, Software, testing, development, engineering, usability, adaptability, accessibility, design, architecture.

Table of contents

Table of figures

1 Summary

1.1 Introduction

Shape Expressions (ShEx)[ShEx13] is a formal language and framework used for specifying and validating the structural constraints of data, particularly in the context of the Semantic Web and Linked Data. It provides a concise and expressive syntax for defining the shapes or structures that data should conform to. These shapes can be applied to various data models, including RDF (Resource Description Framework) graphs, JSON documents, XML, and more.

The fundamental concept behind Shape Expressions is that of a shape, which represents a pattern or template for specifying the structure and constraints that data should adhere to. A shape consists of a collection of constraints and rules that define the allowed properties, their cardinality, value types, and interrelationships within the data.

ShareYASHE is an innovative project aimed at providing a user-friendly and efficient platform for creating, editing, and collaborating on Shape Expressions specifications in a distributed and collaborative manner. This web-based editor leverages the power of the cloud to enable teams of users to work together seamlessly, regardless of their geographical locations. The term “ShareYASHE” is a reference to ShareLaTeX (a collaborative cloud-based LaTeX editor), using YASHE as the base term instead of LaTeX.

Shape Expressions, being a powerful language for specifying structural constraints, plays a crucial role in the Semantic Web and Linked Data domains. However, creating and managing Shape Expressions specifications can be a challenging task, especially when multiple stakeholders are involved. Traditional text-based editors lack the collaborative features required for efficient teamwork, often leading to coordination issues, version conflicts, and time-consuming manual merging of changes.

This collaborative cloud-based ShEx editor addresses these challenges by providing a rich set of features that empower teams to work collaboratively and efficiently. The editor's web-based nature allows users to access and work on Shape Expressions specifications from anywhere, using a standard web browser. This eliminates the need for installing specialized software on individual machines and facilitates easy onboarding for team members.

Imagine a seamless collaboration experience where you and your team members can work on Shape Expressions specifications together, regardless of your geographical locations. With real-time collaboration, you can make edits, provide feedback, and engage in discussions in a synchronized environment. Say goodbye to tedious email chains and manual merging of changes. This editor empowers you to work efficiently, ensuring that everyone is on the same page and accelerating the specification development process.

1.2 Purpose

The purpose of ShareYASHE is to provide a powerful and user-friendly platform for creating, editing, and collaborating on Shape Expressions (ShEx) specifications. The project aims to address the challenges faced by teams working on ShEx by offering a comprehensive solution that promotes seamless collaboration, improves productivity, and simplifies the overall process of managing ShEx specifications.

The primary purpose of the project is to facilitate effective collaboration among team members. The editor enables real-time collaboration, allowing multiple users to work simultaneously on the same ShEx specification. This eliminates the need for manual coordination and the risk of version conflicts. By providing a synchronized environment for editing and discussion, the project enhances teamwork, fosters communication, and accelerates the development of ShEx specifications.

Another key purpose of the project is to enhance productivity. The editor offers a user-friendly interface with features such as syntax highlighting, auto-completion, and error checking. These features assist users in writing valid and error-free ShEx specifications, reducing the time spent on manual troubleshooting. By streamlining the specification creation process and minimizing errors, the project empowers users to focus on shaping their data efficiently.

The project also aims to simplify access and onboarding. As a cloud-based editor, it eliminates the need for users to install specialized software on their machines. Users can access and work on ShEx specifications from anywhere, using a standard web browser. This ease of access ensures that team members can collaborate seamlessly, regardless of their geographical locations. Furthermore, the user-friendly interface reduces the learning curve, making it easier for new users to onboard and contribute to the project quickly.

1.3 Objective

The objective of creating a collaborative cloud-based Shape Expressions (ShEx) editor is to develop a robust and feature-rich platform that enables teams to collaborate seamlessly on ShEx specifications in a distributed and efficient manner. The primary goal is to address the limitations of traditional text-based editors and provide a comprehensive solution that promotes real-time collaboration, enhances productivity, and simplifies the process of working with ShEx.

One of the main objectives of the project is to facilitate effective collaboration among team members. By leveraging cloud-based technologies, the editor allows multiple users to work on the same ShEx specification simultaneously. Real-time collaboration ensures that changes made by one user are instantly reflected to others, enabling a synchronized environment for teamwork. This objective aims to eliminate coordination issues, version conflicts, and delays in the collaboration process.

1.4 Scope

The scope of the ShareYASHE project encompasses the development of a comprehensive platform that allows users to create, edit, and collaborate on Shape Expressions (ShEx) specifications. The project aims to provide a robust set of features and functionalities that enable seamless collaboration, enhance productivity, and simplify the overall process of working with ShEx.

The scope of the project also covers considerations such as performance, scalability, and security. The editor should be designed to handle a considerable number of concurrent users and ensure data privacy and integrity. It should also support efficient search and navigation within specifications, as well as offer export/import functionalities to facilitate interoperability with other tools and platforms.

It is important to note that the scope of the project does not extend to the actual specification of the ShEx language itself. The focus is on providing a collaborative editing environment for working with ShEx specifications rather than defining the language's syntax or semantics.

The project's scope includes the following key components.

1.4.1 Web-based editor

The project focuses on developing a web-based editor that allows users to access and work on ShEx specifications through a standard web browser. The editor will provide an intuitive and user-friendly interface with features such as syntax highlighting, auto-completion, and error checking to facilitate efficient specification creation and editing.

1.4.2 Real-time collaboration

The project aims to enable real-time collaboration, allowing multiple users to work on the same ShEx specification simultaneously. This feature ensures that changes made by one user are instantly reflected to others, promoting synchronized teamwork and reducing coordination issues and version conflicts.

2 Project planning

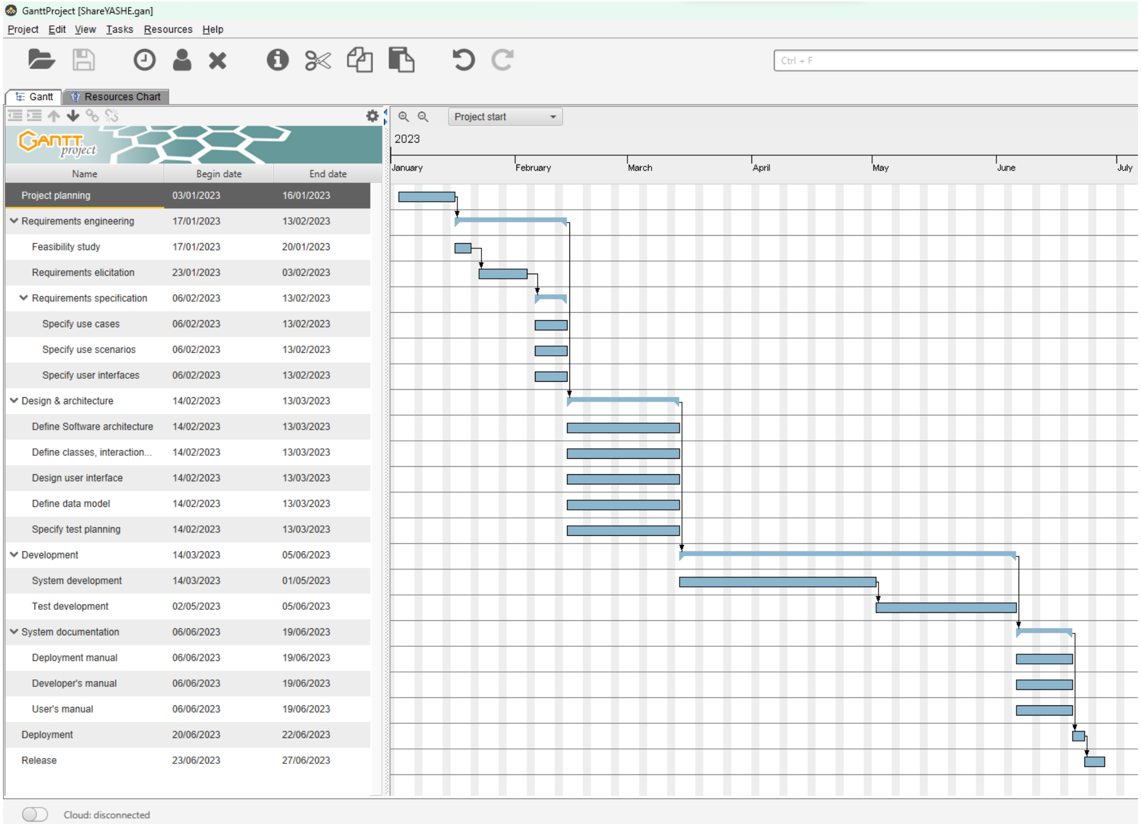
2.1 Initial planning

The project is expected to take place between 1st January 2023 and 1st July 2023. In this chapter we will see the initial planning for the project. The project objectives will be defined, and a comprehensive plan for the project execution will be established. The project planning phase is indeed part of the project planning.

There will only be one developer involved in this project, who will adjust himself to a full-time flexible schedule to carry out these labours: 8h per day, from Monday to Friday; this makes 40 hours per week.

The following Gantt chart presents a visual representation of the project schedule that shows the tasks, milestones, and dependencies over time. It provides a graphical view of the project's timeline, intended to allow the understanding of the project's progress, track tasks, and manage resources effectively.

2.1.1 Gantt chart



Name	Begin date	End date
Project planning	03/01/2023	16/01/2023
Requirements engineering	17/01/2023	13/02/2023
Feasibility study	17/01/2023	20/01/2023
Requirements elicitation	23/01/2023	03/02/2023
Requirements specification	06/02/2023	13/02/2023
Specify use cases	06/02/2023	13/02/2023
Specify use scenarios	06/02/2023	13/02/2023
Specify user interfaces	06/02/2023	13/02/2023
Design & architecture	14/02/2023	13/03/2023
Define Software architecture	14/02/2023	13/03/2023
Define classes, interaction...	14/02/2023	13/03/2023
Design user interface	14/02/2023	13/03/2023
Define data model	14/02/2023	13/03/2023
Specify test planning	14/02/2023	13/03/2023
Development	14/03/2023	05/06/2023
System development	14/03/2023	01/05/2023
Test development	02/05/2023	05/06/2023
System documentation	06/06/2023	19/06/2023
Deployment manual	06/06/2023	19/06/2023
Developer's manual	06/06/2023	19/06/2023
User's manual	06/06/2023	19/06/2023
Deployment	20/06/2023	22/06/2023
Release	23/06/2023	27/06/2023

Figure 1. Gantt chart of the project.

2.1.2 List of project tasks

The project is initially planned to have the following tasks.

1. Project planning.
2. Requirements engineering.
    a. Requirements elicitation.
    b. Requirements analysis and specification.
        i. Specify use cases.
        ii. Specify use scenarios.
3. Design and architecture.
    a. Define Software architecture.
    b. Define classes, interactions, and states.
    c. Design user interface.
    d. Define data model.
    e. Specify test planning.
4. Development.
    a. System development.
    b. Test development.
5. System documentation.
    a. Deployment manual.
    b. Developer’s manual.
    c. User’s manual.
6. Deployment.
7. Release.

These tasks may vary from those at the end of the project, due to this being an initial planning.

2.2 Initial budget

The initial budget section of this document provides a comprehensive overview of the estimated costs associated with the execution of the project. It presents a breakdown of various cost categories, allocations, and total budget projections.

Even though some resources, such as a personal computer, are already given to the developer, it must be written as if the project started from scratch.

The budget is organized into specific cost components, including personnel costs, hardware and software expenses, external services, communication and travel expenses, licensing and intellectual property fees, marketing and promotion costs, contingency, and miscellaneous expenses. Detailed cost estimates are provided for each category, outlining the assumptions and calculations used in the estimation process.

By presenting this initial budget, we ensure a clear understanding of the financial resources required to successfully execute the project while allowing for effective cost management and control.

Item	Concept	Quantity	Amortization	Price per unit	Total
1	*Software resources*				
1.1	Microsoft Windows 11	1	5%	150€	157.5€
1.2	Microsoft Office 365	1	7%	150€	160.5€
2	*Hardware resources*				
2.1	Server on the cloud	1	40%	300 €/month	2520€
2.2	Personal computer	2	20%	1000€	1200€
2.3	Tools: Internet, electricity, paper, pencils…	-	40%	300€	420€
3	*Human resources*				
3.1	Software developer	1	70%	1200 €/month	6120€
Subtotal					10578€
Beneficio (7%)					740.46€
IVA (21%)					2221.38€
TOTAL					13539.84€

Figure 2. Initial budget of the project.

The total cost of this project is **13 539.84€**.

3 Software requirements engineering

The Software requirements engineering process is a crucial phase in any Software development project, which involves gathering, analysing, documenting, and managing the requirements for a Software System[IanS11].

For this project, the three following phases were considered necessary.

- Requirements elicitation. This involves understanding and defining the needs of stakeholders, such as clients, end-users, and domain experts.
- Requirements analysis. The collected requirements are analysed to ensure clarity, consistency, and completeness. Conflicting or ambiguous requirements are resolved, and dependencies between different requirements are identified.
- Requirements specification. The requirements are documented in a formal and structured manner.

3.1 Software requirements elicitation

3.1.1 Identified stakeholders and other sources of information

The requirements elicitation phase requires to identify and involve all relevant stakeholders and any other source of information. The identified stakeholders and sources of information are listed below.

- The advisor. This project is a request made directly by the degree project advisor, who intends to use it as a tool for the labours of the WESO research group. He is both the advisor of this degree project, and the product owner.
- The WESO research group. The WESO research group works with Shape Expressions, and already uses similar tools to the intended product of this project. YASHE, the library which will work a supporting Software for the product of this project, was developed by the WESO group.
- Shape Expressions reference. The Shape Expressions language has its documentation on the W3C website: W3C - ShEx.
- Kevin Jahns, creator of the Yjs Project. He is the creator of the supporting Software module which will be used for implementing the collaborativity of the ShareYASHE editor instances.
- Similar products, such as Overleaf.

3.1.2 Elicitation technique: interviews

Due to the size and complexity of the project, the availability of stakeholders, and the nature of the requirements, I reckon that only one technique of elicitation will be enough: interviews.

Interviews are one of the most common and effective techniques for requirements elicitation. One-on-one discussions are held with stakeholders to understand their needs, expectations, and constraints. Interviews allow for a deeper exploration of stakeholders' perspectives and enable the elicitor to ask follow-up questions and clarify any ambiguities.

The stakeholders' requirements will be elicited from interviews with the product owner, who also is the advisor of this degree project.

3.2 Software requirements analysis

The analysis of the requirements of this System has resulted straightforward, as its requirements were planned from the beginning.

3.2.1 System actors

The users of ShareYASHE may have two roles, in base of its authorisation, and of its ownership of a ShEx document.

3.2.1.1 Authorised user versus non-authorised user

A user will be authorised if he has successfully made the registration process once, and the login process once in the session. Otherwise, he will be a non-authorised user.

3.2.1.2 Owner user versus non-owner user

A user will be an owner of a given ShEx document if he has created the ShEx document, or if he has been added as an owner of a ShEx document created by another owner user of the same ShEx document. He will be able to leave the ownership of the ShEx document.

3.3 System requirements specification

The information provided by the requirements elicitation and analysis phases is specified as follows to ensure clarity, consistency, and completeness.

ID	Requirement description
R.1.	The System will provide a collaborative cloud-based ShEx editor.
R.1.1.	The System will provide **user management**.
R.1.1.1.	The System will provide users with a **registration** in which the following fields will be requested.
R.1.1.1.1.	User's nickname.
R.1.1.1.1.1.	This field will be mandatory and longer than 3 characters.
R.1.1.1.1.2.	This field will be unique for each user.
R.1.1.1.2.	User's email address.
R.1.1.1.2.1.	This field will be mandatory.
R.1.1.1.2.2.	This field will be unique for each user.
R.1.1.1.2.3.	This field will follow the RFC 5322 structure.
R.1.1.1.3.	User's password.
R.1.1.1.3.1.	This field will be mandatory and longer than 3 characters.
R.1.1.2.	The System will provide users with a **login** in which the following fields will be requested. These fields are mandatory and must match with those given in the registration (see R.1.1.1.).
R.1.1.2.1.	User's nickname.
R.1.1.2.2.	User's password.
R.1.1.3.	The System will provide users with a **logout**.
R.1.1.3.1.	This will finish the login status, if there is, specified in R.1.1.2.
R.1.1.4.	The System will provide users with an **unregister**.
R.1.1.4.1.	This will finish the register status, if there is, specified in R.1.1.1.
R.1.1.4.2.	This will remove the user's ownership of all of his ShEx docs (see R.1.2.1.4.).

Figure 3. System requirements specification (1 of 2).

ID	Requirement description
R.1.2.	The System will provide users **ShEx docs management**.
R.1.2. 1.	The System will provide users with a **ShEx doc creation**.
R.1.2. 1.1.	The ShEx doc's title will be mandatorily requested.
R.1.2. 1.2.	A ShEx doc's unique ID will be generated.
R.1.2. 1.3.	Only registered (see R.1.1.1.) and logged in (see R.1.1.2.) users will be permitted by the System to do this.
R.1.2. 1.4.	The user will be owner of this ShEx doc.
R.1.2. 2.	The System will provide users with a **ShEx doc owned list**.
R.1.2. 2.1.	The list will show each ShEx doc's title.
R.1.2. 2.2.	Each entry of the list will grant access to the content edition of its ShEx doc (see R.1.2.3.).
R.1.2. 2.3.	Only registered (see R.1.1.1.) and logged in (see R.1.1.2.) users will have this.
R.1.2. 3.	The System will provide users with a **ShEx doc content editing**.
R.1.2. 3.1.	The content of the ShEx doc will be able to be edited by any user.
R.1.2. 3.2.	This editing will be shown to any user editing this content in real time.
R.1.2. 3.3.	The System will provide a share option. With this option the user will be able to invite other users to the content editing section of this ShEx doc.
R.1.2. 4.	The System will provide users with the capability to **add an owner to the ShEx doc**.
R.1.2. 4.1.	The user adding a new owner must be owner of the ShEx doc.
R.1.2. 4.2.	The user added as a new owner must not already own the ShEx doc.
R.1.2. 4.3.	The new owner will be owner of the ShEx doc (see R.1.2.1.4.), besides the previous owners.
R.1.2. 5.	The System will provide users with the capability to **leave the ownership of the ShEx doc**.
R.1.2. 5.1.	The user leaving the ownership must be owner of the ShEx doc.
R.1.2. 5.2.	The new owner will no longer be owner of the ShEx doc (see R.1.2.1.4.). This does not apply to the previous owners.

Figure 4. System requirements specification (2 of 2).

3.4 Use cases and scenarios

3.4.1 Use case 1. Register

Use case 1. Register	
Requirements	R.1.1.1.
Preconditions	-
Actors	Non-authorised users.
Description	A non- authorised user will create an account.
Sequence of steps	1. The user will open the ShareYASHE application. 2. The user will click the "Register" button. 3. The user will fill the formulary with a random username and a password. 4. The user will submit the formulary.
Postconditions	-
Exceptions	If the username is already taken, the user may repeat the process, using another username.

Figure 5. Use case of register.

3.4.2 Use case 2. Login

Use case 2. Login	
Requirements	R.1.1.2.
Preconditions	The user must have made the registration process (see use case 1) successfully.
Actors	Non-authorised users.
Description	A non-authorised user will login into an account.
Sequence of steps	1. The user will open the ShareYASHE application. 2. The user will click the "Log in" button. 3. The user will fill the formulary with the username and the password that he used in the registration process. 4. The user will submit the formulary
Postconditions	-
Exceptions	-

Figure 6. Use case of login.

3.4.3 Use case 3. Logout

Use case 3. Logout	
Requirements	R.1.1.3.
Preconditions	The user must have made the login process (see use case 2) successfully.
Actors	Authorised users.
Description	An authorised user will logout from an account.
Sequence of steps	1. The user will click the "Log out" button.
Postconditions	The user will be able to carry out the login process again with the username and password that he used.
Exceptions	-

Figure 7. Use case of logout.

3.4.4 Use case 4. Unregister

Use case 4. Unregister	
Requirements	R.1.1.4.
Preconditions	The user must have made the login process (see use case 2) successfully.
Actors	Authorised users.
Description	An authorised user will delete his account.
Sequence of steps	1. The user will click the "Unregister" button. 2. The user will click the confirmation button.
Postconditions	The user will not be able to carry out the login process again with the username and password that he used.
Exceptions	-

Figure 8. Use case of unregister.

3.4.5 Use case 5. Create a ShEx document

Use case 5. Create a ShEx document	
Requirements	R.1.2.1.
Preconditions	The user must have made the login process (see use case 2) successfully.
Actors	Authorised and non-owner users.
Description	An authorised user will create a new ShEx document.
Sequence of steps	1. The user will click the "Create a ShEx doc" button. 2. The user will write a non-empty title for the ShEx document.
Postconditions	-
Exceptions	-

Figure 9. Use case of create a ShEx document.

3.4.6 Use case 6. See the owned ShEx document

Use case 6. See the owned ShEx document	
Requirements	R.1.2.2.
Preconditions	The user must have made the login process (see use case 2) successfully. The user must also have made the ShEx document creation process (see use case 5) successfully.
Actors	Authorised and owner users.
Description	An authorised and owner user will be able to see its ShEx documents.
Sequence of steps	1. The user will open the ShareYASHE application.
Postconditions	The user will be able to see the name of the ShEx document in a list, shown on the Website.
Exceptions	-

Figure 10. Use case of see the owned ShEx document.

3.4.7 Use case 7. Edit an owned ShEx document

Use case 7. Edit an owned ShEx document	
Requirements	R.1.2.3.
Preconditions	One user must have seen the ShEx document (see use case 6) successfully.
Actors	Authorised users, and owner and non-owner users.
Description	Two authorised and owner users will be able to edit a ShEx document.
Sequence of steps	1. The user 1 will open the ShareYASHE application. 2. The user 1 will click on the ShEx document in the owned ShEx documents list. 3. The user 1 will share the ShEx document with the user 2. 4. The user 2 will open the ShareYASHE application through the sharing option that the user 1 shared with him. 5. Both user 1 and user 2 will write random words in the ShEx document at the same time.
Postconditions	Both user 1 and user 2 will be able to see the content of the ShEx document. Both user 1 and user 2 will be able to edit the content of the ShEx document. User 1 will be able to see the user 2 editing the document. User 2 will be able to see the user 1 editing the document.
Exceptions	-

Figure 11. Use case of edit an owned ShEx document.

3.4.8 Use case 8. Add an owner to a ShEx document

Use case 8. Add an owner to a ShEx document	
Requirements	R.1.2.4.
Preconditions	One user (user 1) must have seen the ShEx document (see use case 6) successfully.
Actors	Authorised and owner users.
Description	An authorised and owner user will be able to add a new owner to its ShEx documents.
Sequence of steps	1. The user 1 will open the ShareYASHE application. 2. The user 1 will click on the ShEx document in the owned ShEx documents list. 3. The user 1 will type the username of user 2 in the “Add a new owner” field.
Postconditions	The user will be able to see the name of the ShEx document in a list, shown on the Website.
Exceptions	-

Figure 12. Use case of add an owner to a ShEx document.

3.4.9 Use case 9.1. Leave the ownership of a ShEx document that the user created

Use case 9.1. Leave the ownership of a ShEx document that the user created	
Requirements	R.1.2.5.
Preconditions	One user (user 1) must have seen the ShEx document (see use case 6) successfully.
Actors	Authorised and owner users.
Description	An authorised and owner user will be able to leave the ownership of its ShEx documents, that he created.
Sequence of steps	1. The user 1 will open the ShareYASHE application. 2. The user 1 will click on the ShEx document in the owned ShEx documents list. 3. The user 1 will click on the "Leave the ownership" option.
Postconditions	The user 1 will not be able to see the name of the ShEx document in a list, shown on the Website.
Exceptions	-

Figure 13. Use case of leave the ownership of a ShEx document that the user created.

3.4.10 Use case 9.2. Leave the ownership of a ShEx document invited by another user

Use case 9.2. Leave the ownership of a ShEx document invited by another user	
Requirements	R.1.2.5.
Preconditions	One user (user 1) must have made the ShEx document creation process (see use case 5) successfully. This user (user 1) must have made the ShEx document invitation process (see use case 8) to the other user (user 2) successfully. The other user (user 2) must have seen the ShEx document (see use case 6) successfully.
Actors	Authorised and owner users.
Description	An authorised and owner user will be able to leave the ownership of its ShEx documents, that he was invited by another user.
Sequence of steps	1. The user 2 will open the ShareYASHE application. 2. The user 2 will click on the ShEx document in the owned ShEx documents list. 3. The user 2 will click on the "Leave the ownership" option.
Postconditions	The user 2 will not be able to see the name of the ShEx document in a list, shown on the Website.
Exceptions	-

Figure 14. Use case of leave the ownership of a ShEx document invited by another user.

4 Software design and architecture

4.1 Overview

The ShareYASHE application presents itself as two collaborating Software units: the ShareYASHE server, and the ShareYASHE client.

When the client makes an HTTP request against the ShareYASHE server (for example: GET /shEx/123456789), the server replies with an HTTP response. Not only this response comes with an HTML rendered view, but also with a single JavaScript sheet which will run on client-side; we will name this scriptsheet as **ShareYASHE client**.

The responsibility of this Software unit is to stablish a parallel WebSocket connection which subscribes the user's YASHE editor to its Ydoc object in the WebSocket server of ShareYASHE server. Every ShEx doc on ShareYASHE has its Ydoc object, which works as a version control (like Git does), and which all the users editing it must subscribe to.

Since the ShareYASHE client is a single codesheet, it lacks any structure. Thus, we will be solely reviewing the **ShareYASHE server** architecture.

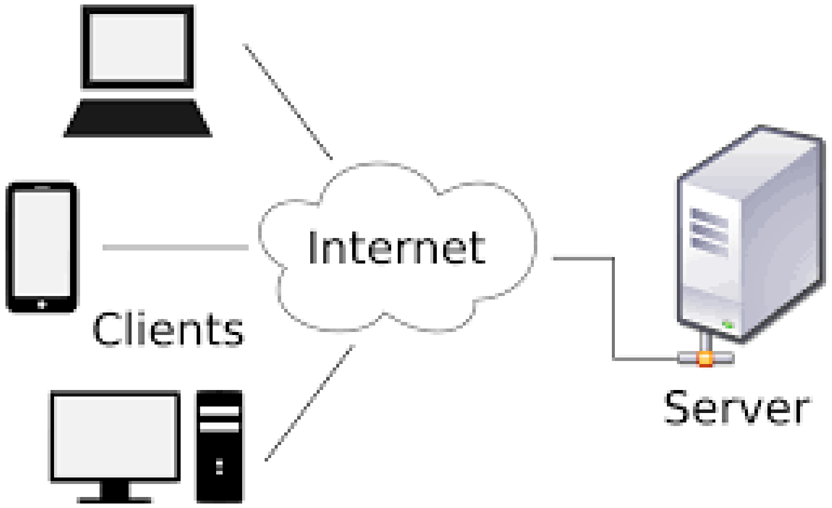


Figure 15. Representation of client-server connection.

4.2 Overall architecture

The general structure of the ShareYASHE server presents **two parallel servers**: one following the **HTTP** protocol, which is the core of the application, and the other following the **WebSocket** protocol.

The ShareYASHE HTTP server implementation follows a 5 layer architecture, which separates: HTTP request handling (Routes), high level functionality (Application layer), view rendering (Presentation layer), low level functionality (Business layer), and data storage (Persistence layer).

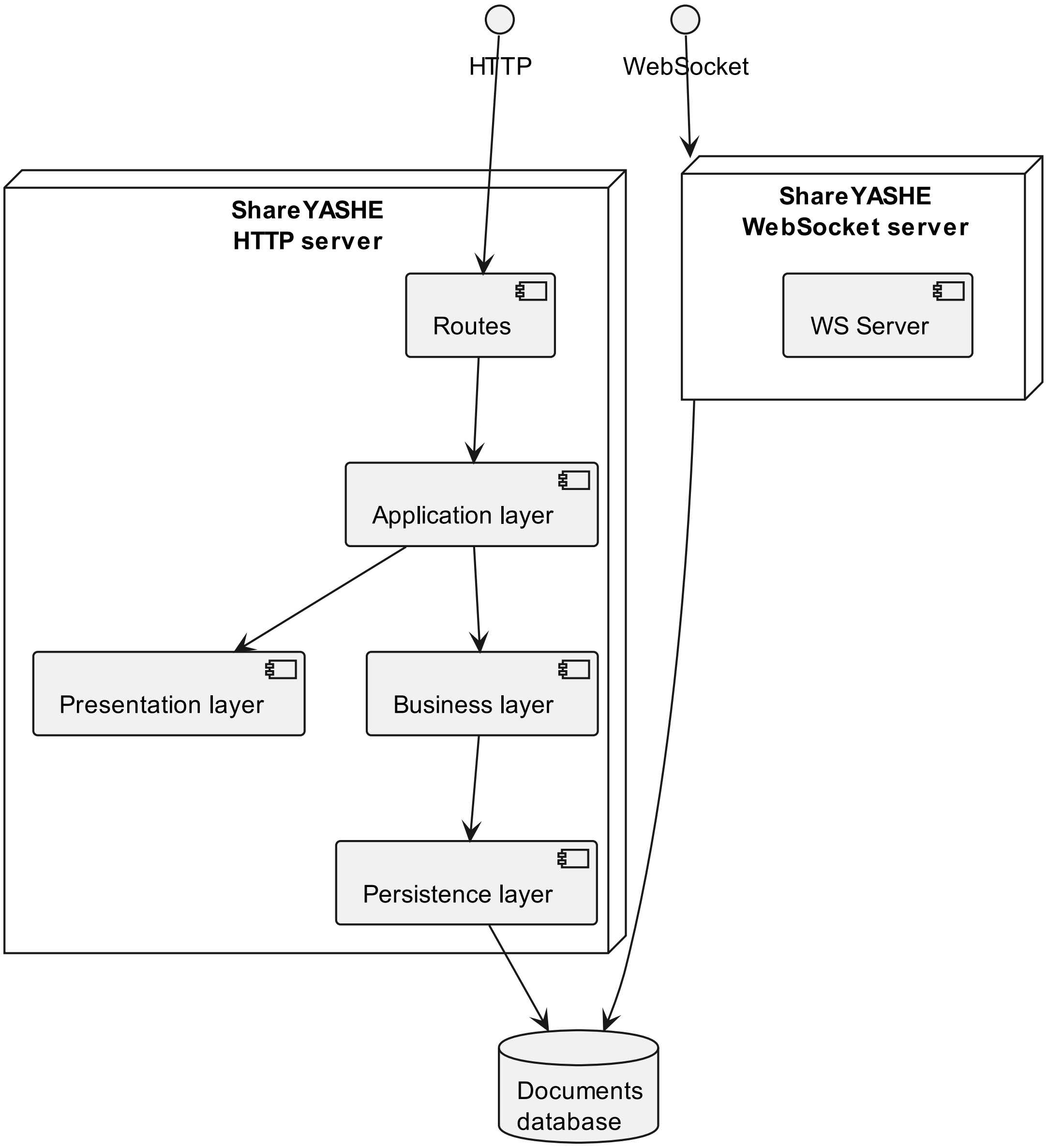


Figure 16. ShareYASHE overall architecture.

4.3 WebSocket server component

The WebSocket server component will contain the logic for carrying out petitions from the user's ShareYASHE client. These petitions are limited to just synchronising the ShEx doc content in real time. It will rely on the database, so changes would persist, even if the server shuts down, or crashes.

4.4 5 layers architecture

4.4.1 Routes layer

The routes component is responsible for grabbing any HTTP request and redirect it to its corresponding implementation.

Each class of this layer responds to a conceptual family of HTTP requests, being those: "ShExDocs", "Auth", and "Index". The handling of the request is delegated to an application layer class through its factory "AppLayerFactory". Note that this layer is the only one whose implementation follows the functional paradigm.

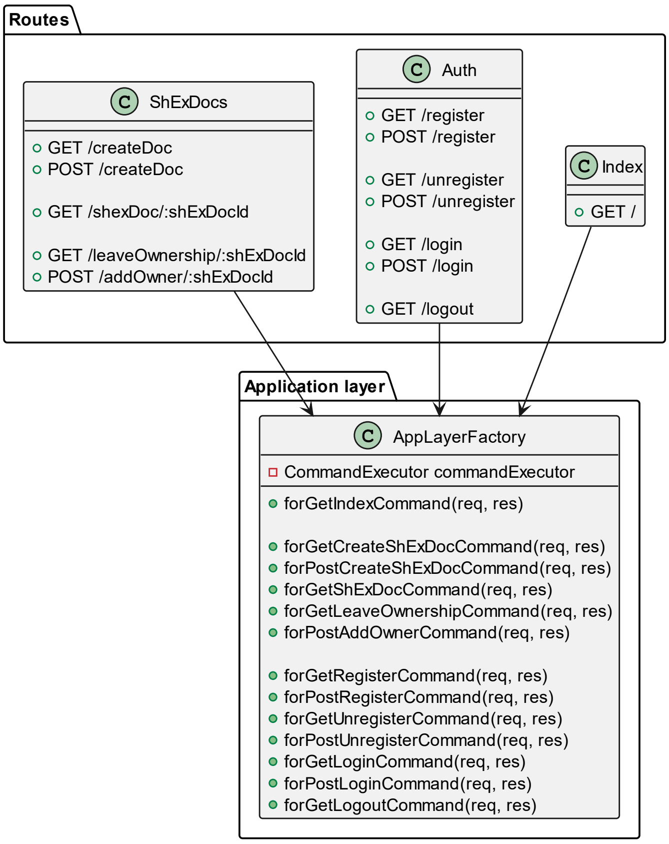

Routes
C ShExDocs
GET /createDoc
POST /createDoc
GET /shexDoc/:shExDocId
GET /leaveOwnership/:shExDocId
POST /addOwner/:shExDocId
C Auth
GET /register
POST /register
GET /unregister
POST /unregister
GET /login
POST /login
GET /logout
C Index
GET /
Application layer
C AppLayerFactory
CommandExecutor commandExecutor
forGetIndexCommand(req, res)
forGetCreateShExDocCommand(req, res)
forPostCreateShExDocCommand(req, res)
forGetShExDocCommand(req, res)
forGetLeaveOwnershipCommand(req, res)
forPostAddOwnerCommand(req, res)
forGetRegisterCommand(req, res)
forPostRegisterCommand(req, res)
forGetUnregisterCommand(req, res)
forPostUnregisterCommand(req, res)
forGetLoginCommand(req, res)
forPostLoginCommand(req, res)
forGetLogoutCommand(req, res)

Figure 17. Routes layer architecture.

4.4.2 Application layer

This layer will provide high level implementation for each HTTP petition, taking advantage of several Software patterns[GoF94]. This layer must be only accessed from a functional factory, which is responsible for dynamically importing each command class from the business layer and instantiating it. Ultimately, errors are unconditionally trapped here.

As we dive into the ShareYASHE server architecture, from top (App) to bottom (Persistence layer), we realise that it follows the procedural paradigm, until we reach any command class from this application layer. Any piece of code down below will pursue the object-oriented paradigm.

All these command objects must implement an “execute(req, res)” façade method, which will carry a chain of responsibility for the “req” and the “res” objects.

Another chain of responsibility exists for the “app” object (from the Express framework), and for the “mongoClient” object. This is carried out in the constructor of the command.

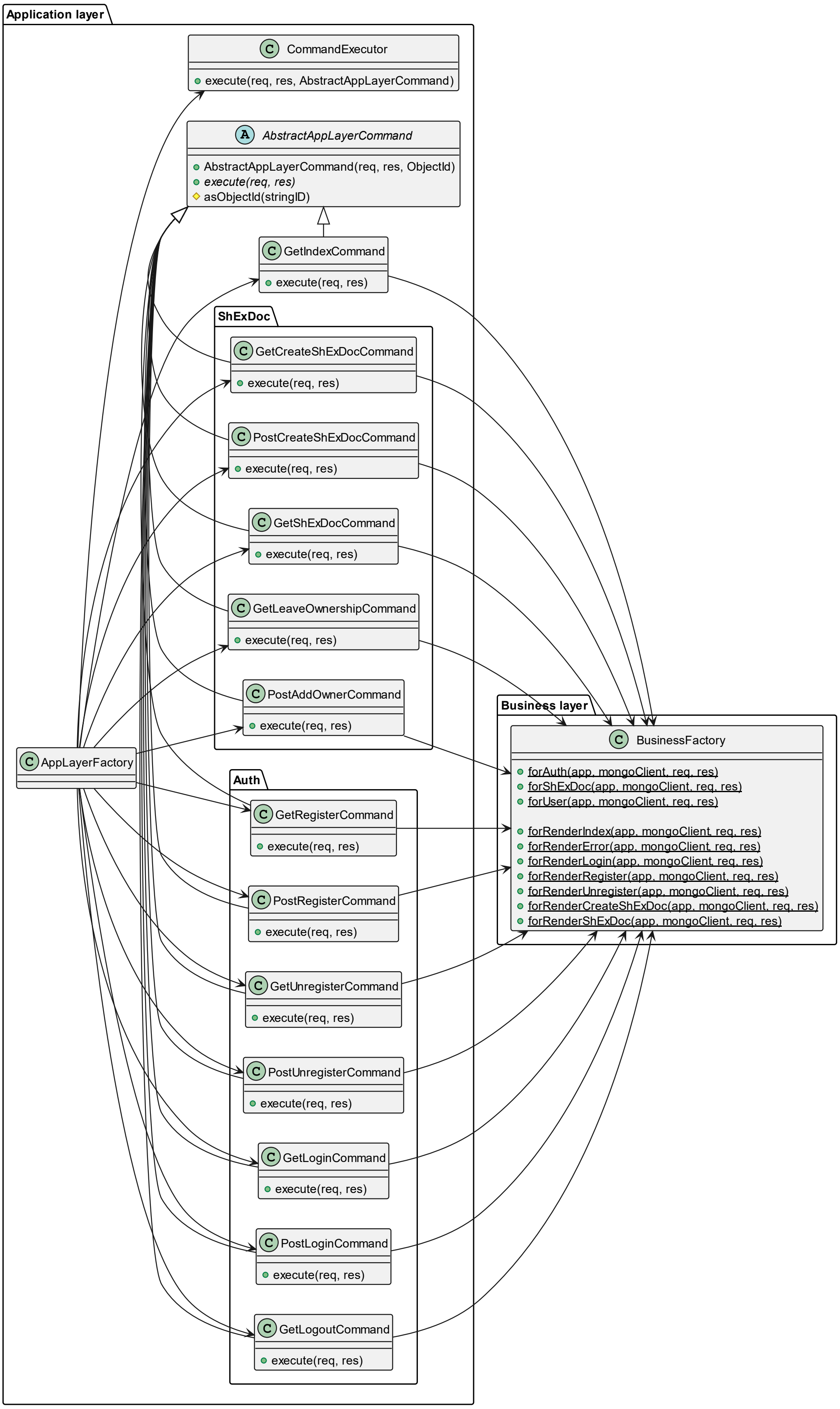

Application layer
C CommandExecutor
execute(req, res, AbstractAppLayerCommand)
A AbstractAppLayerCommand
AbstractAppLayerCommand(req, res, ObjectId)
execute(req, res)
asObjectId(stringID)
C GetIndexCommand
execute(req, res)
ShExDoc
C GetCreateShExDocCommand
execute(req, res)
C PostCreateShExDocCommand
execute(req, res)
C GetShExDocCommand
execute(req, res)
C GetLeaveOwnershipCommand
execute(req, res)
C PostAddOwnerCommand
execute(req, res)
C AppLayerFactory
Auth
C GetRegisterCommand
execute(req, res)
C PostRegisterCommand
execute(req, res)
C GetUnregisterCommand
execute(req, res)
C PostUnregisterCommand
execute(req, res)
C GetLoginCommand
execute(req, res)
C PostLoginCommand
execute(req, res)
C GetLogoutCommand
execute(req, res)
Business layer
C BusinessFactory
forAuth(app, mongoClient, req, res)
forShExDoc(app, mongoClient, req, res)
forUser(app, mongoClient, req, res)
forRenderIndex(app, mongoClient, req, res)
forRenderError(app, mongoClient, req, res)
forRenderLogin(app, mongoClient, req, res)
forRenderRegister(app, mongoClient, req, res)
forRenderUnregister(app, mongoClient, req, res)
forRenderCreateShExDoc(app, mongoClient, req, res)
forRenderShExDoc(app, mongoClient, req, res)

Figure 18. Application layer architecture.

4.4.3 Business layer

The business layer implements common operations relating to the domain of the application, such as validating a user login, or creating a new ShEx doc given an owner user.

The instantiation of any object from this layer upwards converges in a "Business Factory" static class, which follows the simple factory pattern.

This factory is responsible for instantiating a service object, in which associated functionality is encapsulated, as seen in the service layer pattern[Fowler02]. The chains of responsibility that we were talking about in the previous layer join themselves here, in the constructor of the service object.

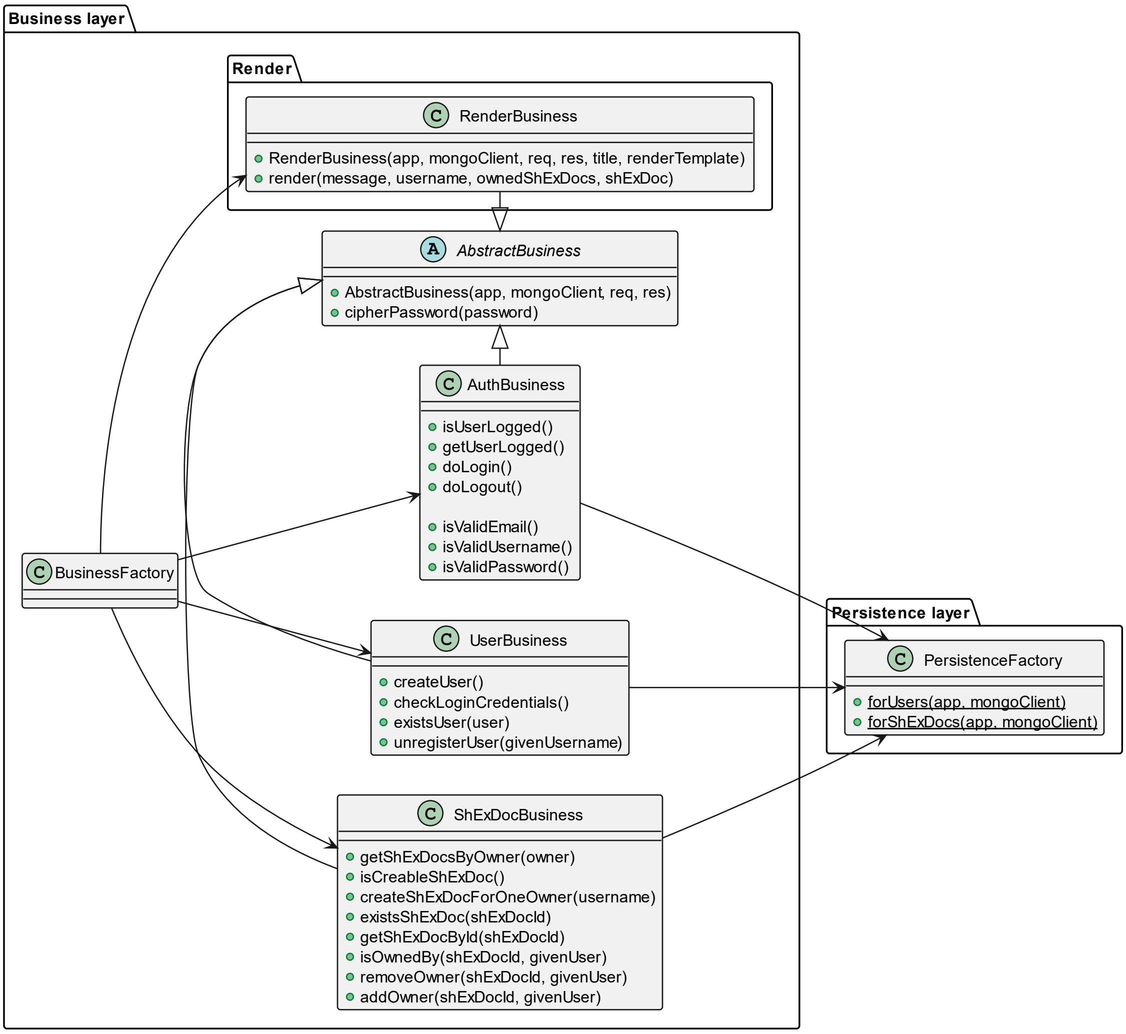

Business layer
Render
C RenderBusiness
RenderBusiness(app, mongoClient, req, res, title, renderTemplate)
render(message, username, ownedShExDocs, shExDoc)
A AbstractBusiness
AbstractBusiness(app, mongoClient, req, res)
cipherPassword(password)
C AuthBusiness
isUserLogged()
getUserLogged()
doLogin()
doLogout()
isValidEmail()
isValidUsername()
isValidPassword()
C BusinessFactory
C UserBusiness
createUser()
checkLoginCredentials()
existsUser(user)
unregisterUser(givenUsername)
C ShExDocBusiness
getShExDocsByOwner(owner)
isCreableShExDoc()
createShExDocForOneOwner(username)
existsShExDoc(shExDocId)
getShExDocById(shExDocId)
isOwnedBy(shExDocId, givenUser)
removeOwner(shExDocId, givenUser)
addOwner(shExDocId, givenUser)
Persistence layer
C PersistenceFactory
forUsers(app, mongoClient)
forShExDocs(app, mongoClient)

Figure 19. Business layer architecture.

4.4.4 Presentation layer

The presentation layer (named as "Render" in the diagram of the previous section) will be responsible for rendering the templates into Web pages, which the user's navigator will interpret. Depending on how the execution of each command from the application layer flows, the rendering logic will be one or another; that must be encapsulated in this layer.

Despite this layer being not different from the business layer, I realised that this differentiation must be created, at least conceptually, because the service objects instantiated here are slightly different from those created in the actual business layer.

The presentation layer service calls the template for rendering the user's final view, and they **must never call the persistence layer**. This decouples business and persistence implementation from rendering implementation.

4.4.5 Persistence layer

The persistence layer directly manipulates the database API in order to abstract it into simple CRUD operations.

This layer has a similar structure to the business layer, given that every DAO object gets created in a "Persistence Factory", another simple factory. In this case, the intention of the objects of this layer is different: they will give functionality for CRUD operations, as specified in the DAO pattern[Fowler02].

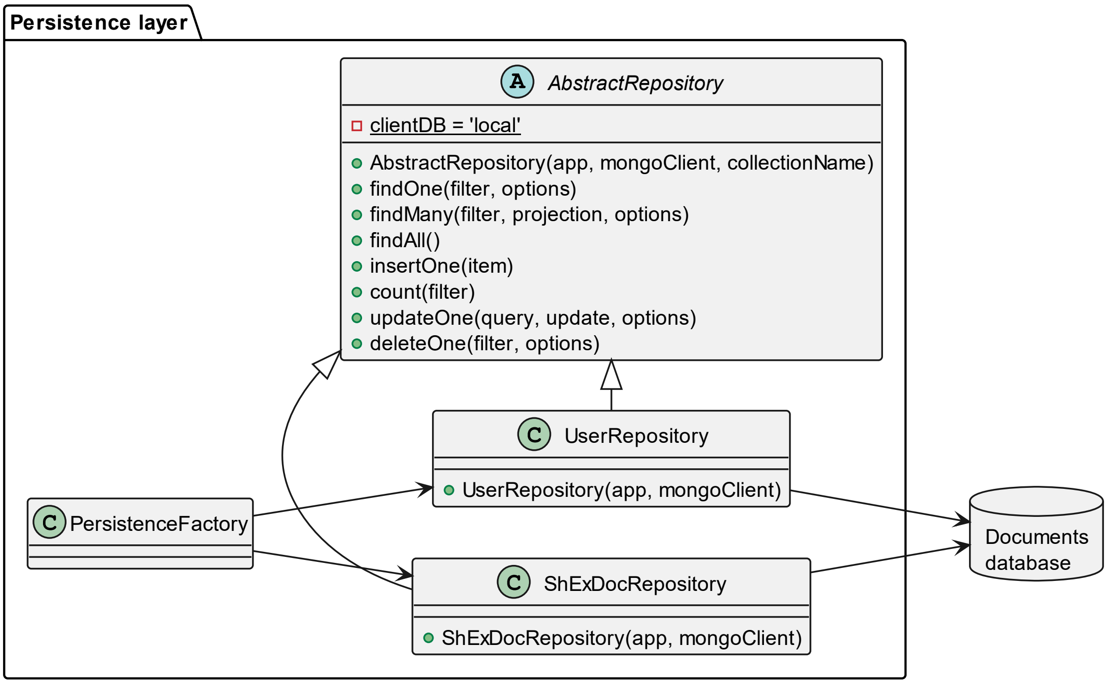

Persistence layer
A AbstractRepository
clientDB = 'local'
AbstractRepository(app, mongoClient, collectionName)
findOne(filter, options)
findMany(filter, projection, options)
findAll()
insertOne(item)
count(filter)
updateOne(query, update, options)
deleteOne(filter, options)
C UserRepository
UserRepository(app, mongoClient)
C PersistenceFactory
C ShExDocRepository
ShExDocRepository(app, mongoClient)
Documents database

Figure 20. Persistence layer architecture.

4.5 Structure in deployment

4.5.1 Deployment of ShareYASHE server

ShareYASHE server can be deployed in any machine. The database shall be deployed in the same machine, but we can easily change the reference to an external database. By default, ShareYASHE server is deployed as a two-headed server, which listens both HTTP and WebSockets requests.

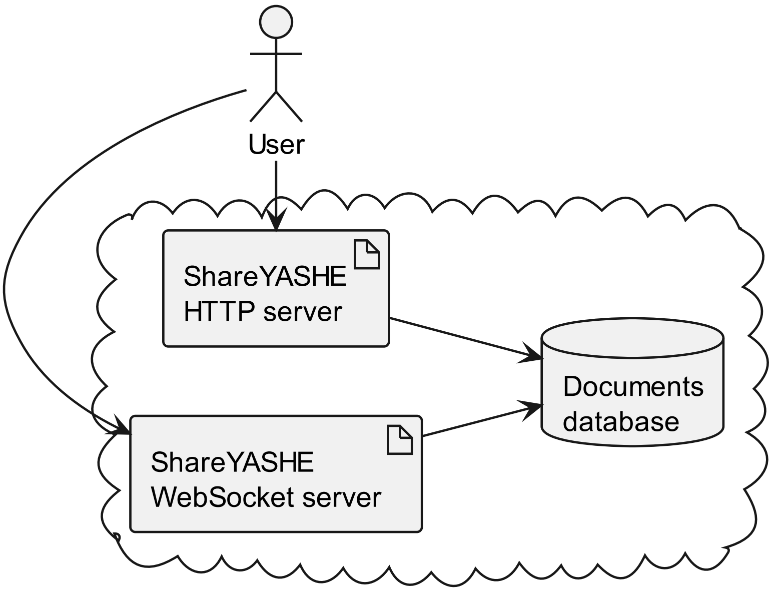


Figure 21. Deployment of ShareYASHE server.

The ShareYASHE client is automatically deployed in the client machine through the HTTP response of the ShEx doc GET request.

4.5.2 Deployment of ShareYASHE client

With the previous information, the deployment of ShareYASHE client can be difficult to understand. Thus, we are going to see in this chapter a more detailed diagram of its deployment.

As we can see, the ShareYASHE server renders the HTML view after an HTTP request was received and returns it to the user. However, when the user accesses the edition of a ShEx document, the server also responds with client-side JavaScript source code, which is ShareYASHE client. ShareYASHE client subscribes itself to the ShareYASHE WebSocket server. This lets the user's editor to subscribe to the updatable content of the ShEx document.

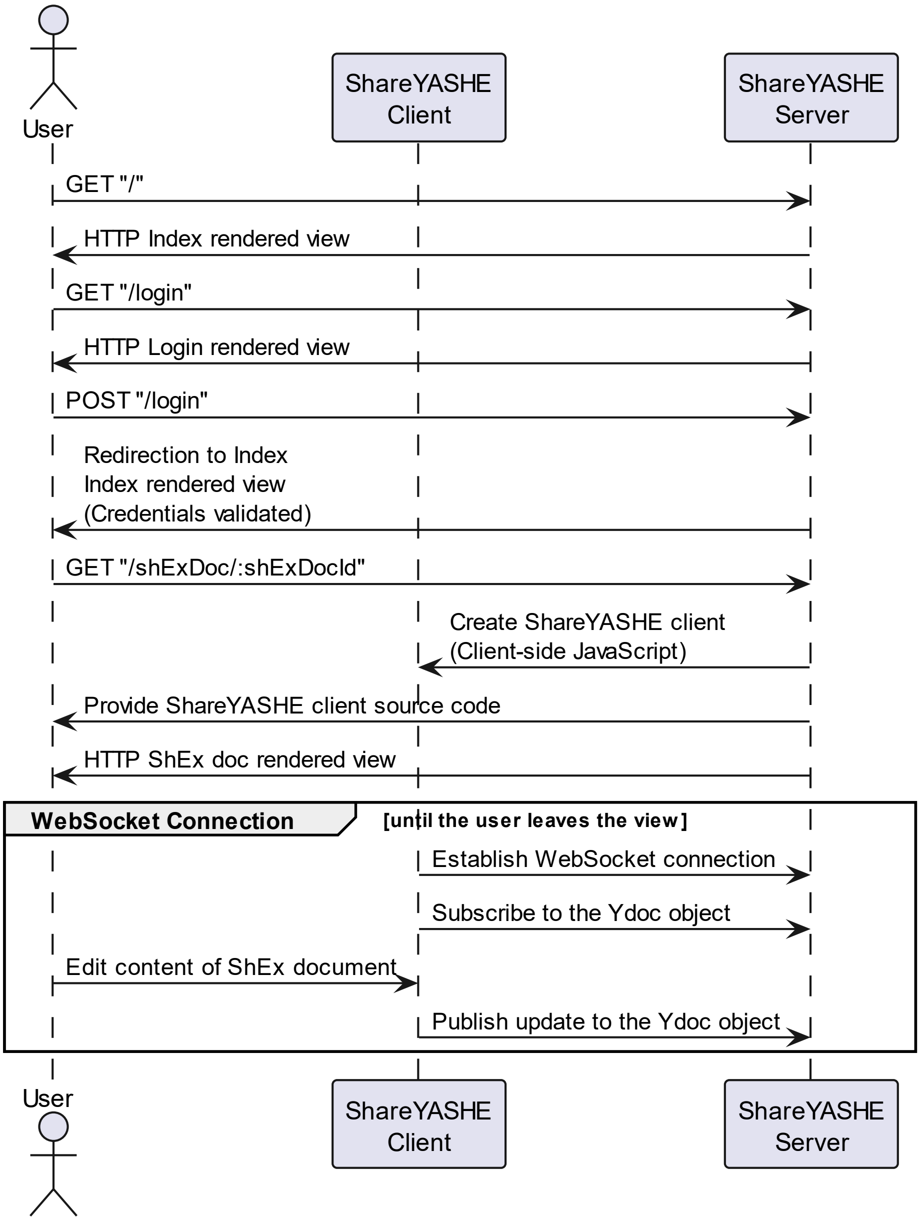


Figure 22. Deployment of ShareYASHE client.

4.6 Design of the DBMS

4.6.1 Document-oriented DBMS

In the development of ShareYASHE, we will be using a **document-oriented database**, which is a type of NoSQL database that stores and retrieves data in a document-like format. In a document-oriented database, data is typically represented using a standardized format such as JSON (JavaScript Object Notation), XML (eXtensible Markup Language), or BSON (Binary JSON). These formats provide a way to store data in a hierarchical structure, similar to nested key-value pairs or a tree-like structure.

Instead of storing data in rows and columns like relational databases, document-oriented databases store data as **self-contained documents**. Each document represents a single entity or object and contains all related information. Documents can have different structures, allowing for schema flexibility.

Document-oriented databases have a **dynamic schema**, meaning that each document can have its own structure and fields. This flexibility enables developers to easily add or modify fields within a document without requiring a predefined schema alteration. It accommodates evolving data models and simplifies application development.

Documents in a document-oriented database can have **nested structures**, allowing for complex and rich data models. This hierarchical representation enables the storage of related data together within a single document, reducing the need for complex joins and facilitating efficient data retrieval.

Document-oriented databases provide **powerful querying capabilities** to retrieve and manipulate data. They typically support querying based on the structure and values within documents, allowing for flexible searches across multiple fields and nested structures. Query languages like MongoDB's Query Language (MQL) or Couchbase's N1QL are used for this purpose.

Document-oriented databases are designed to **scale horizontally**, allowing for distributing data across multiple servers or clusters. This scalability is achieved through techniques such as sharding, where data is partitioned and stored on different machines, enabling high-performance and efficient handling of large data volumes.

Document-oriented databases often offer built-in **replication mechanisms** to ensure data **durability** and **availability**. Replication involves creating multiple copies of data across different servers, enabling fault tolerance and high availability in case of failures.

4.6.2 Data model

The data model tends to be simple, due to the DBMS being document-oriented. In this DBMS, we can find two databases: “Entities” and “Ydocs”. “Entities” will have two collections: “Users” and “ShExDocs”. The first one will store the user info, and the second one will store the metadata of every ShEx document in the System.

Meanwhile, the “Ydocs” database will store one collection per ShEx document. This database is set up and used by the Yjs library in order to save the real-time updates of the concurrent document. Conceptually, the persistence of each shareable ShEx document is treated as if it were a Git repository; this means, there is no “content” field for a ShEx document, in fact there is a collection for each ShEx document, whose entries are each of them updates made to the ShEx document’s content, for example: an insertion of a character, a deletion of one, etc.

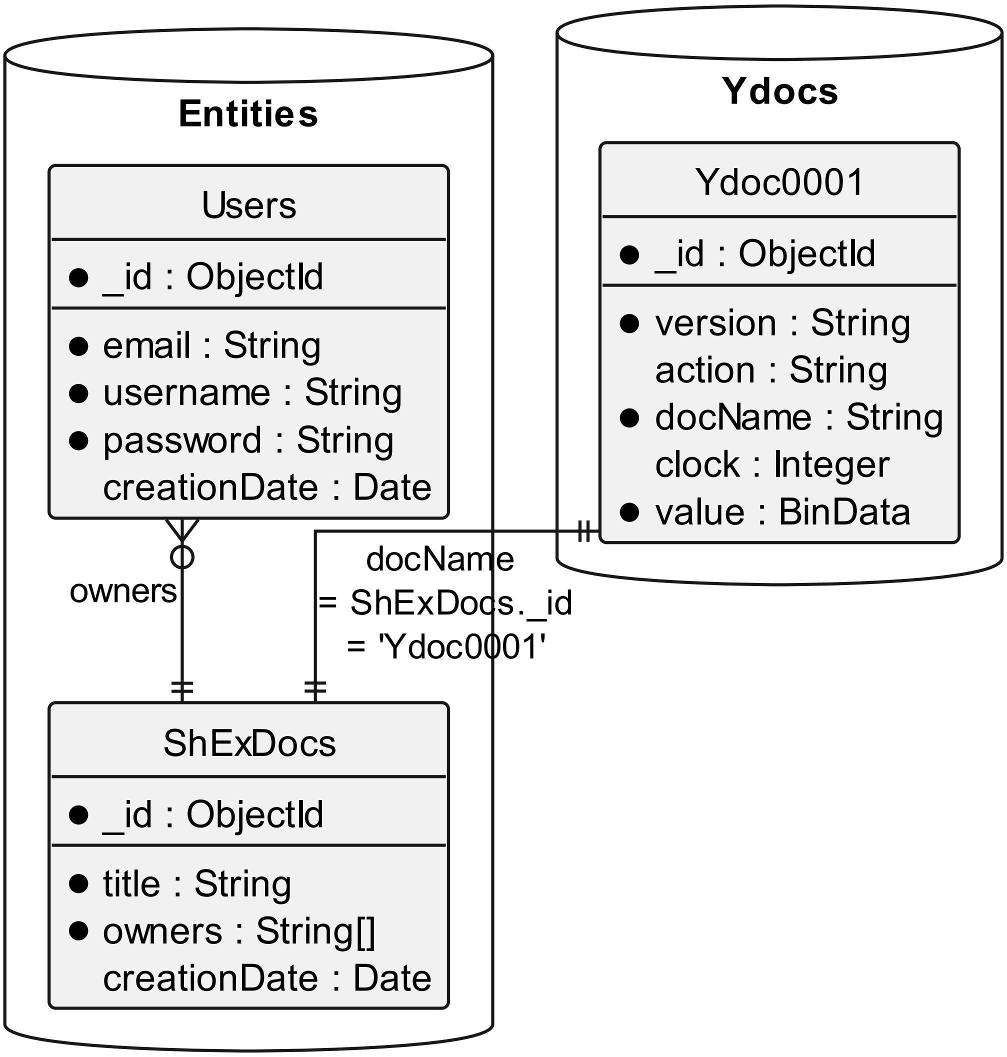


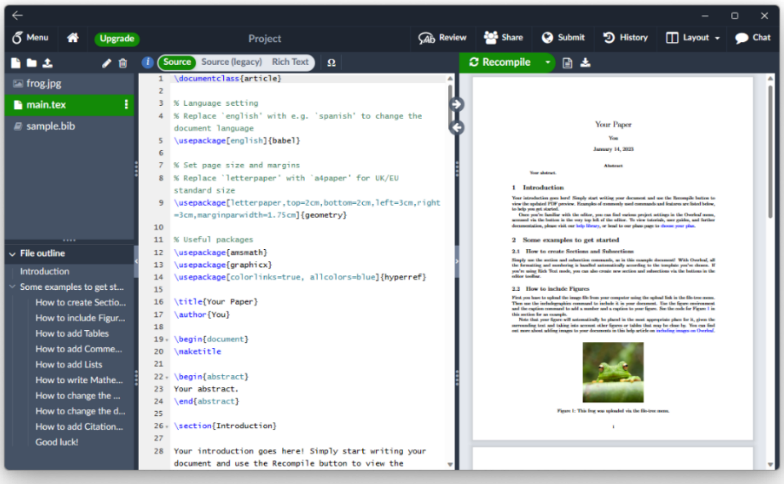
Figure 23. ShareYASHE data model.

4.7 User interface design

4.7.1 Main intention

The design of the user interface of ShareYASHE is meant to be both accessible and usable. As a accessibility reference, the WCAG 2.0 recommendation was followed[WCAG08], besides usability guidelines[Krug13].

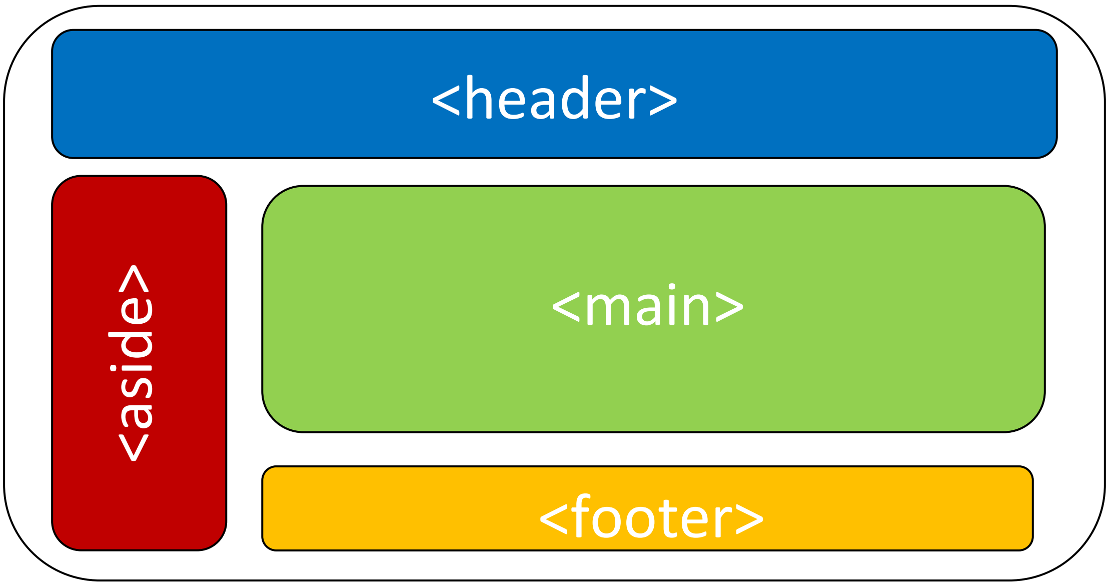
The basic concept of the layout of ShareYASHE was inspired in Overleaf's.



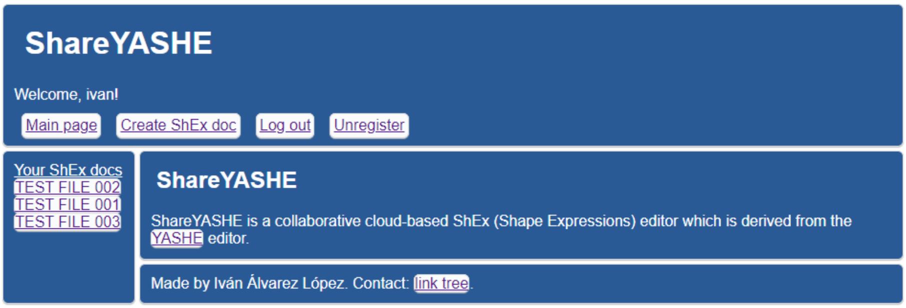
```
\documentclass{article}

% Language setting
% Replace `english' with e.g. `spanish' to change the
document language
\usepackage[english]{babel}

% Set page size and margins
% Replace `letterpaper' with `a4paper' for UK/EU
standard size
\usepackage[letterpaper,top=2cm,bottom=2cm,left=3cm,right
=3cm,marginparwidth=1.75cm]{geometry}

% Useful packages
\usepackage{amsmath}
\usepackage{graphicx}
\usepackage[colorlinks=true, allcolors=blue]{hyperref}

\title{Your Paper}
\author{You}

\begin{document}
\maketitle

\begin{abstract}
Your abstract.
\end{abstract}

\section{Introduction}

Your introduction goes here! Simply start writing your
document and use the Recompile button to view the
```

Figure 24. Inspirational layout for ShareYASHE.

I reinterpreted the layout using pure HTML & CSS to guarantee a fully adaptable, and accessible, Website.



As you can see, the intended look and feel for the index Webpage was completed.



Welcome, ivan!

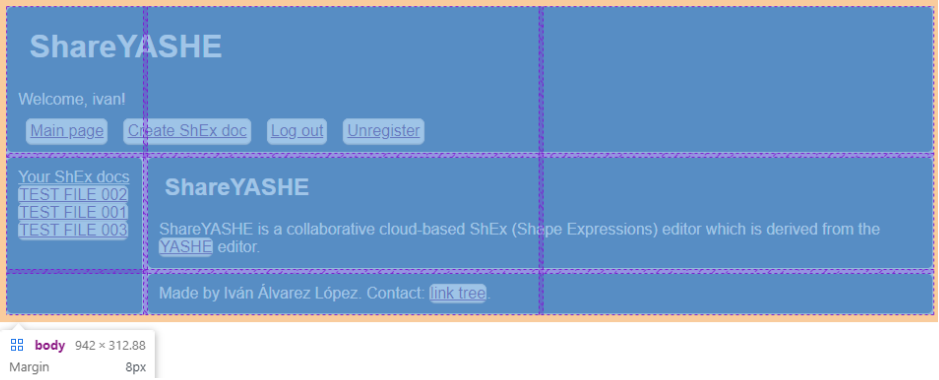
ShareYASHE

ShareYASHE is a collaborative cloud-based ShEx (Shape Expressions) editor which is derived from the YASHE editor.

Made by Iván Álvarez López. Contact: link tree.

Figure 25. Sample of ShareYASHE layout.

This was achieved thanks to the grid layout of CSS, which was intended for adaptability from its conception.



Welcome, ivan!

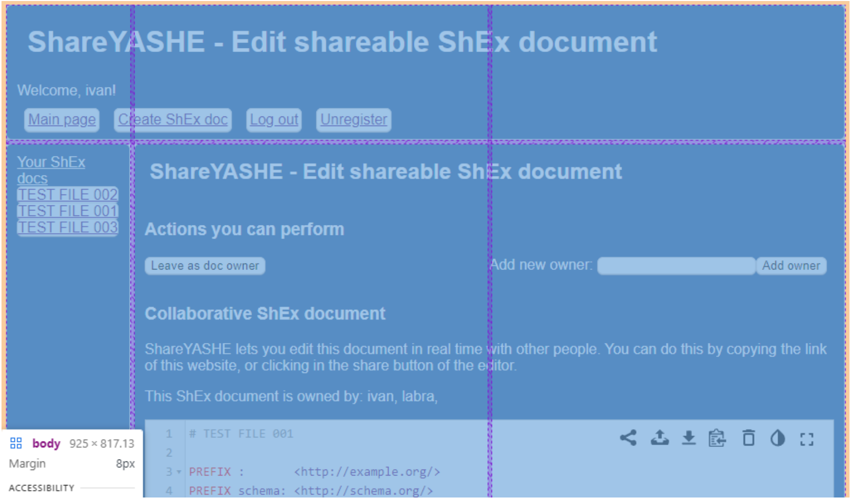
ShareYASHE

ShareYASHE is a collaborative cloud-based ShEx (Shape Expressions) editor which is derived from the YASHE editor.

Made by Iván Álvarez López. Contact: link tree.

Figure 26. Griding of ShareYASHE layout.

This structure is followed in any page of ShareYASHE. The grid layout will take care of automatic adaption.



Welcome, ivan!

ShareYASHE - Edit shareable ShEx document

Actions you can perform

Add new owner:

Collaborative ShEx document

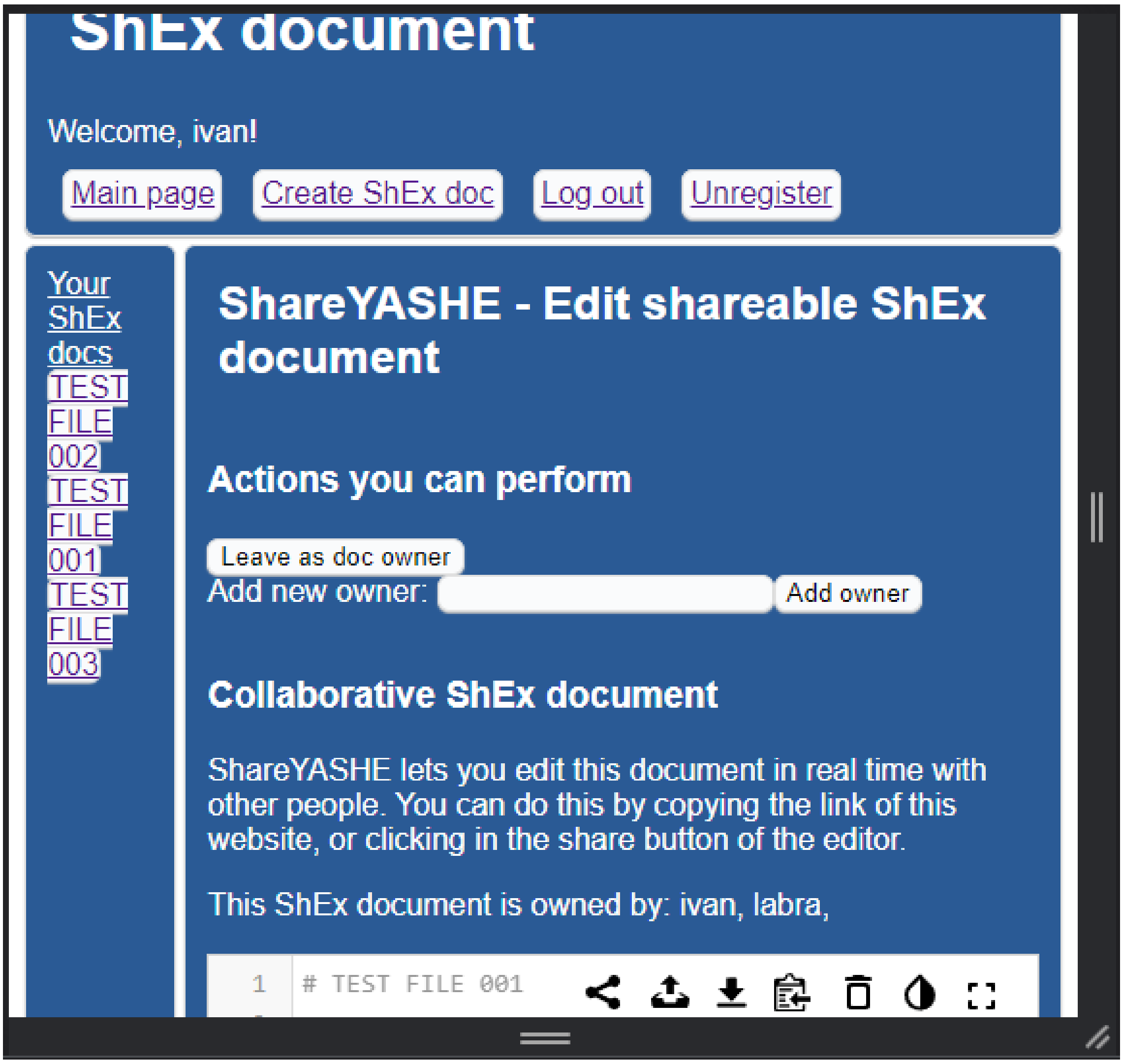
ShareYASHE lets you edit this document in real time with other people. You can do this by copying the link of this website, or clicking in the share button of the editor.

This ShEx document is owned by: ivan, labra,

```
# TEST FILE 001

PREFIX :       <http://example.org/>
PREFIX schema: <http://schema.org/>
```

Figure 27. Griding of ShareYASHE editor layout.

Even in smartphones, ShareYASHE will be fully adaptable.



ShEx document

Welcome, ivan!

Main page | Create ShEx doc | Log out | Unregister

Your ShEx docs
TEST FILE 002
TEST FILE 001
TEST FILE 003

ShareYASHE - Edit shareable ShEx document

Actions you can perform

Leave as doc owner
Add new owner: Add owner

Collaborative ShEx document

ShareYASHE lets you edit this document in real time with other people. You can do this by copying the link of this website, or clicking in the share button of the editor.

This ShEx document is owned by: ivan, labra,

```
# TEST FILE 001
```

Figure 28. ShareYASHE layout in smartphone.

A main template is shared between views, in which each one will override its <main> content for its particular purpose.

4.7.2 Styles of the application

4.7.2.1 General style

As we have already seen, ShareYASHE is governed by a principal layout: a grid-based layout. However, there are several style definitions that I have made to make more friendly the look and feel of the Website.

- A blue background colour (#2A5A95) that achieves enough contrast for the WCAG guidelines to be granted, given that the foreground colour of the application is full white (#FFFFFF).
- For every main element of the Website (header, main, aside, and footer), I defined a semi-white (#E0E0E0) solid border, a semi-grey (#C0C0C0) box shadow, and a border radius of 6px to reduce their sharpness.
- For links (<a> element), inputs, and buttons, I also defined a semi-white (#E0E0E0) solid border, a semi-grey (#C0C0C0) box shadow, and a border radius of 6px to reduce their sharpness. This, with a semi-white background colour (#FAFBFC) will let the users perceive the change of its state, this means, I can then apply a darker white tone (#F4F4F4) to achieve a hover style. For the active pseudoclass, I remove the box shadow to make these elements brighter.

4.7.2.2 General layout

The principal layout of ShareYASHE is a grid layout, having a size of 3x3. Its columns will follow the pattern of: 0.35fr, 1fr, and 1fr. A gap of 0.25em is defined, so elements don't overlap each other, and I also defined word breaking, with the word wrap property, so very long names don't get out of sight for users.

4.7.2.3 Specific layout for the editor actions

When we access the edition of the ShEx document, we can see that there are some actions, which are represented by form elements, which we can make, if we are owners of that ShEx document. These actions appear, by default, in the same row, thanks to a specific stylesheet that is inserted into this only view.

However, this becomes a problem if the width of the window is too small. In this case, using media queries, I remove this specific layout, letting the default flow layout to do its work, and stack the elements which perform these actions.

4.7.2.4 External stylesheets

We must remember that there are some stylesheets that are imported for the correct functionality of the Website. Nevertheless, they are restrained to be applied only to the YASHE editor, which is embedded in a text area element.

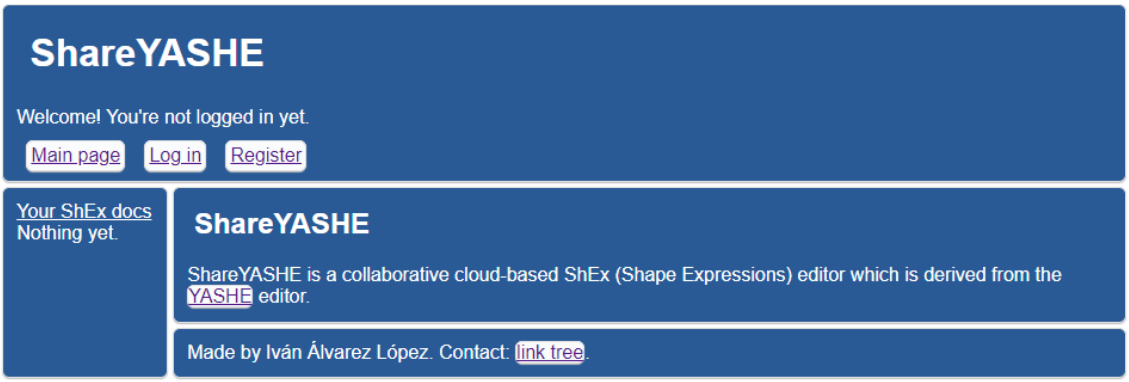
4.7.3 Possible views

In this chapter, we will review each view that ShareYASHE can render to the user.

4.7.3.1 Authenticated versus non authenticated

Depending on the state of authentication of the user, the content of the header element changes, also for the aside element. It shows the operations that the user can perform, given his state of authentication.

A non authenticated user will see this.



ShareYASHE

Welcome! You're not logged in yet.

Nothing yet.

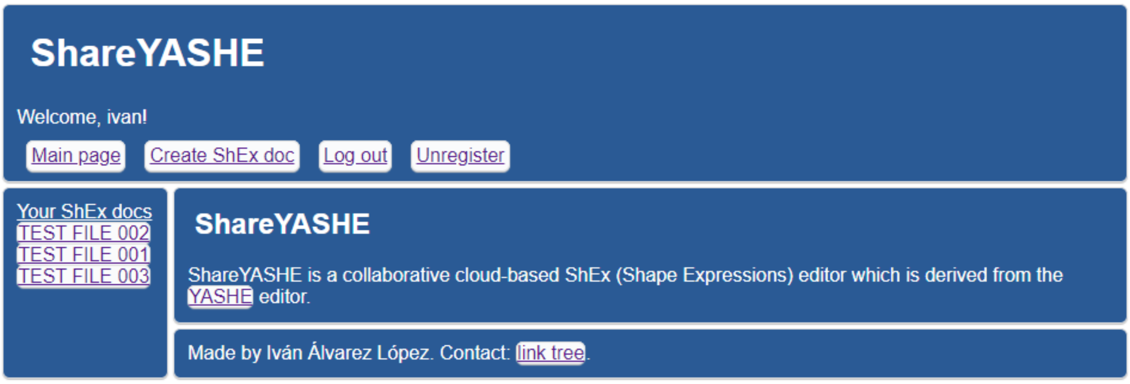
ShareYASHE

ShareYASHE is a collaborative cloud-based ShEx (Shape Expressions) editor which is derived from the YASHE editor.

Made by Iván Álvarez López. Contact: link tree.

Figure 29. ShareYASHE non-authenticated user layout.

An authenticated user will see this.



ShareYASHE

Welcome, ivan!

ShareYASHE

ShareYASHE is a collaborative cloud-based ShEx (Shape Expressions) editor which is derived from the YASHE editor.

Made by Iván Álvarez López. Contact: link tree.

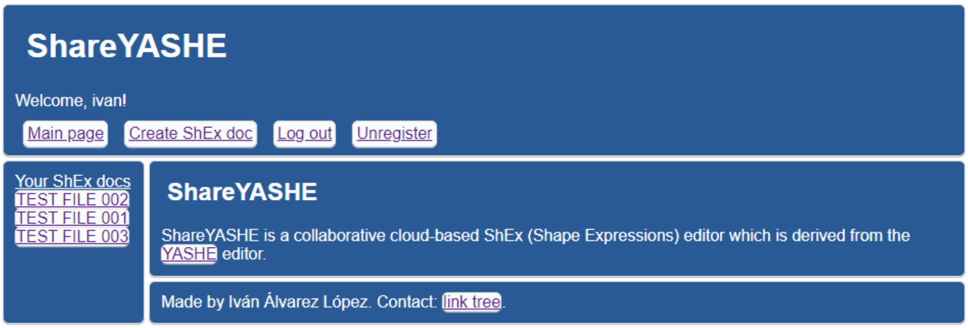
Figure 30. ShareYASHE authenticated layout.

When the user is logged in, he can view its owned ShEx documents. On the other hand, if he does not own any ShEx document, he will see the same content for the aside element as if he was not logged in.

4.7.3.2 Concrete views

The concrete views of ShareYASHE inherit the shape described in the previous chapter, except for the content of the main element. In this chapter, we will see how the content of the main element is shown to a user.

4.7.3.2.1 Main view



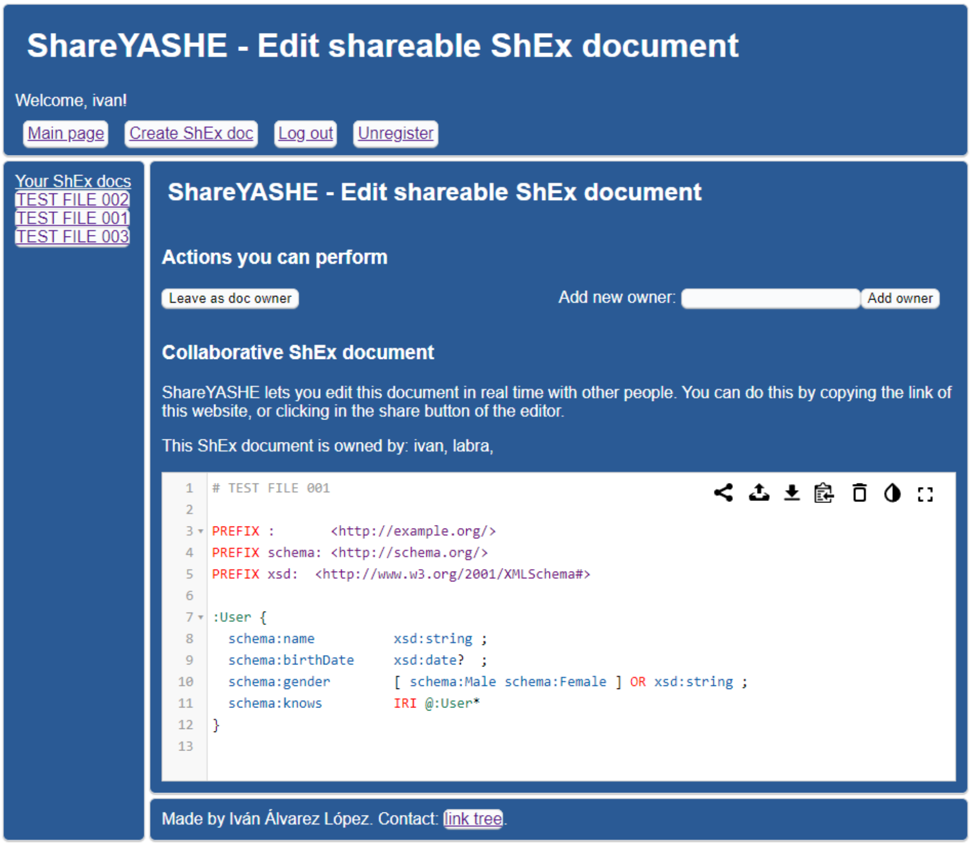
ShareYASHE

ShareYASHE is a collaborative cloud-based ShEx (Shape Expressions) editor which is derived from the YASHE editor.

Made by Iván Álvarez López. Contact: link tree.

Figure 31. ShareYASHE main view layout.

4.7.3.2.2 ShEx document editing view



ShareYASHE - Edit shareable ShEx document

Actions you can perform

Collaborative ShEx document

ShareYASHE lets you edit this document in real time with other people. You can do this by copying the link of this website, or clicking in the share button of the editor.

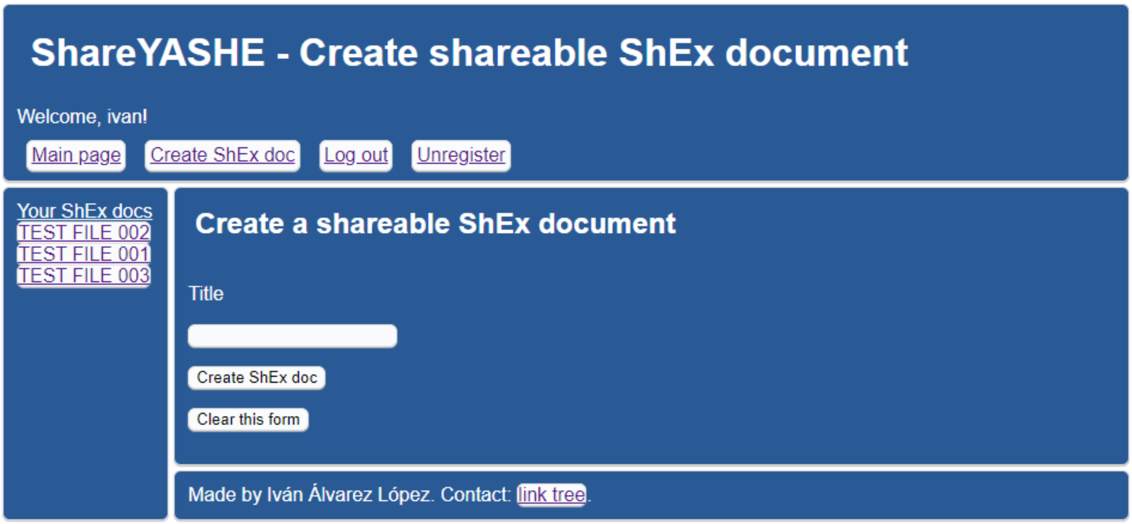
This ShEx document is owned by: ivan, labra,

```
# TEST FILE 001

PREFIX :       <http://example.org/>
PREFIX schema: <http://schema.org/>
PREFIX xsd:  <http://www.w3.org/2001/XMLSchema#>

:User {
  schema:name          xsd:string ;
  schema:birthDate     xsd:date?  ;
  schema:gender        [ schema:Male schema:Female ] OR xsd:string ;
  schema:knows         IRI @:User*
}

```

Made by Iván Álvarez López. Contact: link tree.

Figure 32. ShareYASHE ShEx document editing view layout.

4.7.3.2.3 Create ShEx document view



ShareYASHE - Create shareable ShEx document

Welcome, ivan!

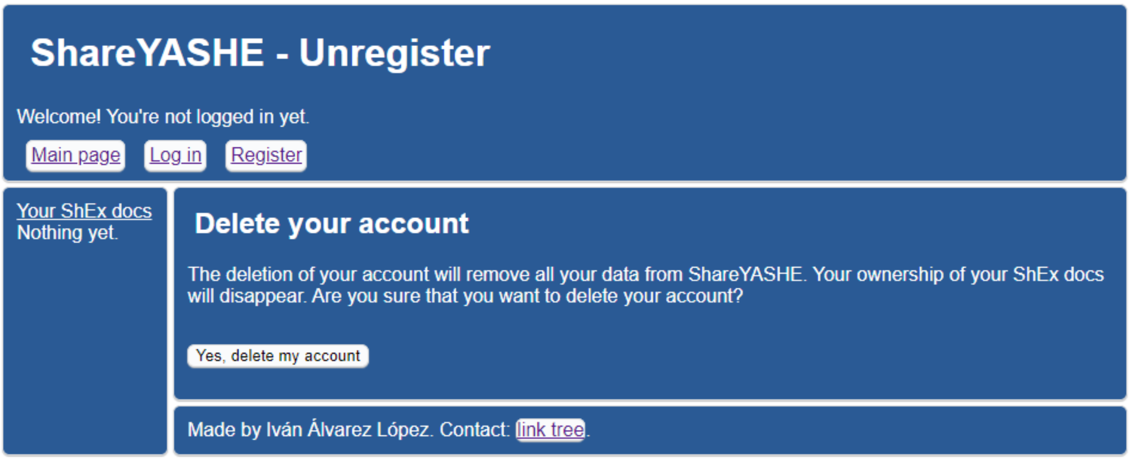
Create a shareable ShEx document

Title

Made by Iván Álvarez López. Contact: link tree.

Figure 33. ShareYASHE create ShEx document view layout.

4.7.3.2.4 Unregister view



ShareYASHE - Unregister

Welcome! You're not logged in yet.

Nothing yet.

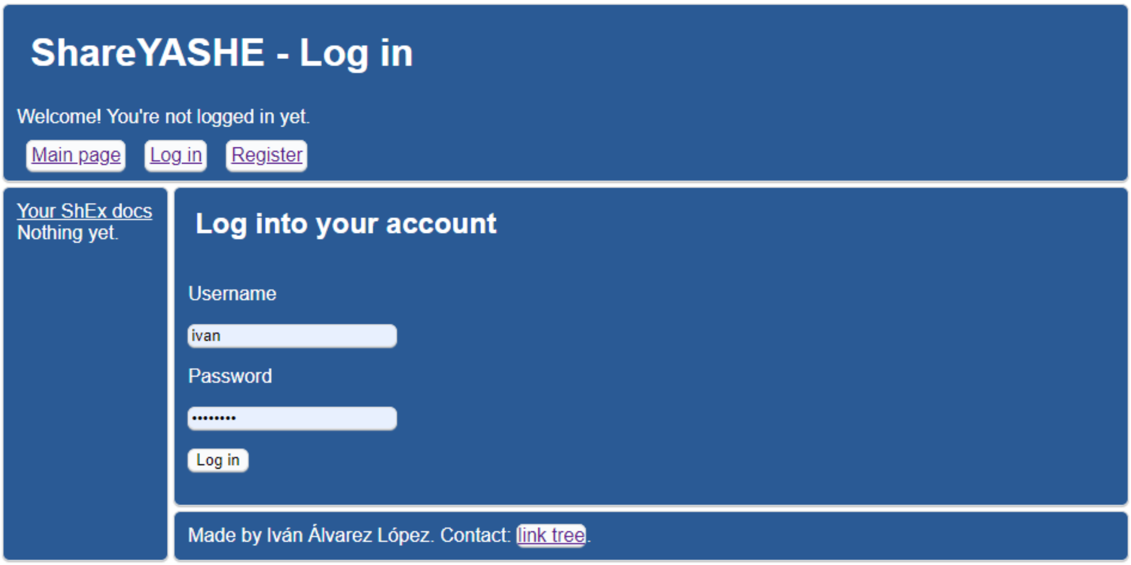
Delete your account

The deletion of your account will remove all your data from ShareYASHE. Your ownership of your ShEx docs will disappear. Are you sure that you want to delete your account?

Made by Iván Álvarez López. Contact: link tree.

Figure 34. ShareYASHE unregister view layout.

4.7.3.2.5 Log-in view



ShareYASHE - Log in

Welcome! You're not logged in yet.

Nothing yet.

Log into your account

Username

ivan

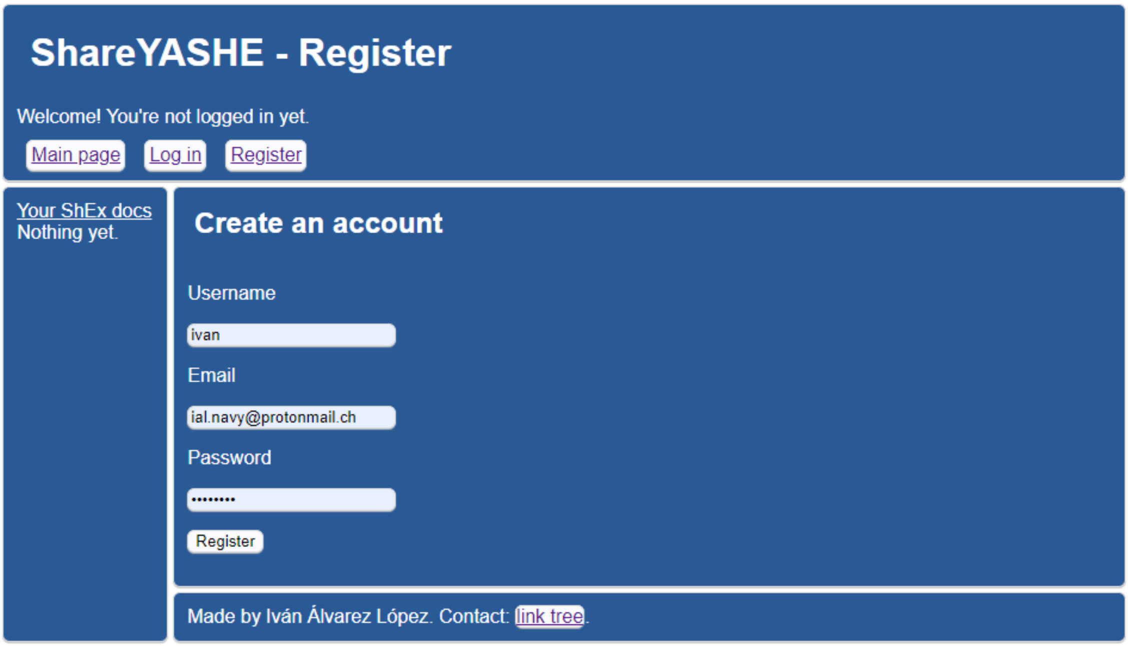
Password

••••••••

Made by Iván Álvarez López. Contact: link tree.

Figure 35. ShareYASHE log-in view layout.

4.7.3.2.6 Register view



ShareYASHE - Register

Welcome! You're not logged in yet.

Nothing yet.

Create an account

Username

ivan

Email

ial.navy@protonmail.ch

Password

••••••••

Made by Iván Álvarez López. Contact: link tree.

Figure 36. ShareYASHE register view layout.

4.8 Test planning

4.8.1 Specification-based testing

In this chapter we will make a design based on the specification of the application.

4.8.1.1 Equivalence classes

These are the found equivalence classes in the ShareYASHE application.

- Username.
  - Empty.
  - Shorter than three characters.
  - Already taken.
  - Not taken.
  - Not taken and longer than three characters.
- Password.
  - Empty.
  - Shorter than three characters.
  - Longer than three characters.
  - Matches with the username.
  - Does not match with the username.
- Email.
  - Empty.
  - Not valid (RFC 5322).
  - Valid (RFC 5322).
- ShEx document title.
  - Empty.
  - Not empty.
- Ownership.
  - Not an owner.
  - Owner by creation.
  - Added as an owner.

4.8.1.2 Combination strategy

In order to combinate the equivalence classes, we are going to take a strategy of combination. In this case, I considered using a base choice for the login and registration classes (a combination of “username”, “password”, and “email”), and an each choice for the rest of equivalence classes.

4.8.1.3 Test situations

This is the result of combination the equivalence classes with the chosen strategy.

4.8.1.3.1 Scenario 1. Registration

1.	**Scenario 1. Registration.**
1.1.	All the parameters of registration are valid. • The username is longer than three characters and not taken. • The password is longer than three characters. • The email is valid (RFC 5322).
1.2.	One parameter of registration is not valid.
1.2.1.	The username is empty.
1.2.2.	The username is shorter than three characters.
1.2.3.	The username is already taken.
1.2.4.	The password is empty.
1.2.5.	The password is shorter than three characters.
1.2.6.	The email is empty.
1.2.7.	The email is not valid (RFC 5322).

Figure 37. Scenario of registration.

4.8.1.3.2 Scenario 2. Login

2.	**Scenario 2. Login.**
2.1.	All the parameters of login are valid. • The username is longer than three characters and already taken. • The password is longer than three characters and matches the username.
2.2.	One parameter of login is not valid.
2.2.1.	The username is empty.
2.2.2.	The username is shorter than three characters.
2.2.3.	The username is not taken.
2.2.4.	The password is empty.
2.2.5.	The password is shorter than three characters.
2.2.6.	The password does not match the username

Figure 38. Scenario of login.

4.8.1.3.3 Scenario 3. Create ShEx document

3.	**Scenario 3. Create ShEx document.**
3.1.	The title is empty.
3.2.	The title is not empty.

Figure 39. Scenario of create ShEx document.

4.8.1.3.4 Scenario 4. Leave ownership

4.	Scenario 4. Leave ownership.
4.1.	The user owns the ShEx document.
4.2.	The user does not own the ShEx document.
4.3.	The user is not logged in.

Figure 40. Scenario of leave ownership.

4.8.1.3.5 Scenario 5. Add owner

5.	Scenario 5. Add owner.
5.1.	The user does not own the ShEx document.
5.2.	The user owns the ShEx document.
5.2.1.	The username is empty.
5.2.2.	The username is shorter than three characters.
5.2.3.	The username is not taken.
5.2.4.	The username equals the user's username.
5.2.5.	The username is already taken by someone else.

Figure 41. Scenario of add owner.

4.8.1.4 Test cases

4.8.1.4.1 Scenario 1. Registration

TSs	ID	Description	Initial conditions	Test case steps	Expected output
-	**1.**	Scenario 1. Registration			
1.1.	**1.1.**	All the parameters of registration are valid.	There is not any user in the database.	Enter to the registration form. Type username "user". Type password "user123". Type email "user@email.com". Submit registration.	The database has one user with the data specified in the test case steps.
1.2.1.	**1.2.**	The username is empty.	There is not any user in the database.	Enter to the registration form. Do not type any username. Type password "user123". Type email "user@email.com". Submit registration.	The database is not modified.

Figure 42. Test cases for scenario of registration (1 of 3).

TSs	ID	Description	Initial conditions	Test case steps	Expected output
1.2.2.	**1.3.**	The username is shorter than three characters.	There is not any user in the database.	Enter to the registration form. Type username "us". Type password "user123". Type email "user@email.com". Submit registration.	The database is not modified.
1.2.3.	**1.4.**	The username is already taken.	There is a user in the database with the data specified in the test case steps.	Enter to the registration form. Type username "user". Type password "user123". Type email "user@email.com". Submit registration.	The database is not modified.
1.2.4.	**1.5.**	The password is empty.	There is not any user in the database.	Enter to the registration form. Type username "user". Do not type any password. Type email "user@email.com". Submit registration.	The database is not modified.
1.2.5.	**1.6.**	The password is shorter than three characters.	There is not any user in the database.	Enter to the registration form. Type username "user". Type password "us". Type email "user@email.com". Submit registration.	The database is not modified.
1.2.6.	**1.7.**	The email is empty.	There is not any user in the database.	Enter to the registration form. Type username "user". Type password "user123". Do not type any email. Submit registration.	The database is not modified.

Figure 43. Test cases for scenario of registration (2 of 3).

TSs	ID	Description	Initial conditions	Test case steps	Expected output
1.2.7.	**1.8.**	The email is not valid (RFC 5322).	There is not any user in the database.	Enter to the registration form. Type username “user”. Type password “user123”. Type email “useremail.com”. Submit registration.	The database is not modified.

Figure 44. Test cases for scenario of registration (3 of 3).

4.8.1.4.2 Scenario 2. Login

TSs	ID	Description	Initial conditions	Test case steps	Expected output
-	**2.**	Scenario 2. Login			
2.1.	**2.1.**	All the parameters of login are valid.	There is a user in the database, having username “user”, password “user123”, and email “user@email.com”.	Enter to the login form. Type username “user”. Type password “user123”. Submit login.	The database is not modified. The user logs in successfully.
2.2.1.	**2.2.**	The username is empty.	There is a user in the database, having username “user”, password “user123”, and email “user@email.com”.	Enter to the login form. Do not type any username. Type password “user123”. Submit login.	The database is not modified. The user does not log in.
2.2.2.	**2.3.**	The username is shorter than three characters.	There is a user in the database, having username “user”, password “user123”, and email “user@email.com”.	Enter to the login form. Type username “us”. Type password “user123”. Submit login.	The database is not modified. The user does not log in.
2.2.3.	**2.4.**	The username is not taken.	There is a user in the database, having username “user”, password “user123”, and email “user@email.com”.	Enter to the login form. Type username “userX”. Type password “user123”. Submit login.	The database is not modified. The user does not log in.

Figure 45. Test cases for scenario of login (1 of 2).

TSs	ID	Description	Initial conditions	Test case steps	Expected output
2.2.4.	**2.5.**	The password is empty.	There is a user in the database, having username “user”, password “user123”, and email “user@email.com”.	Enter to the login form. Type username “user”. Do not type any password. Submit login.	The database is not modified. The user does not log in.
2.2.5.	**2.6.**	The password is shorter than three characters.	There is a user in the database, having username “user”, password “user123”, and email “user@email.com”.	Enter to the login form. Type username “user”. Type password “us”. Submit login.	The database is not modified. The user does not log in.
2.2.6.	**2.7.**	The password is shorter than three characters.	There is a user in the database, having username “user”, password “user123”, and email “user@email.com”.	Enter to the login form. Type username “user”. Type password “user321”. Submit login.	The database is not modified. The user does not log in.

Figure 46. Test cases for scenario of login (2 of 2).

4.8.1.4.3 Scenario 3. Create ShEx document

TSs	ID	Description	Initial conditions	Test case steps	Expected output
-	**3.**	Scenario 3. Create ShEx document			
3.1.	**3.1.**	The title is empty.	There is a user in the database, having username “user”, password “user123”, and email “user@email.com”.	Login with username “user” and password “user123”. Enter to the ShEx document creation form. Do not type any title. Submit ShEx document creation.	The database is not modified.
3.2.	**3.2.**	The title is not empty.	There is a user in the database, having username “user”, password “user123”, and email “user@email.com”.	Login with username “user” and password “user123”. Enter to the ShEx document creation form. Type title “Example”. Submit ShEx document creation.	There is a new ShEx document in the database with title “Example”, and one owner “user”.

Figure 47. Test cases for scenario of create ShEx document.

4.8.1.4.4 Scenario 4. Leave ownership

TSs	ID	Description	Initial conditions	Test case steps	Expected output
-	**4.**	Scenario 4. Leave ownership			
4.1.	**4.1.**	The user owns the ShEx document.	There is a user in the database, having username “user1”, password “user123”, and email “user1@email.com”. There is a user in the database, having username “user2”, password “user123”, and email “user2@email.com”. There is a ShEx document in the database, having title “Example”, and two owners: “user1” and “user2”.	Login with username “user2” and password “user123”. Enter to the edition of ShEx document with title “Example”. Use the “leave ownership” option.	In the database, the ShEx document with title “Example” has now only one owner: “user1”.
4.2.	**4.2.**	The user does not own the ShEx document.	There is a user in the database, having username “user1”, password “user123”, and email “user1@email.com”. There is a user in the database, having username “user2”, password “user123”, and email “user2@email.com”. There is a ShEx document in the database, having title “Example”, and one owner: “user1”.	Login with username “user2” and password “user123”. Enter to the edition of ShEx document with title “Example”. Use the “leave ownership” option.	The database is not modified.

Figure 48. Test cases for scenario of leave ownership (1 of 2).

TSs	ID	Description	Initial conditions	Test case steps	Expected output
4.3.	**4.3.**	The user is not logged in.	There is a user in the database, having username “user1”, password “user123”, and email “user1@email.com”. There is a ShEx document in the database, having title “Example”, and one owner: “user1”.	Enter to the edition of ShEx document with title “Example”. Use the “leave ownership” option.	The database is not modified.

Figure 49. Test cases for scenario of leave ownership (2 of 2).

4.8.1.4.5 Scenario 5. Add owner

TSs	ID	Description	Initial conditions	Test case steps	Expected output
-	**5.**	Scenario 5. Add owner			
5.1.	**5.1.**	The user does not own the ShEx document.	There is a user in the database, having username “user1”, password “user123”, and email “user1@email.com”. There is a user in the database, having username “user2”, password “user123”, and email “user2@email.com”. There is a user in the database, having username “user3”, password “user123”, and email “user3@email.com”. There is a ShEx document in the database, having title “Example”, and one owner: “user1”.	Login with username “user2” and password “user123”. Enter to the edition of ShEx document with title “Example”. Use the “add owner” option for adding the user with username “user3”.	The database is not modified.

Figure 50. Test cases for scenario of add owner (1 of 4).

TSs	ID	Description	Initial conditions	Test case steps	Expected output
5.2.1.	**5.2.**	The user owns the ShEx document, but the username is empty.	There is a user in the database, having username “user1”, password “user123”, and email “user1@email.com”. There is a user in the database, having username “user2”, password “user123”, and email “user2@email.com”. There is a ShEx document in the database, having title “Example”, and one owner: “user1”.	Login with username “user1” and password “user123”. Enter to the edition of ShEx document with title “Example”. Use the “add owner” option, and do not type any username.	The database is not modified.
5.2.2.	**5.3.**	The user owns the ShEx document, but the username is shorter than three characters.	There is a user in the database, having username “user1”, password “user123”, and email “user1@email.com”. There is a user in the database, having username “user2”, password “user123”, and email “user2@email.com”. There is a ShEx document in the database, having title “Example”, and one owner: “user1”.	Login with username “user1” and password “user123”. Enter to the edition of ShEx document with title “Example”. Use the “add owner” option and type the username “us”.	The database is not modified.

Figure 51. Test cases for scenario of add owner (2 of 4).

TSs	ID	Description	Initial conditions	Test case steps	Expected output
5.2.3.	**5.4.**	The user owns the ShEx document, but the username is not taken by anybody.	There is a user in the database, having username “user1”, password “user123”, and email “user1@email.com”. There is a user in the database, having username “user2”, password “user123”, and email “user2@email.com”. There is a ShEx document in the database, having title “Example”, and one owner: “user1”.	Login with username “user1” and password “user123”. Enter to the edition of ShEx document with title “Example”. Use the “add owner” option and type the username “user3”.	The database is not modified.
5.2.4.	**5.5.**	The user owns the ShEx document, but the username is the current owner’s username.	There is a user in the database, having username “user1”, password “user123”, and email “user1@email.com”. There is a user in the database, having username “user2”, password “user123”, and email “user2@email.com”. There is a ShEx document in the database, having title “Example”, and one owner: “user1”.	Login with username “user1” and password “user123”. Enter to the edition of ShEx document with title “Example”. Use the “add owner” option and type the username “user1”.	The database is not modified.

Figure 52. Test cases for scenario of add owner (3 of 4).

TSs	ID	Description	Initial conditions	Test case steps	Expected output
5.2.5.	**5.6.**	The user owns the ShEx document, and the username added as owner is correct.	There is a user in the database, having username "user1", password "user123", and email "user1@email.com". There is a user in the database, having username "user2", password "user123", and email "user2@email.com". There is a ShEx document in the database, having title "Example", and one owner: "user1".	Login with username "user1" and password "user123". Enter to the edition of ShEx document with title "Example". Use the "add owner" option and type the username "user2".	In the database, the ShEx document with title "Example" has now two owners: "user1" and "user2".

Figure 53. Test cases for scenario of add owner (4 of 4).

4.8.2 Usability testing

For the usability testing, the following tasks will be performed by a given group of users.

1. Register an account, login, and create a ShEx document.
    a. Register an account.
    b. Login into that account.
    c. Create a ShEx document.
2. Share a ShEx document and edit it collaboratively.
    a. Login into the account that the user registered in task 1.
    b. Access to the ShEx document that the user created in task 1.
    c. Share the ShEx document with another user.
    d. Both users, edit the ShEx document at the same time.

The first task will be performed individually, while the second task will be made in pairs.

4.8.3 Accessibility testing

Some aspects of accessibility may be taken into consideration in the usability testing. However, accessibility is meant to grant access in our application to every user, regardless of its conditions. Colour-blinded people must be able to use our application, people who is only able to use computers through the keyboard must be able to use our application, etc., otherwise our application will not be accessible.

Because of this, in the accessibility testing we will be taking advantage of tools of automation of accessibility testing for ensuring a WCAG 2.0 AAA accessibility level. These tools are listed below.

- W3C HTML Markup Validation Service[W3CHTML23].
- W3C CSS Validation Service[W3CCSS23].
- Google Mobile Optimization Checker[GAdapCheck23].
- Web Accessibility Evaluation Tool[WAVE23].
- aChecker[ACH23].
- Test of Web Accessibility[TAW23].
- WebAIM Contrast Checker[WAIMCC23].

We will be testing the concrete views which are public: main view, ShEx document editing view, log-in view, and register view, focusing specially in the ShEx document editing view, because of its YASHE editor, which may be a hotspot for accessibility problems. The other views shall present a strong reliability in this concern, because of our strict policy in sticking to the W3C specifications.

I encountered a problem with WAVE having difficulties to import the ShareYASHE CSS stylesheets. This is why I am using WebAIM Contrast Checker to grant enough colour contrast.

5 System development

5.1 Technologies

In this chapter, we will see the technologies which were used to carry this project out.

5.1.1 ECMAScript ES6 (JavaScript)

Figure 54. JavaScript ES6 logo.

The core of the functionality of ShareYASHE was made using the JavaScript language. It was developed using the **ECMAScript ES6 specification of June 2015**.

It's important to note that ES6 is a specific version of the ECMAScript specification, while JavaScript is a programming language that implements that specification. JavaScript evolves over time and new versions are released, introducing several changes.

This is the reason why I considered worth it to use the well stablished ECMAScript ES6 version of the specification. This technology was used both in the ShareYASHE server (server-side), and the ShareYASHE client (client-side).

5.1.2 Node.js

Figure 55. Node.js logo.

Node.js is an open-source, **server-side JavaScript runtime environment** that allows developers to run JavaScript code outside of a web browser. It uses the V8 JavaScript engine, which is the same engine that powers the Google Chrome browser, to execute JavaScript code on the server side.

Node.js provides an event-driven, non-blocking I/O model, making it **highly efficient and well-suited for building scalable and real-time applications**. It has a rich ecosystem of modules and libraries available through its package manager, npm, which allows developers to easily leverage existing code to build their applications.

For the ShareYASHE server, both HTTP and WebSocket, I used Node.js v18.15.0. Node.js v18, codenamed “Hydrogen”, has been designated as an LTS version on 25th October 2022, being the current LTS version in the moment of developing this project.

This means that the Node.js distribution that ShareYASHE uses will be supported with active maintenance and updates for an extended period. LTS versions typically receive bug fixes, security updates, and stability improvements, making them suitable for production environments requiring long-term stability and support.

5.1.3 Express

Express is a popular web application framework for Node.js, providing a minimalistic and flexible set of features for building web applications and APIs. It is designed to simplify the process of developing server-side applications in Node.js by providing a robust set of utilities and middleware.

Figure 56. Node.js Express logo.

Express is unopinionated, which means it does not impose a specific structure or architecture on your application. It allows developers to have greater control and freedom in designing their application's structure and logic. This flexibility makes Express suitable for a wide range of applications, from small projects to large-scale, enterprise-level systems.

In the case of ShareYASHE, we will use Express for routing different HTTP methods and URLs.

5.1.4 Yjs framework

The Yjs project is an open-source collaboration framework, created by Kevin Jahns[Jahns14], that enables real-time collaboration in web applications. It provides a simple and efficient way to synchronize shared data across multiple clients in real-time, allowing users to collaborate and edit content simultaneously.

Figure 57. The Yjs framework logo.

Yjs stands for "Y (Yup)", which represents the data structure used by the framework. Yjs utilizes a data type called "CRDT" (Conflict-Free Replicated Data Type) to handle concurrent changes made by different users. CRDTs are designed to ensure that data remains consistent across distributed systems without the need for centralized coordination or conflict resolution.

The Yjs project provides client-side libraries for JavaScript and has implementations in various programming languages, making it accessible for developers working on different platforms.

In ShareYASHE, our Shape Expressions documents’ content will be of this CRDT type.

5.1.5 WebSocket (y-websocket)

Figure 58. WebSocket logo.

WebSockets is a communication protocol that provides full-duplex communication channels over a single TCP connection between a client and a server. It enables real-time, bidirectional communication between a web browser (or any WebSocket-enabled client) and a server.

Unlike traditional HTTP communication, which follows a request-response model, WebSockets allow for ongoing, persistent connections where both the client and the server can send data to each other at any time. This real-time, event-driven nature makes WebSockets well-suited for applications that require instant data updates, such as chat applications, real-time collaboration tools, live dashboards, multiplayer games, and more.

The use of the WebSocket technology resulted crucial in the development of ShareYASHE. This technology perfectly suits our necessity of stablishing a Shape Expressions collaborative editor in real-time. Both the ShareYASHE server of WebSocket, and the ShareYASHE client will be supporting this technology for exchanging the real-time updates of any Shape Expressions document that some users could be editing.

There is an extension of the Yjs framework that simplifies creating a WebSocket publish-subscribe model for the CRDT object, called y-websocket, which ShareYASHE will be using.

5.1.6 Codemirror (y-codemirror)

Figure 59. CodeMirror logo.

Codemirror is an open-source text editor component for web applications. It provides a versatile and customizable code editing experience with support for syntax highlighting, code folding, auto-completion, linting, and many other features commonly found in code editors.

Developed in JavaScript, CodeMirror is designed to be easily integrated into web applications, offering a flexible API that allows developers to tailor the editor's behavior and appearance to their specific needs. It supports a wide range of programming languages and file formats, making it suitable for various coding environments and purposes.

Specifically, there is a collaborative code editor built on top of the Yjs framework, which is called y-codemirror, and whose supporting Software is both Codemirror and Yjs. It combines the functionality of the CodeMirror text editor with the real-time collaboration capabilities provided by Yjs. This allows multiple users to edit code simultaneously and see each other's changes in real-time.

The editor of ShareYASHE will be using the functionality of y-codemirror.

5.1.7 YASHE

YASHE is a ShEx editor which started as a fork of YASQE (which is based on SPARQL) This tool performs lexical and syntactic analysis of the content of the editor, thus offering the user a realtime syntactic error detector. It has features like: syntax highlighting, visual aid elements (tooltips) and autocomplete mechanisms. In addition, it offers a simple way of integrating into other projects.

Figure 60. YASHE banner.

YASHE is registered as a Node.js package (npm), so it can be used in any application that runs on Node.js. ShareYASHE is conceived to be the direct evolution of YASHE, implementing the collaborativity addition.

5.1.8 MongoDB (y-mongodb-provider)

MongoDB is a popular open-source, NoSQL database management system that falls under the category of document-oriented databases. It is designed to provide high performance, scalability, and flexibility for handling large volumes of structured, semi-structured, and unstructured data.

Figure 61. MongoDB logo.

MongoDB is widely used in various applications, ranging from small-scale projects to enterprise-level systems. It is particularly well-suited for use cases such as content management, real-time analytics, cataloguing, logging, and user data management.

The Yjs framework provides an extension called y-mongodb-provider, which lets us to save in real-time the CRDT type objects into our MongoDB database. This is why I used MongoDB for the persistence because this extension prevents the lose of changes when the application closes, or it crashes.

5.1.9 dotenv

dotenv is a popular npm package used in Node.js applications to manage environment variables. It simplifies the process of loading configuration settings from environment-specific files into the application's runtime environment.

ShareYASHE will be using dotenv for declaring references that could possibly change in the future, such as the URI reference to the MongoDB, which currently is a local MongoDB database, or the entropy key for encrypting passwords.

5.1.10 body-parser

body-parser is a middleware package for Node.js applications, specifically designed for parsing HTTP request bodies. It simplifies the process of extracting data from incoming requests by handling various content types and providing a convenient interface for accessing the parsed data.

In Node.js, when a client sends an HTTP request with a payload (such as form data or JSON data), the request body contains that data. However, the raw request body is typically received as a stream of bytes. Parsing and extracting meaningful data from the request body requires additional processing.

body-parser eliminates the need for manual parsing of request bodies by providing middleware that automatically parses the request body and makes it accessible in a convenient format. It supports different content types, including JSON, URL-encoded, and multipart form data. ShareYASHE will be using body-parser for this purpose.

5.1.11 Pug.js

Pug.js, formerly known as Jade, is a high-performance, feature-rich template engine for Node.js and web browsers. It is designed to simplify the process of writing HTML markup by providing a concise and expressive syntax.

Figure 62. Pug.js logo.

Pug.js allows you to create HTML templates using indentation-based syntax instead of explicit opening and closing tags, similarly to Python. This results in cleaner and more readable code.

Pug.js has gained popularity due to its simplicity, readability, and powerful features. It is widely adopted in the Node.js ecosystem and is used in many web frameworks and applications.

All the views in ShareYASHE are rendered using Pug.js.

5.2 Tools and programs

5.2.1 Git

Git is a widely used distributed version control system (VCS) designed to track changes to files and manage software development projects. It provides a way for multiple developers to collaborate on a project, making it easier to manage code, track modifications, and handle different versions of files.

Figure 63. Git logo.

The code of ShareYASHE will be versioned using Git.

5.2.2 GitHub

GitHub is a web-based platform and hosting service for software development projects that use the Git version control system. It provides a collaborative environment for developers to work on code, track changes, manage projects, and collaborate with others.

The code of ShareYASHE is published in a GitHub repository[Alv23] as an open-source project.

Figure 64. GitHub logo.

5.2.3 PowerShell 7

PowerShell 7 is an open-source, cross-platform command-line shell and scripting language developed by Microsoft. It is the successor to Windows PowerShell, designed to bring the power and flexibility of PowerShell to various operating systems, including Windows, macOS, and Linux.

Figure 65. PoweShell 7 logo.

PowerShell 7 expands the reach of PowerShell beyond the Windows ecosystem, allowing users to leverage its capabilities in multi-platform environments. It provides a powerful command-line shell and scripting language that enables system administrators, developers, and IT professionals to manage and automate tasks efficiently across different operating systems.

I will be using PowerShell 7 for deploying ShareYASHE locally, with Node.js, for testing purposes, although ShareYASHE can be deployed in any machine that supports Node.js.

5.2.4 PlantUML

PlantUML is an open-source tool for creating diagrams using a simple and intuitive textual syntax. It allows developers, architects, and designers to express ideas and concepts through a text-based representation of diagrams, which are then rendered into visual diagrams.

Figure 66. PlantUML logo.

PlantUML provides a convenient and efficient way to create diagrams using a simple text-based syntax. It is widely used in software development, system design, documentation, and other fields where visual representation of concepts and ideas is required.

This tool was used to draw the diagrams in the chapter of Software design and architecture.

5.2.5 IntelliJ IDEA

IntelliJ IDEA, commonly referred to as IntelliJ, is a popular integrated development environment (IDE) created by JetBrains. It is designed to provide a comprehensive and productive environment for software development in various programming languages, including Java, Kotlin, JavaScript, TypeScript, Python, and more.

Figure 67. IntelliJ IDEA logo.

IntelliJ is widely regarded as one of the most feature-rich and powerful IDEs available. It caters to a wide range of programming languages and provides a comprehensive set of tools to support developers throughout the entire software development lifecycle.

This is the reason why I began the ShareYASHE project using IntelliJ IDEA, because I thought that it could serve for depuration purposes, although I finally ended using the following IDE.

5.2.6 Visual Studio Code

Visual Studio Code (VS Code) is a free and open-source source code editor developed by Microsoft. It is designed to provide a lightweight yet powerful editing environment for developers working on various programming languages and platforms.

Figure 68. Visual Studio Code logo.

Visual Studio Code has gained significant popularity among developers due to its combination of performance, versatility, and extensibility.

I switched to VS Code because it offers a PlantUML extension. At the end, I realised that switching IDEs did not make a difference, so I stayed in VS Code.

5.2.7 GanttProject

GanttProject is a free and open-source project management tool specifically designed to create and manage Gantt charts. It provides a comprehensive set of features for planning, scheduling, and tracking projects.

GanttProject is a valuable tool for project managers, teams, and individuals seeking a straightforward and accessible solution for creating and managing Gantt charts. It offers essential project management features, allowing users to plan, visualize, and track project schedules effectively.

Figure 69. GanttProject logo.

Being an open-source tool, it is free to use and benefits from contributions and enhancements from a community of users and developers. Despite having a license of Microsoft Project from my account of the University of Oviedo, I chose GanttProject because of this. I wanted to get some experience with this open-source tool, which resulted very helpful.

GanttProject also offers a collaborative way of editing Gantt charts, through an extension called GanttProject Cloud[GanPr20].

6 Test development

6.1 Reporting of the specification-based testing

These are the results of performing the specification-based testing. For each test case, we will see the output of the test. Note that the ID column represents the ID of the test case. The result column refers to whether the expected output of the test case was successfully given.

6.1.1 Scenario 1. Registration

ID	Description	Result
1.1.	All the parameters of registration are valid.	**PASSED**
1.2.	The username is empty.	**PASSED**
1.3.	The username is shorter than three characters.	**PASSED**
1.4.	The username is already taken.	**PASSED**
1.5.	The password is empty.	**PASSED**
1.6.	The password is shorter than three characters.	**PASSED**
1.7.	The email is empty.	**PASSED**
1.8.	The email is not valid (RFC 5322).	**PASSED**

Figure 70. Results of test for scenario of registration.

6.1.2 Scenario 2. Login

ID	Description	Result
2.1.	All the parameters of login are valid.	**PASSED**
2.2.	The username is empty.	**PASSED**
2.3.	The username is shorter than three characters.	**PASSED**
2.4.	The username is not taken.	**PASSED**
2.5.	The password is empty.	**PASSED**
2.6.	The password is shorter than three characters.	**PASSED**
2.7.	The password is shorter than three characters.	**PASSED**

Figure 71. Results of test for scenario of login.

6.1.3 Scenario 3. Create ShEx document

ID	Description	Result
3.1.	The title is empty.	**PASSED**
3.2.	The title is not empty.	**PASSED**

Figure 72. Results of test for scenario of create ShEx document.

6.1.4 Scenario 4. Leave ownership

ID	Description	Result
4.1.	The user owns the ShEx document.	**PASSED**
4.2.	The user does not own the ShEx document.	**PASSED**
4.3.	The user is not logged in.	**PASSED**

Figure 73. Results of test for scenario of leave ownership.

6.1.5 Scenario 5. Add owner

ID	Description	Result
5.1.	The user does not own the ShEx document.	**PASSED**
5.2.	The user owns the ShEx document, but the username is empty.	**PASSED**
5.3.	The user owns the ShEx document, but the username is shorter than three characters.	**PASSED**
5.4.	The user owns the ShEx document, but the username is not taken by anybody.	**PASSED**
5.5.	The user owns the ShEx document, but the username is the current owner's username.	**PASSED**
5.6.	The user owns the ShEx document, and the username added as owner is correct.	**PASSED**

Figure 74. Results of test for scenario of add owner.

6.2 Usability testing

6.2.1 Results of task 1

In this task, each user will have to accomplish the following steps.

1. Register an account.
2. Login into that account.
3. Create a ShEx document.

There were four users, performing the task individually, and the results of the usability testing are these.

	User 1	User 2	User 3	User 4
Age	46	13	22	34
Level of expertise in informatics (out of 10)	3	6	8	5
Time spent in step 1	50s	24s	12s	42s
Time spent in step 2	30s	13s	5s	20s
Time spent in step 3	56s	32s	22s	46s
User’s appraisal of ShareYASHE’s usability (out of 10)	8	9	10	7

Figure 75. Results of task 1 of usability testing.

6.2.2 Results of task 2

In this task, each user will have to accomplish the following steps.

1. Login into the account that the user registered in task 1.
2. Access to the ShEx document that the user created in task 1.
3. Share the ShEx document with another user.
4. Both users, edit the ShEx document at the same time.

There were four users, performing the task in pairs, and the results of the usability testing are these.

	User 1	User 2	User 3	User 4
Age	46	13	22	34
Level of expertise in informatics (out of 10)	3	6	8	5
Time spent in step 1	1m 30s	34s	19s	1m 12s
Time spent in step 2	23s	12s	6s	17s
Time spent in step 3	3m 23s	1m 49s	1m 20s	2m 43s
Time spent in step 4	34s	22s	19s	31s
User's appraisal of ShareYASHE's usability (out of 10)	8	9	10	7

Figure 76. Results of task 2 of usability testing.

6.3 Accessibility testing

6.3.1 Summary

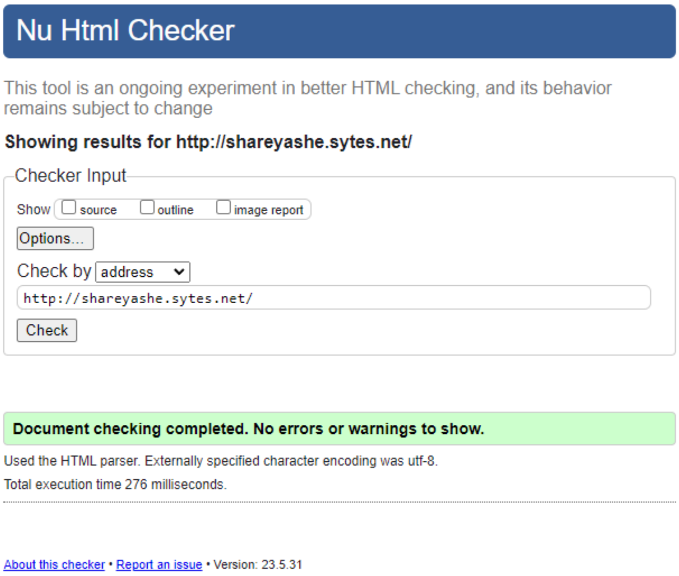
In this chapter we will see the results of the accessibility testing, which are summarised below.

- W3C HTML Markup Validation. The whole Website has passed this test.
- W3C CSS Validation. The whole Website has passed this test.
- Google Mobile Optimization Check. The whole Website has passed this test.
- WAVE Test. The whole Website has passed this test.
- aChecker Test. The whole Website has passed this test.
- TAW Test. We can see that the ShEx document editing view did not fully passed this test. This has the following two explanations.
    - TAW follows WCAG 2.1, while the purpose of this project is to achieve WCAG 2.0.
    - The errors of accessibility that TAW finds in our Website are related to the YASHE editor, which is an external library, and modifying it by ourselves exceeds the scope of this project.
- WebAIM Contrast Check. The whole Website has passed this test.

6.3.2 Results

6.3.2.1 HTML Markup Validation

6.3.2.1.1 Main view



Nu Html Checker

This tool is an ongoing experiment in better HTML checking, and its behavior remains subject to change

Showing results for http://shareyashe.sytes.net/

Checker Input

Show □ source □ outline □ image report

Options...

Check by address

http://shareyashe.sytes.net/

Check

Document checking completed. No errors or warnings to show.

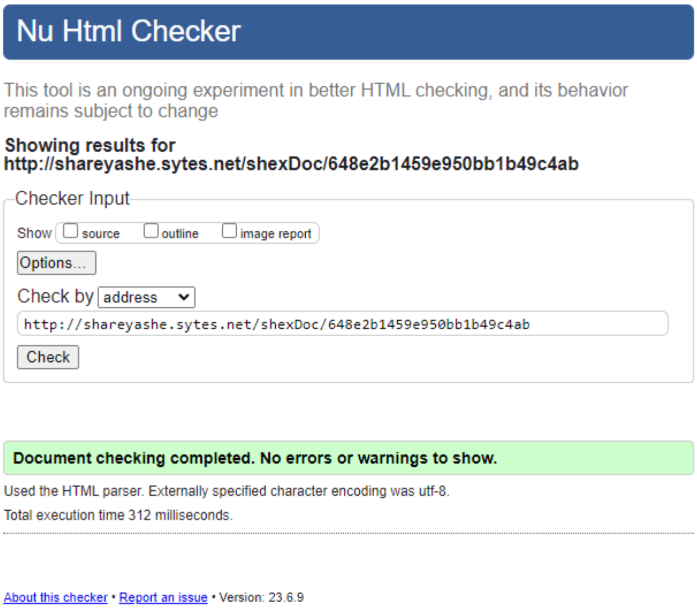
Used the HTML parser. Externally specified character encoding was utf-8.

Total execution time 276 milliseconds.

About this checker • Report an issue • Version: 23.5.31

Figure 77. Results of HTML markup validation for main view.

6.3.2.1.2 ShEx document editing view



Nu Html Checker

This tool is an ongoing experiment in better HTML checking, and its behavior remains subject to change

Showing results for http://shareyashe.sytes.net/shexDoc/648e2b1459e950bb1b49c4ab

Checker Input

Show □ source □ outline □ image report

Options...

Check by address

http://shareyashe.sytes.net/shexDoc/648e2b1459e950bb1b49c4ab

Check

Document checking completed. No errors or warnings to show.

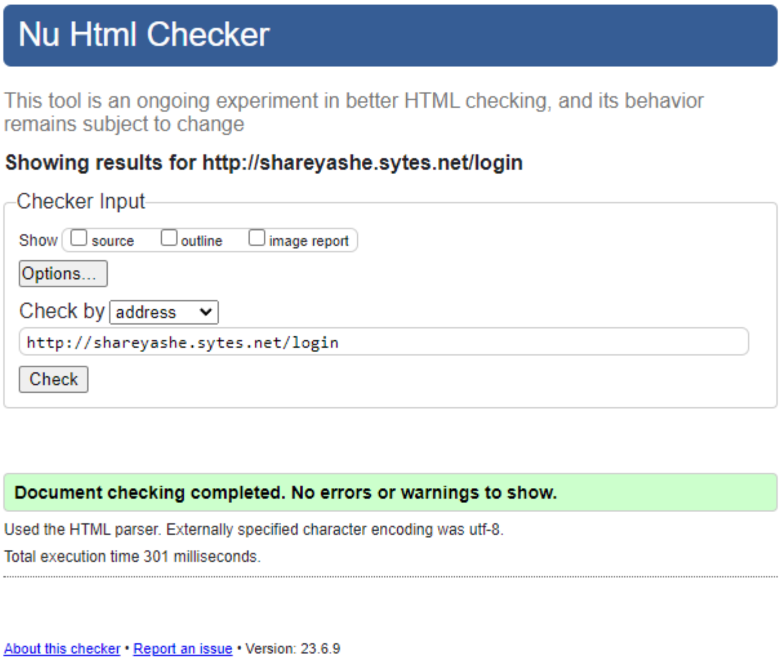
Used the HTML parser. Externally specified character encoding was utf-8.

Total execution time 312 milliseconds.

About this checker • Report an issue • Version: 23.6.9

Figure 78. Results of HTML markup validation for ShEx document editing view.

6.3.2.1.3 Log-in view



Nu Html Checker

This tool is an ongoing experiment in better HTML checking, and its behavior remains subject to change

Showing results for http://shareyashe.sytes.net/login

Checker Input

Show □ source □ outline □ image report

Options...

Check by address

http://shareyashe.sytes.net/login

Check

Document checking completed. No errors or warnings to show.

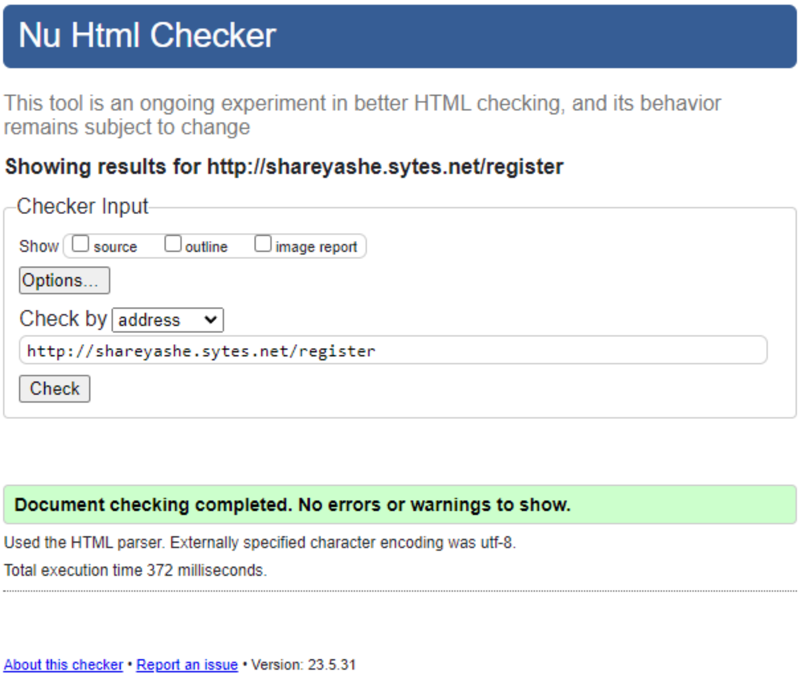
Used the HTML parser. Externally specified character encoding was utf-8.

Total execution time 301 milliseconds.

About this checker • Report an issue • Version: 23.6.9

Figure 79. Results of HTML markup validation for log-in view.

6.3.2.1.4 Register view



Nu Html Checker

This tool is an ongoing experiment in better HTML checking, and its behavior remains subject to change

Showing results for http://shareyashe.sytes.net/register

Checker Input

Show □ source □ outline □ image report

Options...

Check by address

http://shareyashe.sytes.net/register

Check

Document checking completed. No errors or warnings to show.

Used the HTML parser. Externally specified character encoding was utf-8.

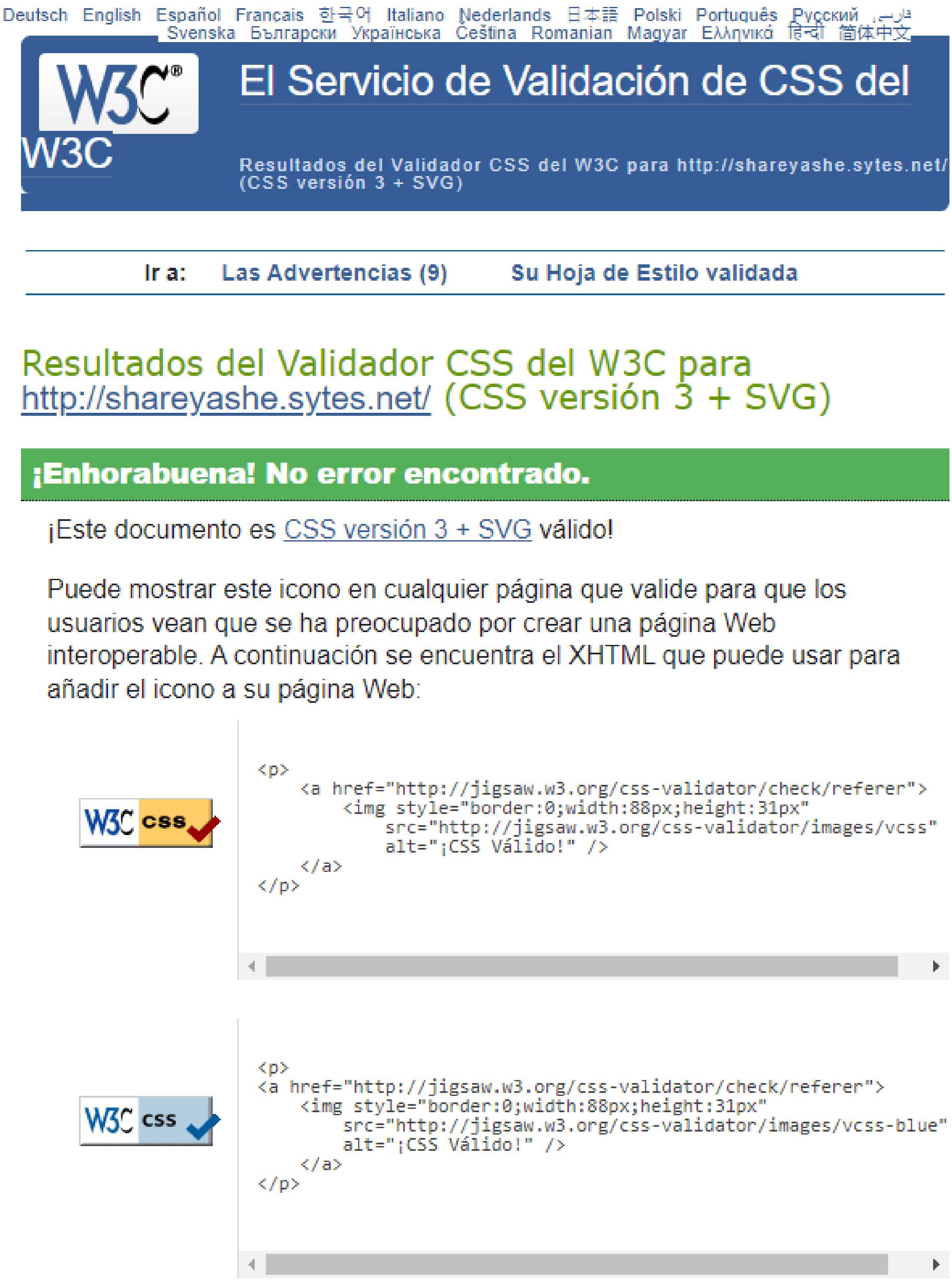
Total execution time 372 milliseconds.

About this checker • Report an issue • Version: 23.5.31

Figure 80. Results of HTML markup validation for register view.

6.3.2.2 CSS Validation

6.3.2.2.1 Main view



Deutsch English Español Francais 한국어 Italiano Nederlands 日本語 Polski Português Русский فارسی
Svenska Български Українська Čeština Romanian Magyar Ελληνικά हिन्दी 简体中文

W3C

El Servicio de Validación de CSS del W3C

Resultados del Validador CSS del W3C para http://shareyashe.sytes.net/ (CSS versión 3 + SVG)

Ir a: Las Advertencias (9) Su Hoja de Estilo validada

Resultados del Validador CSS del W3C para http://shareyashe.sytes.net/ (CSS versión 3 + SVG)

¡Enhorabuena! No error encontrado.

¡Este documento es CSS versión 3 + SVG válido!

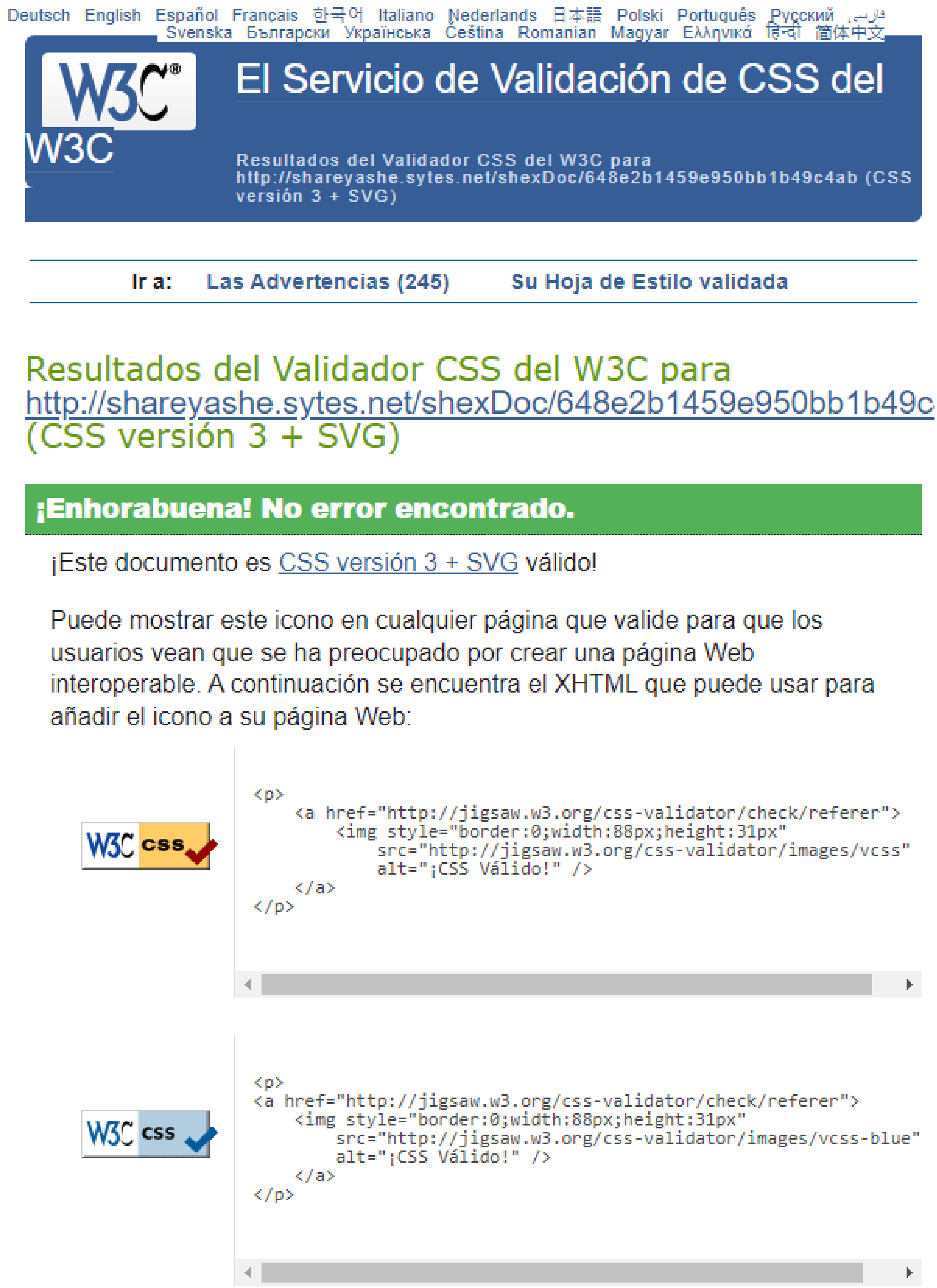
Puede mostrar este icono en cualquier página que valide para que los usuarios vean que se ha preocupado por crear una página Web interoperable. A continuación se encuentra el XHTML que puede usar para añadir el icono a su página Web:

```
<p>
    <a href="http://jigsaw.w3.org/css-validator/check/referer">
        <img style="border:0;width:88px;height:31px"
            src="http://jigsaw.w3.org/css-validator/images/vcss"
            alt="¡CSS Válido!" />
    </a>
</p>
```

```
<p>
<a href="http://jigsaw.w3.org/css-validator/check/referer">
    <img style="border:0;width:88px;height:31px"
        src="http://jigsaw.w3.org/css-validator/images/vcss-blue"
        alt="¡CSS Válido!" />
    </a>
</p>
```

Figure 81. Results of CSS validation for main view.

6.3.2.2.2 ShEx document editing view



Deutsch English Español Français 한국어 Italiano Nederlands 日本語 Polski Português Русский فارسی Svenska Български Українська Čeština Romanian Magyar Ελληνικά हिन्दी 简体中文

W3C

El Servicio de Validación de CSS del W3C

Resultados del Validador CSS del W3C para http://shareyashe.sytes.net/shexDoc/648e2b1459e950bb1b49c4ab (CSS versión 3 + SVG)

Ir a: Las Advertencias (245) Su Hoja de Estilo validada

Resultados del Validador CSS del W3C para http://shareyashe.sytes.net/shexDoc/648e2b1459e950bb1b49c (CSS versión 3 + SVG)

¡Enhorabuena! No error encontrado.

¡Este documento es CSS versión 3 + SVG válido!

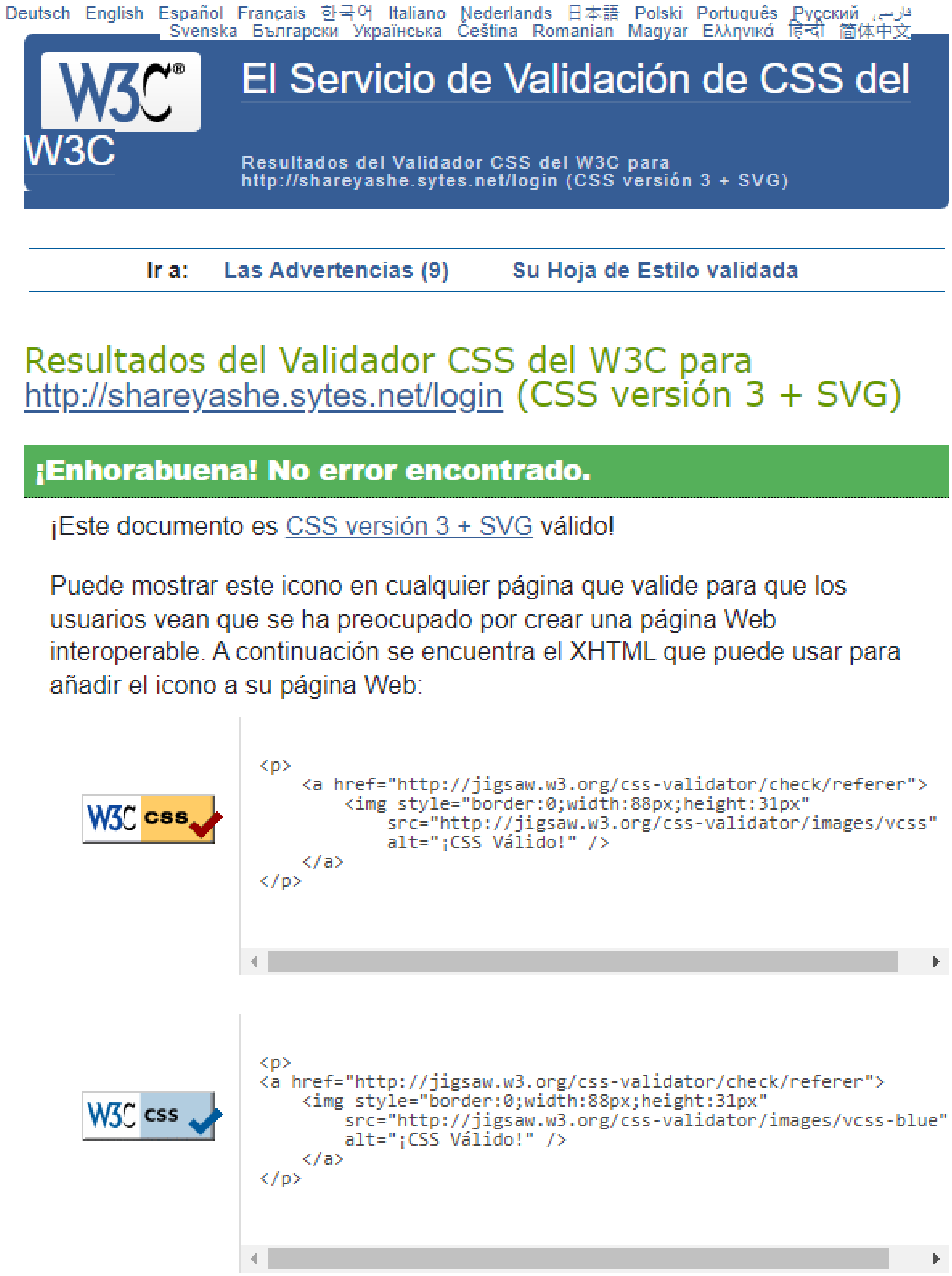
Puede mostrar este icono en cualquier página que valide para que los usuarios vean que se ha preocupado por crear una página Web interoperable. A continuación se encuentra el XHTML que puede usar para añadir el icono a su página Web:

```
<p>
    <a href="http://jigsaw.w3.org/css-validator/check/referer">
        <img style="border:0;width:88px;height:31px"
            src="http://jigsaw.w3.org/css-validator/images/vcss"
            alt="¡CSS Válido!" />
    </a>
</p>
```

```
<p>
<a href="http://jigsaw.w3.org/css-validator/check/referer">
    <img style="border:0;width:88px;height:31px"
        src="http://jigsaw.w3.org/css-validator/images/vcss-blue"
        alt="¡CSS Válido!" />
    </a>
</p>
```

Figure 82. Results of CSS validation for ShEx document editing view.

6.3.2.2.3 Log-in view



Deutsch English Español Francais 한국어 Italiano Nederlands 日本語 Polski Português Русский فارسی Svenska Български Українська Čeština Romanian Magyar Ελληνικά हिन्दी 简体中文

El Servicio de Validación de CSS del W3C

Resultados del Validador CSS del W3C para http://shareyashe.sytes.net/login (CSS versión 3 + SVG)

Ir a: Las Advertencias (9) Su Hoja de Estilo validada

Resultados del Validador CSS del W3C para http://shareyashe.sytes.net/login (CSS versión 3 + SVG)

¡Enhorabuena! No error encontrado.

¡Este documento es CSS versión 3 + SVG válido!

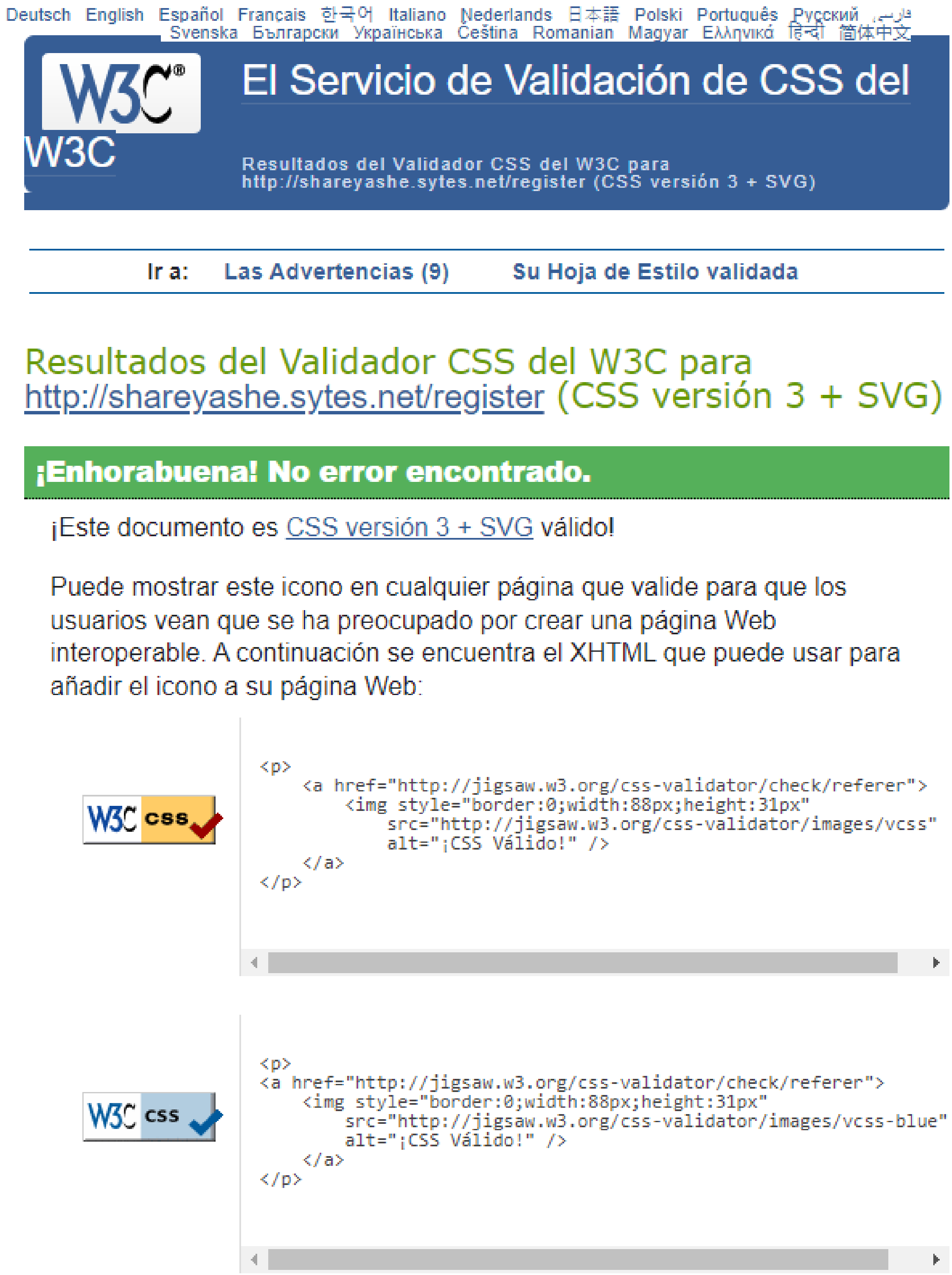
Puede mostrar este icono en cualquier página que valide para que los usuarios vean que se ha preocupado por crear una página Web interoperable. A continuación se encuentra el XHTML que puede usar para añadir el icono a su página Web:

```
<p>
    <a href="http://jigsaw.w3.org/css-validator/check/referer">
        <img style="border:0;width:88px;height:31px"
            src="http://jigsaw.w3.org/css-validator/images/vcss"
            alt="¡CSS Válido!" />
    </a>
</p>
```

```
<p>
<a href="http://jigsaw.w3.org/css-validator/check/referer">
    <img style="border:0;width:88px;height:31px"
        src="http://jigsaw.w3.org/css-validator/images/vcss-blue"
        alt="¡CSS Válido!" />
    </a>
</p>
```

Figure 83. Results of CSS validation for log-in view.

6.3.2.2.4 Register view



Deutsch English Español Francais 한국어 Italiano Nederlands 日本語 Polski Português Русский فارسی Svenska Български Українська Čeština Romanian Magyar Ελληνικά हिन्दी 简体中文

El Servicio de Validación de CSS del W3C

Resultados del Validador CSS del W3C para http://shareyashe.sytes.net/register (CSS versión 3 + SVG)

Ir a: Las Advertencias (9) Su Hoja de Estilo validada

Resultados del Validador CSS del W3C para http://shareyashe.sytes.net/register (CSS versión 3 + SVG)

¡Enhorabuena! No error encontrado.

¡Este documento es CSS versión 3 + SVG válido!

Puede mostrar este icono en cualquier página que valide para que los usuarios vean que se ha preocupado por crear una página Web interoperable. A continuación se encuentra el XHTML que puede usar para añadir el icono a su página Web:

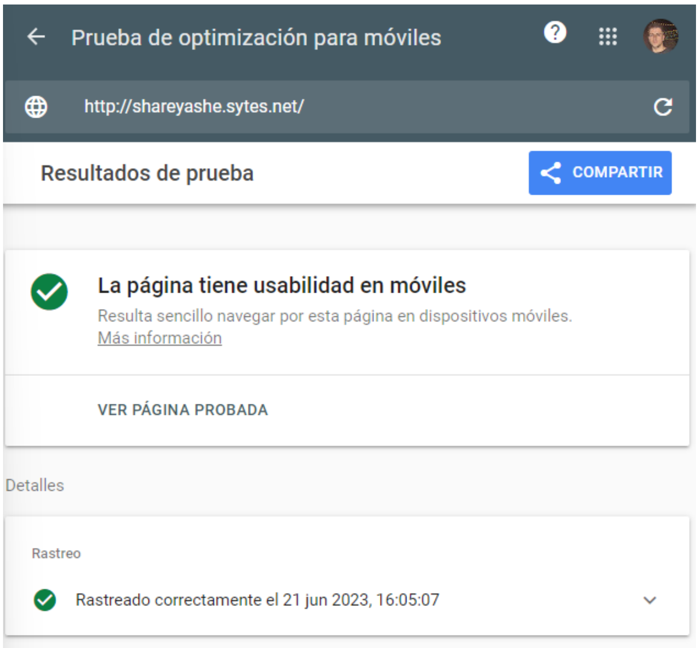
```
<p>
    <a href="http://jigsaw.w3.org/css-validator/check/referer">
        <img style="border:0;width:88px;height:31px"
            src="http://jigsaw.w3.org/css-validator/images/vcss"
            alt="¡CSS Válido!" />
    </a>
</p>
```

```
<p>
<a href="http://jigsaw.w3.org/css-validator/check/referer">
    <img style="border:0;width:88px;height:31px"
        src="http://jigsaw.w3.org/css-validator/images/vcss-blue"
        alt="¡CSS Válido!" />
    </a>
</p>
```

Figure 84. Results of CSS validation for register view.

6.3.2.3 Mobile Optimization Check

6.3.2.3.1 Main view


Prueba de optimización para móviles
http://shareyashe.sytes.net/
COMPARTIR
Más información
VER PÁGINA PROBADA

Resultados de prueba

La página tiene usabilidad en móviles

Resulta sencillo navegar por esta página en dispositivos móviles.

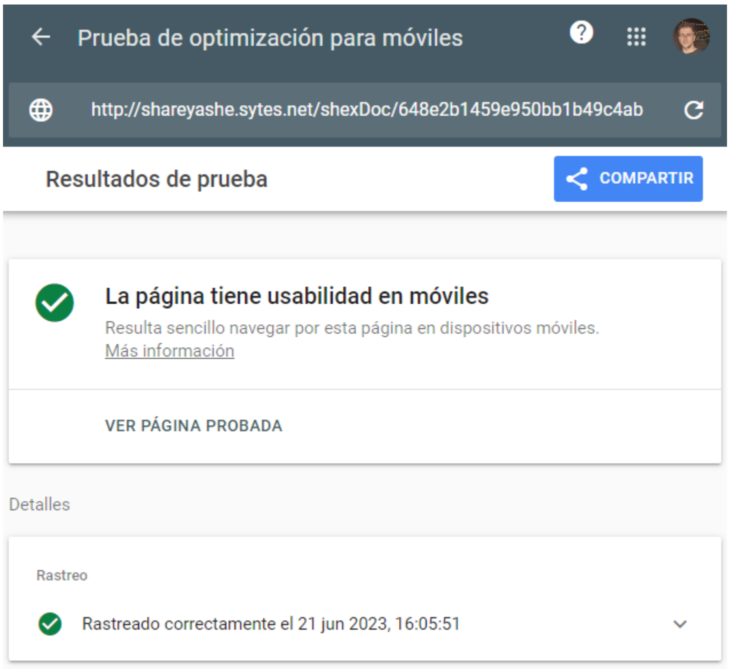
Detalles

Rastreo

Rastreado correctamente el 21 jun 2023, 16:05:07

Figure 85. Results of mobile optimization check for main view.

6.3.2.3.2 ShEx document editing view


Prueba de optimización para móviles
http://shareyashe.sytes.net/shexDoc/648e2b1459e950bb1b49c4ab
COMPARTIR
Más información
VER PÁGINA PROBADA

Resultados de prueba

La página tiene usabilidad en móviles

Resulta sencillo navegar por esta página en dispositivos móviles.

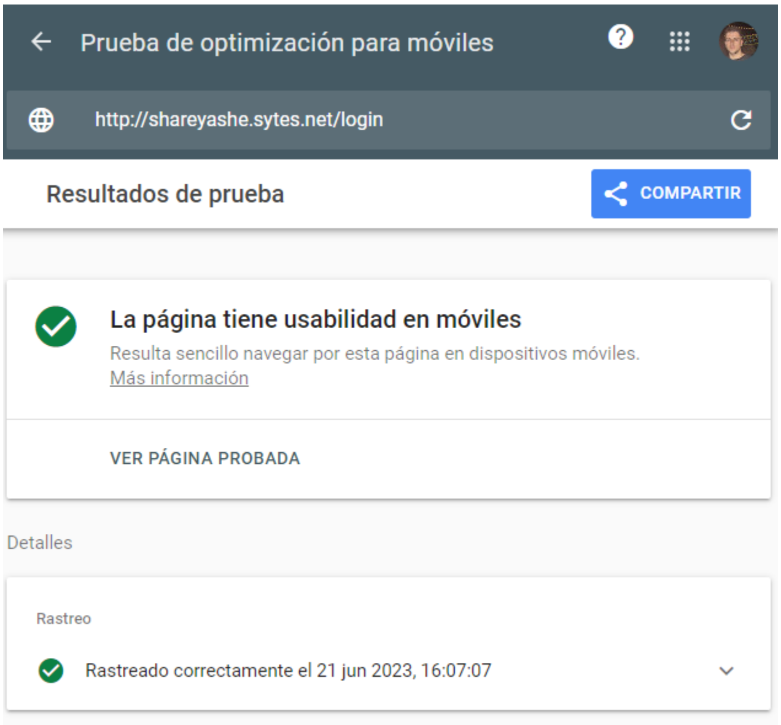
Detalles

Rastreo

Rastreado correctamente el 21 jun 2023, 16:05:51

Figure 86. Results of mobile optimization check for ShEx document editing view.

6.3.2.3.3 Log-in view



Prueba de optimización para móviles
http://shareyashe.sytes.net/login
Resultados de prueba
COMPARTIR

La página tiene usabilidad en móviles

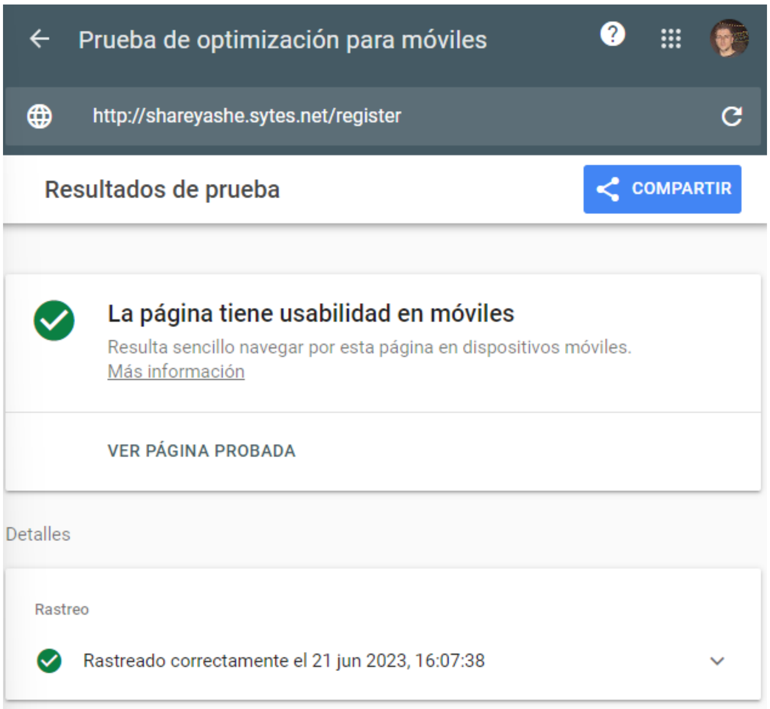
Resulta sencillo navegar por esta página en dispositivos móviles.
Más información

VER PÁGINA PROBADA
Detalles
Rastreo

✓ Rastreado correctamente el 21 jun 2023, 16:07:07

Figure 87. Results of mobile optimization check for log-in view.

6.3.2.3.4 Register view



Prueba de optimización para móviles
http://shareyashe.sytes.net/register
Resultados de prueba
COMPARTIR

La página tiene usabilidad en móviles

Resulta sencillo navegar por esta página en dispositivos móviles.
Más información

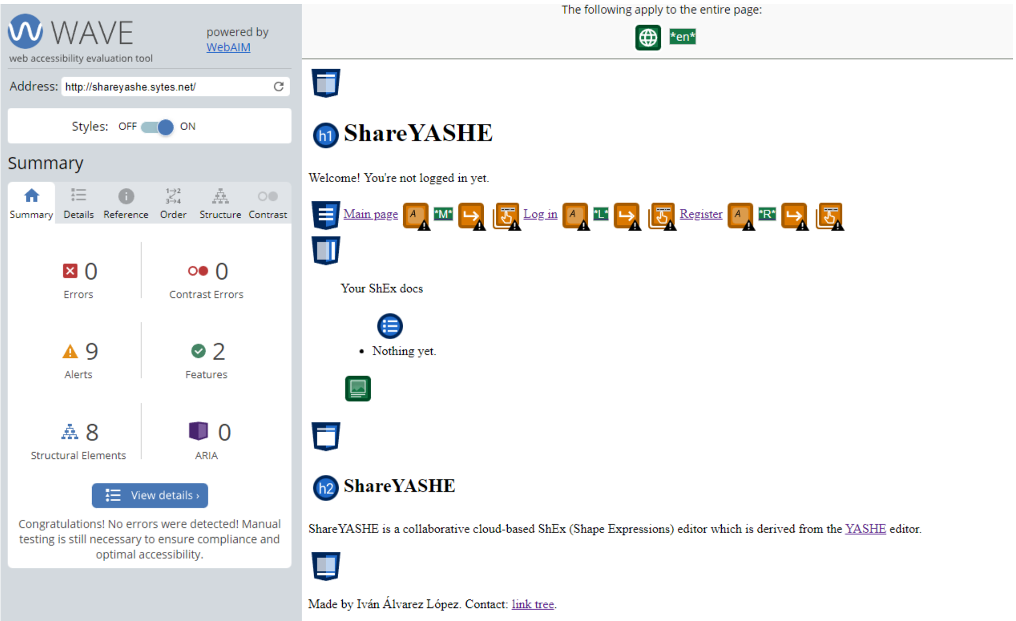
VER PÁGINA PROBADA
Detalles
Rastreo

✓ Rastreado correctamente el 21 jun 2023, 16:07:38

Figure 88. Results of mobile optimization check for register view.

6.3.2.4 WAVE test

6.3.2.4.1 Main view



Address: http://shareyashe.sytes.net/

Styles: OFF ON

Summary

0 Errors

0 Contrast Errors

9 Alerts

2 Features

8 Structural Elements

0 ARIA

Congratulations! No errors were detected! Manual testing is still necessary to ensure compliance and optimal accessibility.

The following apply to the entire page: *en*

ShareYASHE

Welcome! You're not logged in yet.

Main page "M" Log in "L" Register "R"

Your ShEx docs

- Nothing yet.

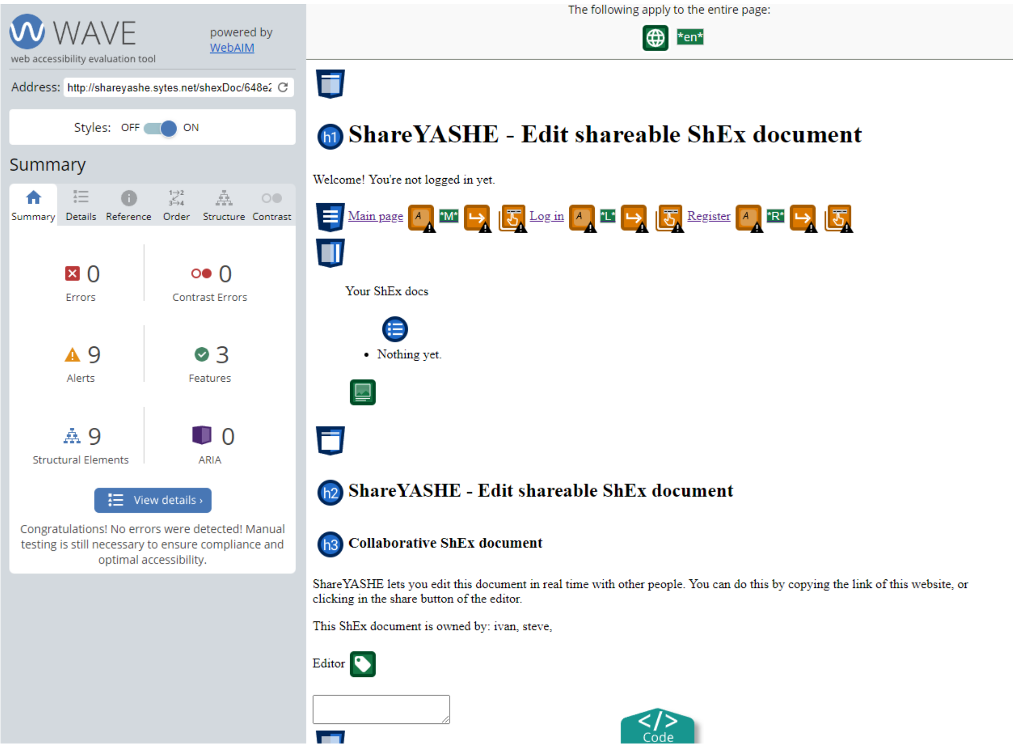
ShareYASHE

ShareYASHE is a collaborative cloud-based ShEx (Shape Expressions) editor which is derived from the YASHE editor.

Made by Iván Álvarez López. Contact: link tree.

Figure 89. Results of WAVE test for main view.

6.3.2.4.2 ShEx document editing view



Address: http://shareyashe.sytes.net/shexDoc/648e

Styles: OFF ON

Summary

0 Errors

0 Contrast Errors

9 Alerts

3 Features

9 Structural Elements

0 ARIA

Congratulations! No errors were detected! Manual testing is still necessary to ensure compliance and optimal accessibility.

The following apply to the entire page: *en*

ShareYASHE - Edit shareable ShEx document

Welcome! You're not logged in yet.

Main page "M" Log in "L" Register "R"

Your ShEx docs

- Nothing yet.

ShareYASHE - Edit shareable ShEx document

Collaborative ShEx document

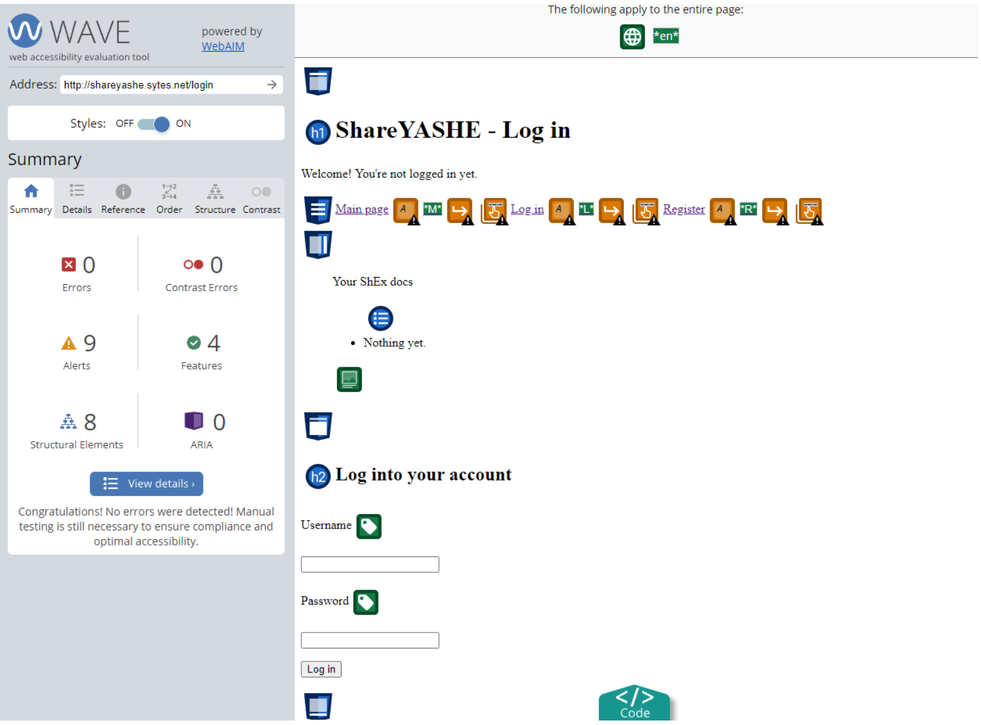
ShareYASHE lets you edit this document in real time with other people. You can do this by copying the link of this website, or clicking in the share button of the editor.

This ShEx document is owned by: ivan, steve,

Editor

Figure 90. Results of WAVE test for ShEx document editing view.

6.3.2.4.3 Log-in view


WAVE
web accessibility evaluation tool
powered by WebAIM
Styles: OFF ON
Summary Details Reference Order Structure Contrast
View details ›
en
M
L
R
Log in
Code

Address: http://shareyashe.sytes.net/login

Summary

0 Errors

0 Contrast Errors

9 Alerts

4 Features

8 Structural Elements

0 ARIA

Congratulations! No errors were detected! Manual testing is still necessary to ensure compliance and optimal accessibility.

The following apply to the entire page:

ShareYASHE - Log in

Welcome! You're not logged in yet.

Main page Log in Register

Your ShEx docs

- Nothing yet.

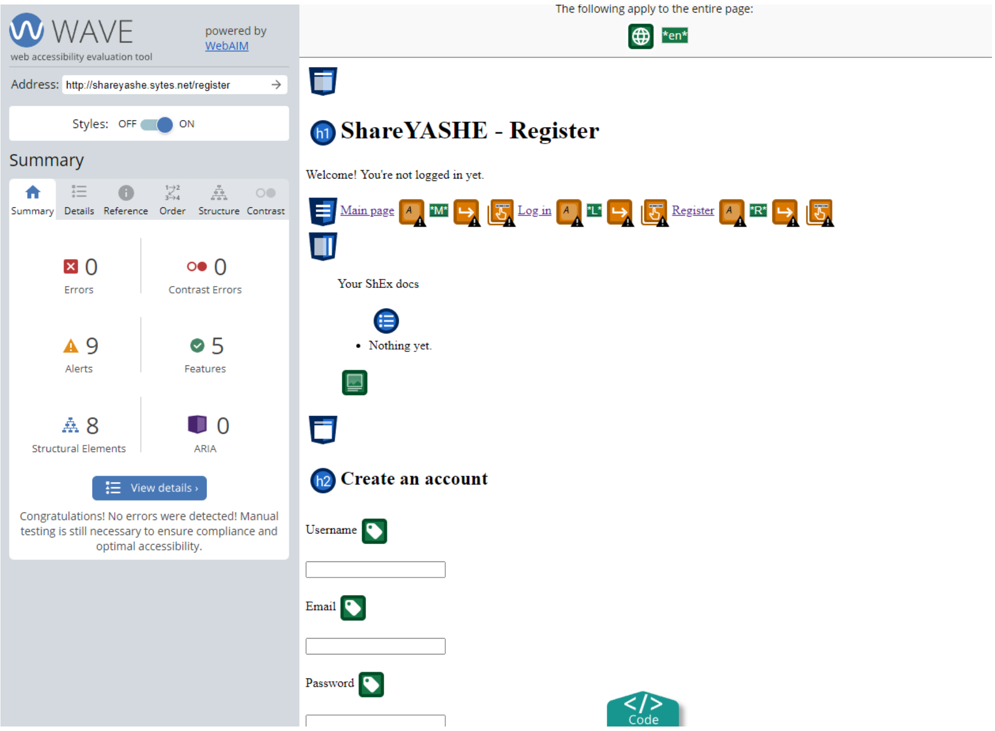
Log into your account

Username

Password

Figure 91. Results of WAVE test for log-in view.

6.3.2.4.4 Register view


WAVE
web accessibility evaluation tool
powered by WebAIM
Styles: OFF ON
Summary Details Reference Order Structure Contrast
View details ›
en
M
L
R
Code

Address: http://shareyashe.sytes.net/register

Summary

0 Errors

0 Contrast Errors

9 Alerts

5 Features

8 Structural Elements

0 ARIA

Congratulations! No errors were detected! Manual testing is still necessary to ensure compliance and optimal accessibility.

The following apply to the entire page:

ShareYASHE - Register

Welcome! You're not logged in yet.

Main page Log in Register

Your ShEx docs

- Nothing yet.

Create an account

Username

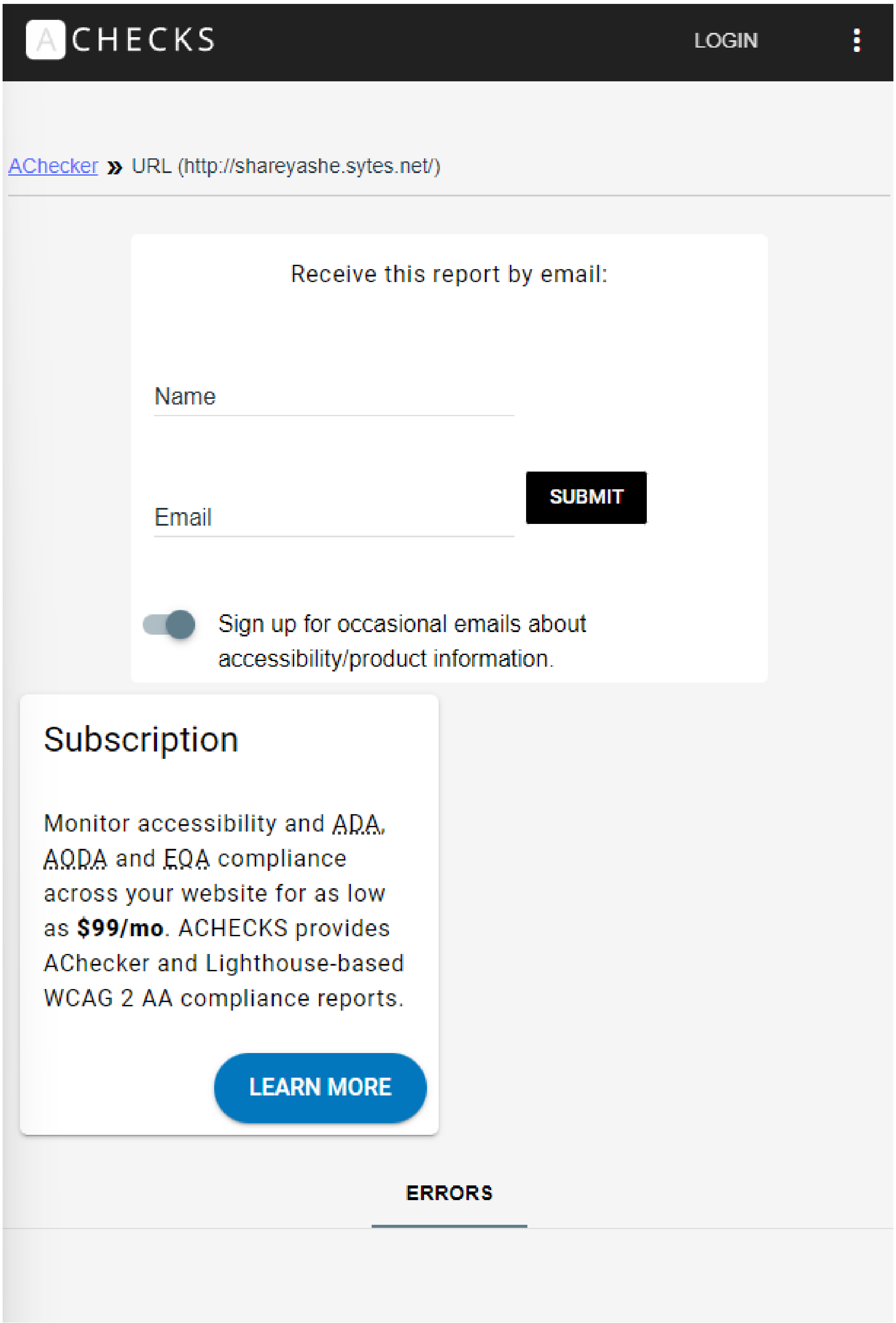
Email

Password

Figure 92. Results of WAVE test for register view.

6.3.2.5 aChecker test

6.3.2.5.1 Main view



AChecker » URL (http://shareyashe.sytes.net/)

Receive this report by email:

Name

Email

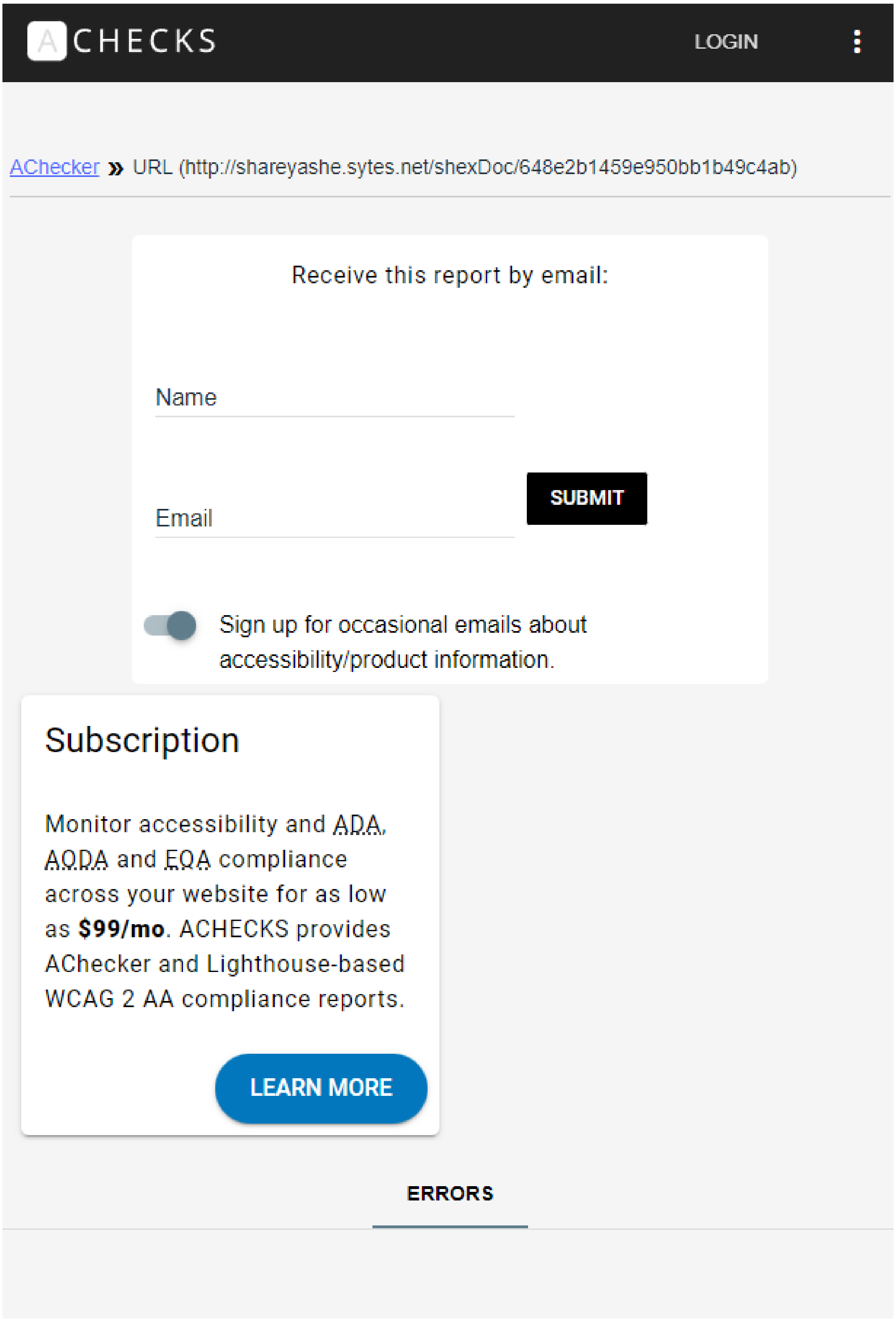
Sign up for occasional emails about accessibility/product information.

Subscription

Monitor accessibility and ADA, AODA and EOA compliance across your website for as low as **$99/mo**. ACHECKS provides AChecker and Lighthouse-based WCAG 2 AA compliance reports.

Figure 93. Results of aChecker test for main view.

6.3.2.5.2 ShEx document editing view



AChecker » URL (http://shareyashe.sytes.net/shexDoc/648e2b1459e950bb1b49c4ab)

Receive this report by email:

Name

Email

Sign up for occasional emails about accessibility/product information.

Subscription

Monitor accessibility and ADA, AODA and EOA compliance across your website for as low as **$99/mo**. ACHECKS provides AChecker and Lighthouse-based WCAG 2 AA compliance reports.

Figure 94. Results of aChecker test for ShEx document editing view.

6.3.2.5.3 Log-in view

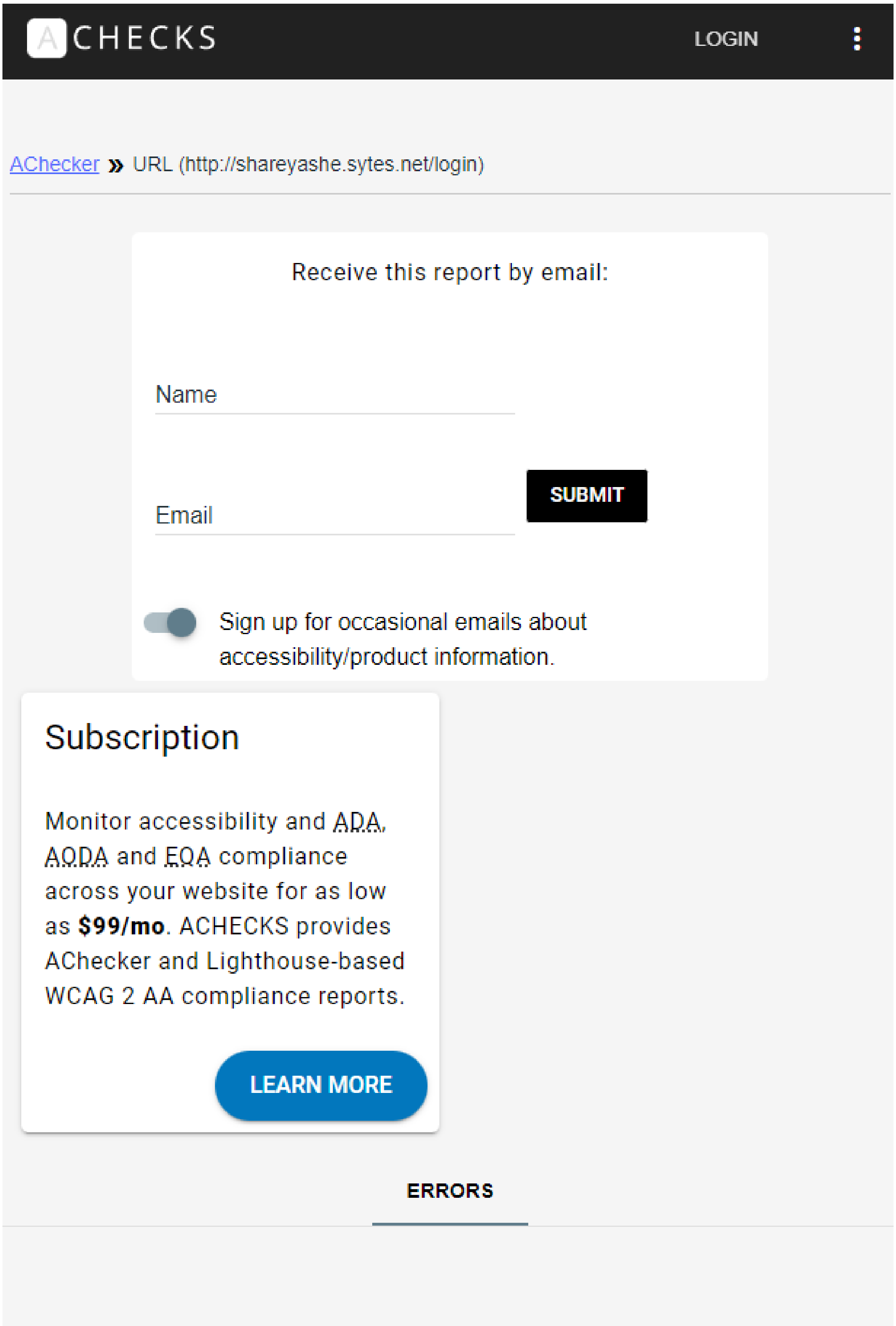
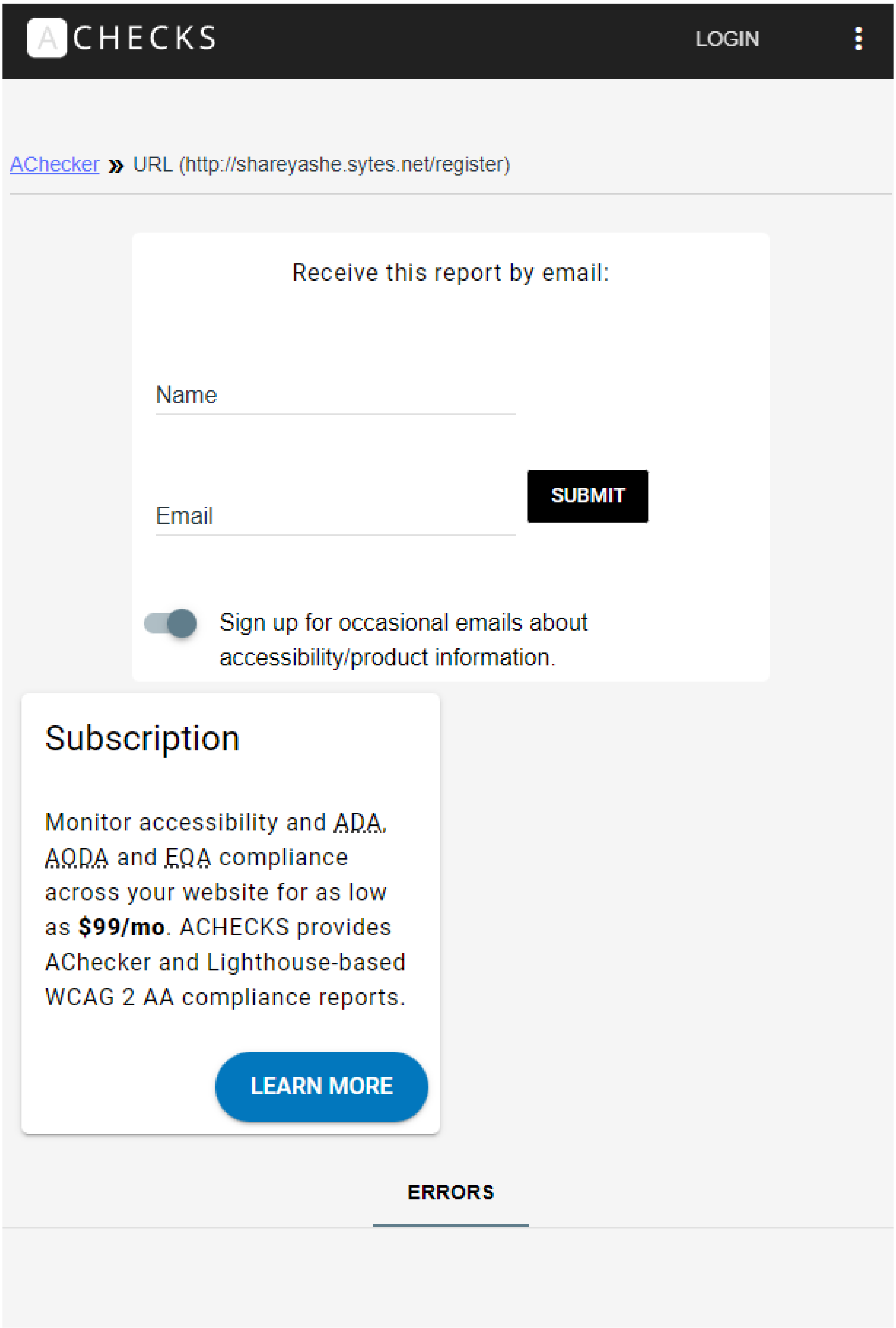

ACHECKS
LOGIN
AChecker » URL (http://shareyashe.sytes.net/login)
Receive this report by email:
Name
Email
SUBMIT
Sign up for occasional emails about accessibility/product information.
Subscription
Monitor accessibility and ADA, AODA and EOA compliance across your website for as low as $99/mo. ACHECKS provides AChecker and Lighthouse-based WCAG 2 AA compliance reports.
LEARN MORE
ERRORS

Figure 95. Results of aChecker test for log-in view.

6.3.2.5.4 Register view



AChecker » URL (http://shareyashe.sytes.net/register)

Receive this report by email:

Name

Email

Sign up for occasional emails about accessibility/product information.

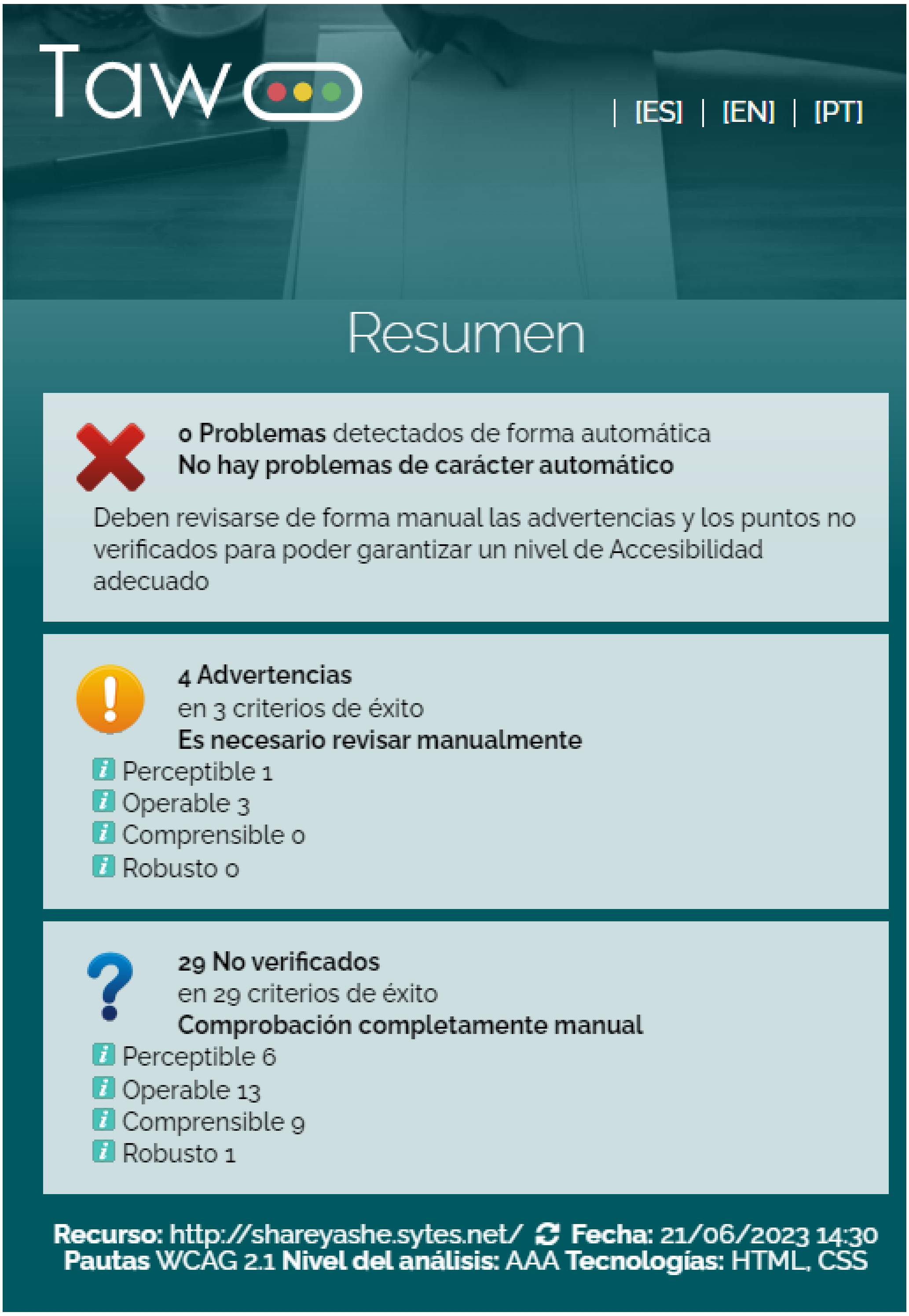
Subscription

Monitor accessibility and ADA, AODA and EOA compliance across your website for as low as **$99/mo**. ACHECKS provides AChecker and Lighthouse-based WCAG 2 AA compliance reports.

Figure 96. Results of aChecker test for register view.

6.3.2.6 TAW test

6.3.2.6.1 Main view



Resumen

0 **Problemas** detectados de forma automática
No hay problemas de carácter automático

Deben revisarse de forma manual las advertencias y los puntos no verificados para poder garantizar un nivel de Accesibilidad adecuado

4 **Advertencias**
en 3 criterios de éxito
Es necesario revisar manualmente

- Perceptible 1
- Operable 3
- Comprensible 0
- Robusto 0

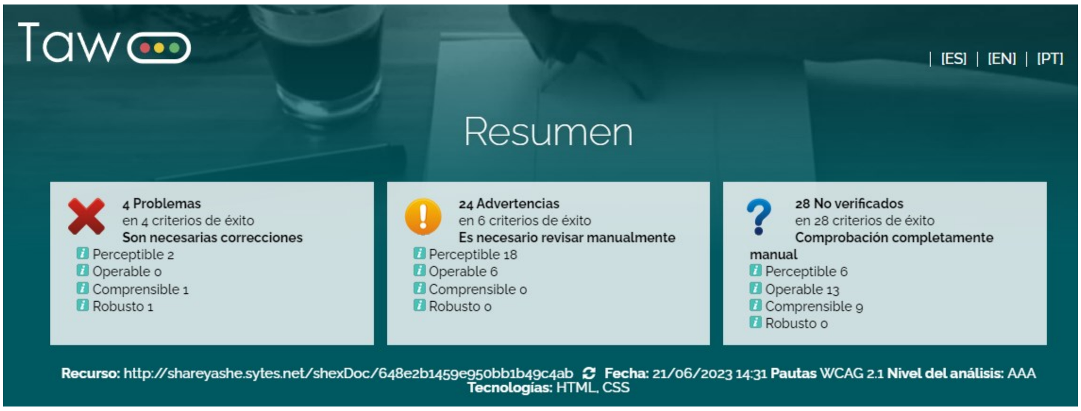
29 **No verificados**
en 29 criterios de éxito
Comprobación completamente manual

- Perceptible 6
- Operable 13
- Comprensible 9
- Robusto 1

Recurso: http://shareyashe.sytes.net/ **Fecha:** 21/06/2023 14:30
Pautas WCAG 2.1 **Nivel del análisis:** AAA **Tecnologías:** HTML, CSS

Figure 97. Results of TAW test for main view.

6.3.2.6.2 ShEx document editing view



Resumen

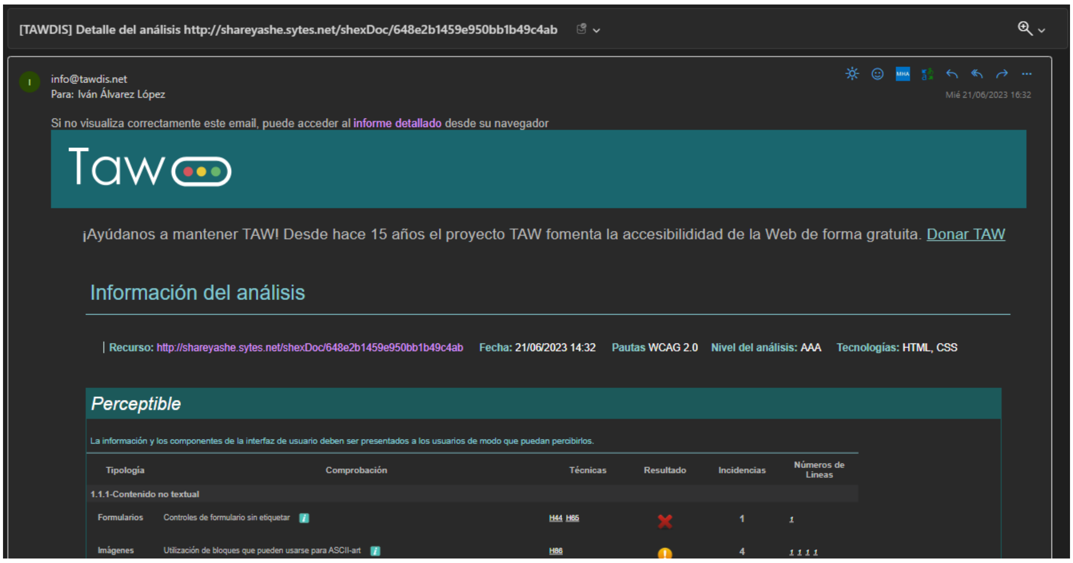
4 Problemas
en 4 criterios de éxito
Son necesarias correcciones
Perceptible 2
Operable 0
Comprensible 1
Robusto 1

24 Advertencias
en 6 criterios de éxito
Es necesario revisar manualmente
Perceptible 18
Operable 6
Comprensible 0
Robusto 0

28 No verificados
en 28 criterios de éxito
Comprobación completamente manual
Perceptible 6
Operable 13
Comprensible 9
Robusto 0

Recurso: http://shareyashe.sytes.net/shexDoc/648e2b1459e950bb1b49c4ab **Fecha:** 21/06/2023 14:31 **Pautas** WCAG 2.1 **Nivel del análisis:** AAA
Tecnologías: HTML, CSS

Figure 98. Results of TAW test for ShEx document editing view (1 of 13).



[TAWDIS] Detalle del análisis http://shareyashe.sytes.net/shexDoc/648e2b1459e950bb1b49c4ab

info@tawdis.net
Para: Iván Álvarez López
Mié 21/06/2023 16:32

Si no visualiza correctamente este email, puede acceder al informe detallado desde su navegador

¡Ayúdanos a mantener TAW! Desde hace 15 años el proyecto TAW fomenta la accesibilidad de la Web de forma gratuita. Donar TAW

Información del análisis

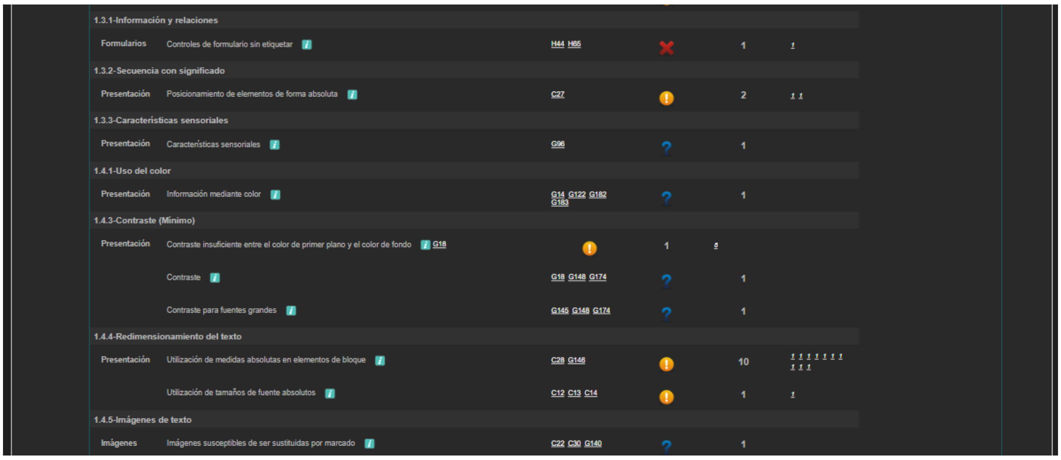
| Recurso: http://shareyashe.sytes.net/shexDoc/648e2b1459e950bb1b49c4ab Fecha: 21/06/2023 14:32 Pautas WCAG 2.0 Nivel del análisis: AAA Tecnologías: HTML, CSS

Perceptible

La información y los componentes de la interfaz de usuario deben ser presentados a los usuarios de modo que puedan percibirlos.

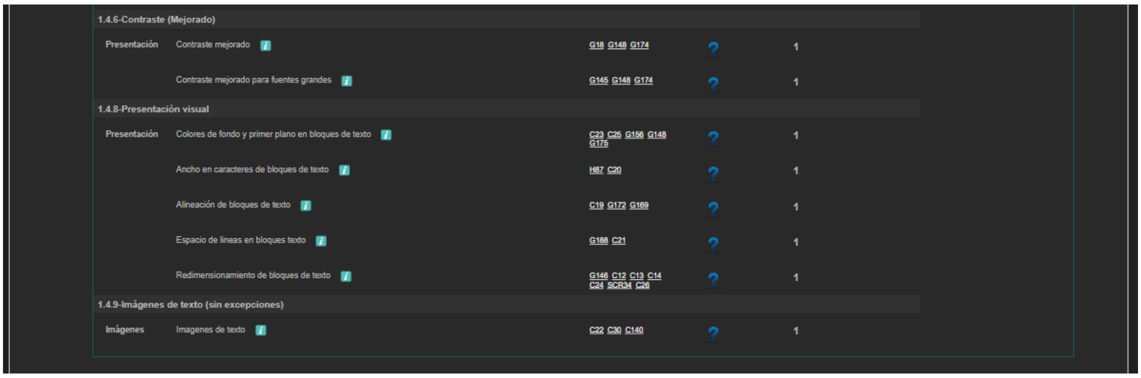
Tipología	Comprobación	Técnicas	Resultado	Incidencias	Números de Líneas
1.1.1-Contenido no textual					
Formularios	Controles de formulario sin etiquetar	H44 H65	✗	1	1
Imágenes	Utilización de bloques que pueden usarse para ASCII-art	H86	!	4	1 1 1 1

Figure 99. Results of TAW test for ShEx document editing view (2 of 13).



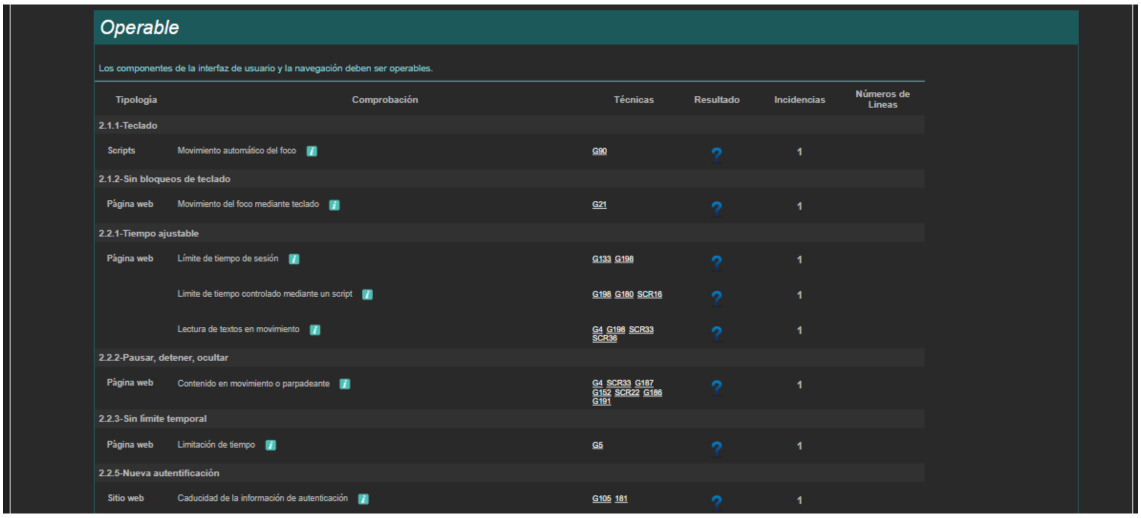
Tipología	Comprobación	Técnicas	Resultado	Incidencias	Números de Líneas
1.3.1-Información y relaciones					
Formularios	Controles de formulario sin etiquetar	H44 H65	✗	1	1
1.3.2-Secuencia con significado					
Presentación	Posicionamiento de elementos de forma absoluta	C27	!	2	1 1
1.3.3-Características sensoriales					
Presentación	Características sensoriales	G96	?	1	
1.4.1-Uso del color					
Presentación	Información mediante color	G14 G122 G182 G183	?	1	
1.4.3-Contraste (Mínimo)					
Presentación	Contraste insuficiente entre el color de primer plano y el color de fondo G18	!	1	8	
	Contraste	G18 G148 G174	?	1	
	Contraste para fuentes grandes	G145 G148 G174	?	1	
1.4.4-Redimensionamiento del texto					
Presentación	Utilización de medidas absolutas en elementos de bloque	C28 G146	!	10	1 1 1 1 1 1 1 1 1 1
	Utilización de tamaños de fuente absolutos	C12 C13 C14	!	1	1
1.4.5-Imágenes de texto					
Imágenes	Imágenes susceptibles de ser sustituidas por marcado	C22 C30 G140	?	1	

Figure 100. Results of TAW test for ShEx document editing view (3 of 13).



1.4.6-Contraste (Mejorado)					
Presentación	Contraste mejorado	G18 G148 G174	?	1	
	Contraste mejorado para fuentes grandes	G145 G148 G174	?	1	
1.4.8-Presentación visual					
Presentación	Colores de fondo y primer plano en bloques de texto	C23 C25 G156 G148 G175	?	1	
	Ancho en caracteres de bloques de texto	H87 C20	?	1	
	Alineación de bloques de texto	C19 G172 G169	?	1	
	Espacio de lineas en bloques texto	G188 C21	?	1	
	Redimensionamiento de bloques de texto	G146 C12 C13 C14 C24 SCR34 C26	?	1	
1.4.9-Imágenes de texto (sin excepciones)					
Imágenes	Imagenes de texto	C22 C30 C140	?	1	

Figure 101. Results of TAW test for ShEx document editing view (4 of 13).

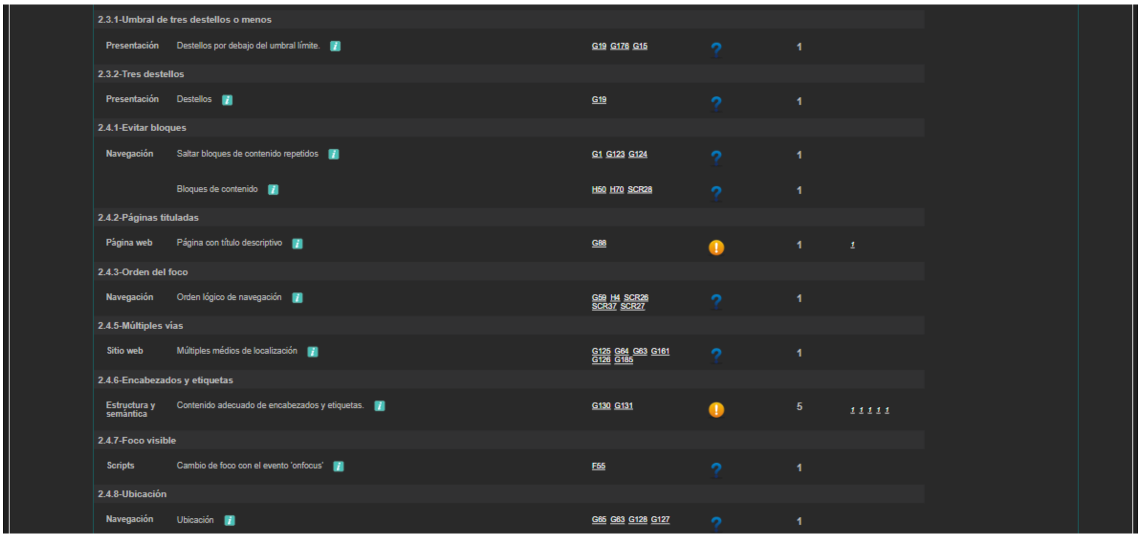


Operable

Los componentes de la interfaz de usuario y la navegación deben ser operables.

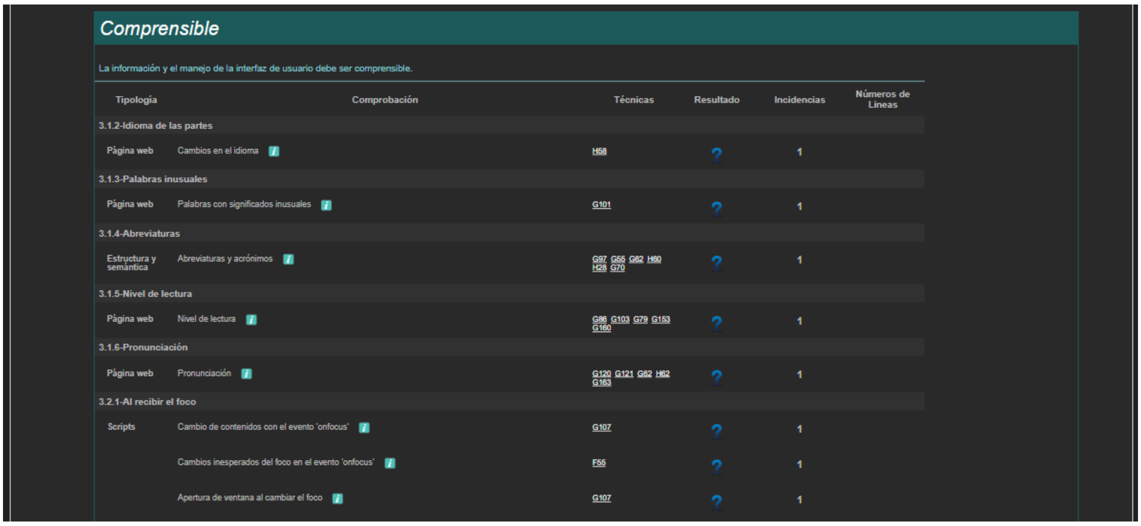
Tipología	Comprobación	Técnicas	Resultado	Incidencias	Números de Lineas
2.1.1-Teclado					
Scripts	Movimiento automático del foco	G90	?	1	
2.1.2-Sin bloqueos de teclado					
Página web	Movimiento del foco mediante teclado	G21	?	1	
2.2.1-Tiempo ajustable					
Página web	Límite de tiempo de sesión	G133 G198	?	1	
	Limite de tiempo controlado mediante un script	G198 G180 SCR16	?	1	
	Lectura de textos en movimiento	G4 G198 SCR33 SCR36	?	1	
2.2.2-Pausar, detener, ocultar					
Página web	Contenido en movimiento o parpadeante	G4 SCR33 G187 G152 SCR22 G186 G191	?	1	
2.2.3-Sin límite temporal					
Página web	Limitación de tiempo	G5	?	1	
2.2.5-Nueva autentificación					
Sitio web	Caducidad de la información de autenticación	G105 181	?	1	

Figure 102. Results of TAW test for ShEx document editing view (5 of 13).



2.3.1-Umbral de tres destellos o menos					
Presentación	Destellos por debajo del umbral límite.	G19 G176 G15	?	1	
2.3.2-Tres destellos					
Presentación	Destellos	G19	?	1	
2.4.1-Evitar bloques					
Navegación	Saltar bloques de contenido repetidos	G1 G123 G124	?	1	
	Bloques de contenido	H50 H70 SCR28	?	1	
2.4.2-Páginas tituladas					
Página web	Página con título descriptivo	G88	!	1	1
2.4.3-Orden del foco					
Navegación	Orden lógico de navegación	G59 H4 SCR26 SCR37 SCR27	?	1	
2.4.5-Múltiples vías					
Sitio web	Múltiples médios de localización	G125 G64 G63 G161 G126 G185	?	1	
2.4.6-Encabezados y etiquetas					
Estructura y semántica	Contenido adecuado de encabezados y etiquetas.	G130 G131	!	5	1 1 1 1 1
2.4.7-Foco visible					
Scripts	Cambio de foco con el evento 'onfocus'	F55	?	1	
2.4.8-Ubicación					
Navegación	Ubicación	G65 G63 G128 G127	?	1	

Figure 103. Results of TAW test for ShEx document editing view (6 of 13).

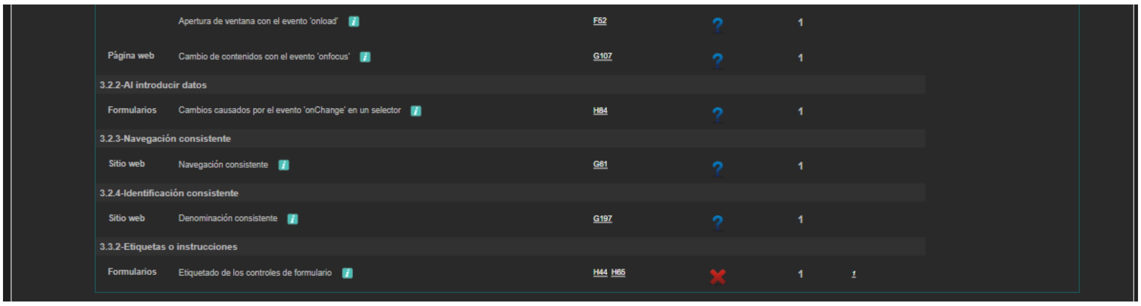


Comprensible

La información y el manejo de la interfaz de usuario debe ser comprensible.

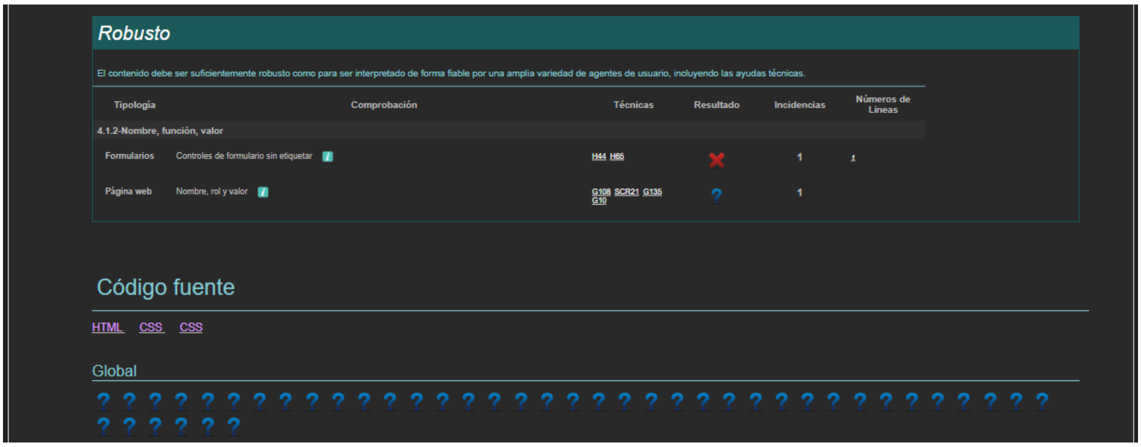
Tipología	Comprobación	Técnicas	Resultado	Incidencias	Números de Lineas
3.1.2-Idioma de las partes					
Página web	Cambios en el idioma ℹ	H58	?	1	
3.1.3-Palabras inusuales					
Página web	Palabras con significados inusuales ℹ	G101	?	1	
3.1.4-Abreviaturas					
Estructura y semántica	Abreviaturas y acrónimos ℹ	G97 G55 G62 H60 H28 G70	?	1	
3.1.5-Nivel de lectura					
Página web	Nivel de lectura ℹ	G86 G103 G79 G153 G160	?	1	
3.1.6-Pronunciación					
Página web	Pronunciación ℹ	G120 G121 G62 H62 G163	?	1	
3.2.1-Al recibir el foco					
Scripts	Cambio de contenidos con el evento 'onfocus' ℹ	G107	?	1	
	Cambios inesperados del foco en el evento 'onfocus' ℹ	F55	?	1	
	Apertura de ventana al cambiar el foco ℹ	G107	?	1	

Figure 104. Results of TAW test for ShEx document editing view (7 of 13).



Tipología	Comprobación	Técnicas	Resultado	Incidencias	Números de Lineas
	Apertura de ventana con el evento 'onload' ℹ	F52	?	1	
Página web	Cambio de contenidos con el evento 'onfocus' ℹ	G107	?	1	
3.2.2-Al introducir datos					
Formularios	Cambios causados por el evento 'onChange' en un selector ℹ	H84	?	1	
3.2.3-Navegación consistente					
Sitio web	Navegación consistente ℹ	G61	?	1	
3.2.4-Identificación consistente					
Sitio web	Denominación consistente ℹ	G197	?	1	
3.3.2-Etiquetas o instrucciones					
Formularios	Etiquetado de los controles de formulario ℹ	H44 H65	✗	1	1

Figure 105. Results of TAW test for ShEx document editing view (8 of 13).



Robusto

El contenido debe ser suficientemente robusto como para ser interpretado de forma fiable por una amplia variedad de agentes de usuario, incluyendo las ayudas técnicas.

Tipología	Comprobación	Técnicas	Resultado	Incidencias	Números de Lineas
4.1.2-Nombre, función, valor					
Formularios	Controles de formulario sin etiquetar ℹ	H44 H65	✗	1	1
Página web	Nombre, rol y valor ℹ	G108 SCR21 G135 G10	?	1	

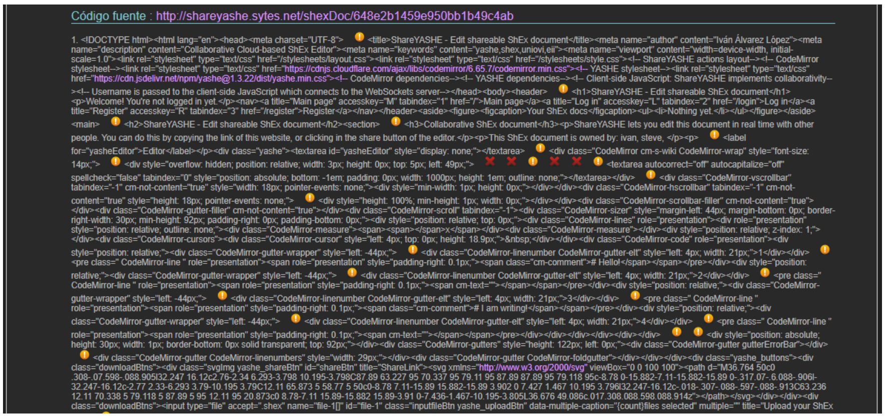
Código fuente

HTML CSS CSS

Global

? ?
? ? ? ? ? ?

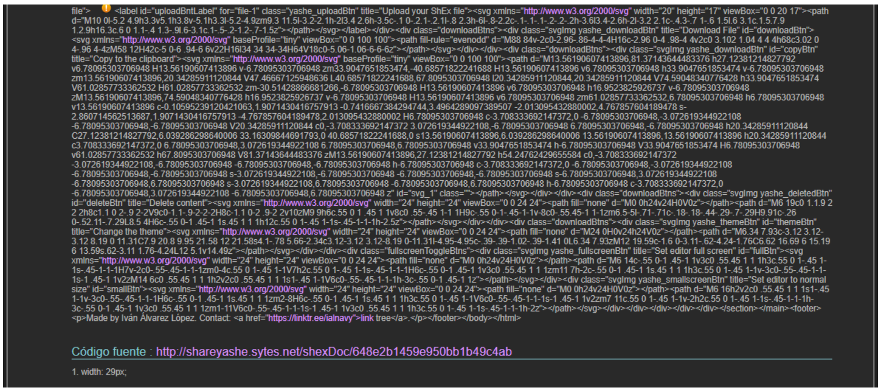
Figure 106. Results of TAW test for ShEx document editing view (9 of 13).



Código fuente : http://shareyashe.sytes.net/shexDoc/648e2b1459e950bb1b49c4ab

```
1. <!DOCTYPE html><html lang="en"><head><meta charset="UTF-8">   <title>ShareYASHE - Edit shareable ShEx document</title><meta name="author" content="Iván Álvarez López"><meta
name="description" content="Collaborative Cloud-based ShEx Editor"><meta name="keywords" content="yashe,shex,uniovi,eii"><meta name="viewport" content="width=device-width, initial-
scale=1.0"><link rel="stylesheet" type="text/css" href="/stylesheets/layout.css"><link rel="stylesheet" type="text/css" href="/stylesheets/style.css"><!-- ShareYASHE actions layout--><!-- CodeMirror
stylesheet--><link rel="stylesheet" type="text/css" href="https://cdnjs.cloudflare.com/ajax/libs/codemirror/6.65.7/codemirror.min.css"><!-- YASHE stylesheet--><link rel="stylesheet" type="text/css"
href="https://cdn.jsdelivr.net/npm/yashe@1.3.22/dist/yashe.min.css"><!-- CodeMirror dependencies--><!-- YASHE dependencies--><!-- Client-side JavaScript: ShareYASHE implements collaborativity--
><!-- Username is passed to the client-side JavaScript which connects to the WebSockets server--></head><body><header>   <h1>ShareYASHE - Edit shareable ShEx document</h1>
<p>Welcome! You're not logged in yet.</p><nav><a title="Main page" accesskey="M" tabindex="1" href="/">Main page</a><a title="Log in" accesskey="L" tabindex="2" href="/login">Log in</a><a
title="Register" accesskey="R" tabindex="3" href="/register">Register</a></nav></header><aside><figure><figcaption>Your ShEx docs</figcaption><ul><li>Nothing yet.</li></ul></figure></aside>
<main>   <h2>ShareYASHE - Edit shareable ShEx document</h2><section>   <h3>Collaborative ShEx document</h3><p>ShareYASHE lets you edit this document in real time with other
people. You can do this by copying the link of this website, or clicking in the share button of the editor.</p><p>This ShEx document is owned by: ivan, steve, </p><p>   <label
for="yasheEditor">Editor</label></p><div class="yashe"><textarea id="yasheEditor" style="display: none;"></textarea>   <div class="CodeMirror cm-s-wiki CodeMirror-wrap" style="font-size:
14px;">   <div style="overflow: hidden; position: relative; width: 3px; height: 0px; top: 5px; left: 49px;">   ✗   ✗      ✗   ✗      <textarea autocorrect="off" autocapitalize="off"
spellcheck="false" tabindex="0" style="position: absolute; bottom: -1em; padding: 0px; width: 1000px; height: 1em; outline: none;"></textarea></div>   <div class="CodeMirror-vscrollbar"
tabindex="-1" cm-not-content="true" style="width: 18px; pointer-events: none;"><div style="min-width: 1px; height: 0px;"></div></div><div class="CodeMirror-hscrollbar" tabindex="-1" cm-not-
content="true" style="height: 18px; pointer-events: none;">   <div style="height: 100%; min-height: 1px; width: 0px;"></div></div><div class="CodeMirror-scrollbar-filler" cm-not-content="true">
</div><div class="CodeMirror-gutter-filler" cm-not-content="true"></div><div class="CodeMirror-scroll" tabindex="-1"><div class="CodeMirror-sizer" style="margin-left: 44px; margin-bottom: 0px; border-
right-width: 30px; min-height: 92px; padding-right: 0px; padding-bottom: 0px;"><div style="position: relative; top: 0px;"><div class="CodeMirror-lines" role="presentation"><div role="presentation"
style="position: relative; outline: none;"><div class="CodeMirror-measure"><span><span></span>x</span></div><div class="CodeMirror-measure"></div><div style="position: relative; z-index: 1;">
</div><div class="CodeMirror-cursors"><div class="CodeMirror-cursor" style="left: 4px; top: 0px; height: 18.9px;">&nbsp;</div></div><div class="CodeMirror-code" role="presentation"><div
style="position: relative;"><div class="CodeMirror-gutter-wrapper" style="left: -44px;">   <div class="CodeMirror-linenumber CodeMirror-gutter-elt" style="left: 4px; width: 21px;">1</div></div>
<pre class=" CodeMirror-line " role="presentation"><span role="presentation" style="padding-right: 0.1px;"><span class="cm-comment"># Hello!</span></span></pre></div><div style="position:
relative;"><div class="CodeMirror-gutter-wrapper" style="left: -44px;">   <div class="CodeMirror-linenumber CodeMirror-gutter-elt" style="left: 4px; width: 21px;">2</div></div>   <pre class="
CodeMirror-line " role="presentation"><span role="presentation" style="padding-right: 0.1px;"><span cm-text=""></span></span></pre></div><div style="position: relative;"><div class="CodeMirror-
gutter-wrapper" style="left: -44px;">   <div class="CodeMirror-linenumber CodeMirror-gutter-elt" style="left: 4px; width: 21px;">3</div></div>   <pre class=" CodeMirror-line "
role="presentation"><span role="presentation" style="padding-right: 0.1px;"><span class="cm-comment"># I am writing!</span></span></pre></div><div style="position: relative;"><div
class="CodeMirror-gutter-wrapper" style="left: -44px;">   <div class="CodeMirror-linenumber CodeMirror-gutter-elt" style="left: 4px; width: 21px;">4</div></div>   <pre class=" CodeMirror-line "
role="presentation"><span role="presentation" style="padding-right: 0.1px;"><span cm-text=""></span></span></pre></div></div></div></div></div>      <div style="position: absolute;
height: 30px; width: 1px; border-bottom: 0px solid transparent; top: 92px;"></div><div class="CodeMirror-gutters" style="height: 122px; left: 0px;"><div class="CodeMirror-gutter gutterErrorBar"></div>
   <div class="CodeMirror-gutter CodeMirror-linenumbers" style="width: 29px;"></div><div class="CodeMirror-gutter CodeMirror-foldgutter"></div></div></div><div class="yashe_buttons"><div
class="downloadBtns"><div class="svgImg yashe_shareBtn" id="shareBtn" title="ShareLink"><svg xmlns="http://www.w3.org/2000/svg" viewBox="0 0 100 100"><path d="M36.764 50c0
.308-.07.598-.088.905l32.247 16.12c2.76-2.34 6.293-3.798 10.195-3.798C87.89 63.227 95 70.337 95 79.11 95 87.89 87.89 95 79.118 95c-8.78 0-15.882-7.11-15.882-15.89 0-.317.07-.6.088-.906l-
32.247-16.12c-2.77 2.33-6.293 3.79-10.195 3.79C12.11 65.873 5 58.77 5 50c0-8.78 7.11-15.89 15.882-15.89 3.902 0 7.427 1.467 10.195 3.796l32.247-16.12c-.018-.307-.088-.597-.088-.913C63.236
12.11 70.338 5 79.118 5 87.89 5 95 12.11 95 20.873c0 8.78-7.11 15.89-15.882 15.89-3.91 0-7.436-1.467-10.195-3.805L36.676 49.086c.017.308.088.598.088.914z"></path></svg></div></div><div
class="downloadBtns"><input type="file" accept=".shex" name="file-1[]" id="file-1" class="inputfileBtn yashe_uploadBtn" data-multiple-caption="{count}files selected" multiple="" title="Upload your ShEx
```

Figure 107. Results of TAW test for ShEx document editing view (10 of 13).

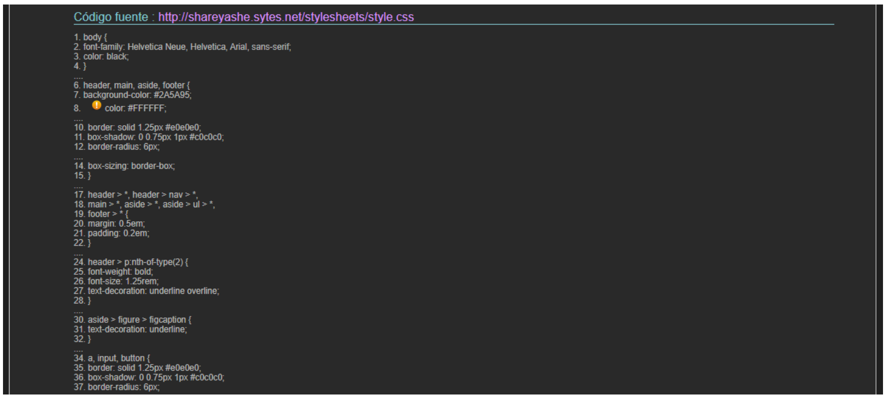


```
file">   <label id="uploadBntLabel" for="file-1" class="yashe_uploadBtn" title="Upload your ShEx file"><svg xmlns="http://www.w3.org/2000/svg" width="20" height="17" viewBox="0 0 20 17"><path
d="M10 0l-5.2 4.9h3.3v5.1h3.8v-5.1h3.3l-5.2-4.9zm9.3 11.5l-3.2-2.1h-2l3.4 2.6h-3.5c-.1 0-.2.1-.2.1l-.8 2.3h-6l-.8-2.2c-.1-.1-.1-.2-.2-.2h-3.6l3.4-2.6h-2l-3.2 2.1c-.4.3-.7 1-.6 1.5l.6 3.1c.1.5.7.9
1.2.9h16.3c.6 0 1.1-.4 1.3-.9l.6-3.1c.1-.5-.2-1.2-.7-1.5z"></path></svg></label></div><div class="downloadBtns"><div class="svgImg yashe_downloadBtn" title="Download File" id="downloadBtn">
<svg xmlns="http://www.w3.org/2000/svg" baseProfile="tiny" viewBox="0 0 100 100"><path fill-rule="evenodd" d="M88 84v-2c0-2.96-.86-4-4-4H16c-2.96 0-4 .98-4 4v2c0 3.102 1.04 4 4 4h68c3.02 0
4-.96 4-4zM58 12H42c-5 0-6 .94-6 6v22H16l34 34 34-34H64V18c0-5.06-1.06-6-6-6z"></path></svg></div></div><div class="downloadBtns"><div class="svgImg yashe_downloadBtn" id="copyBtn"
title="Copy to the clipboard"><svg xmlns="http://www.w3.org/2000/svg" baseProfile="tiny" viewBox="0 0 100 100"><path d="M13.56190607413896,81.37143644483376 h27.12381214827792
v6.78095303706948 H13.56190607413896 v-6.78095303706948 zm33.9047651853474,-40.68571822241688 H13.56190607413896 v6.78095303706948 h33.9047651853474 v-6.78095303706948
zm13.56190607413896,20.34285911120844 V47.46667125948636 L40.68571822241688,67.8095303706948 l20.34285911120844,20.34285911120844 V74.59048340776428 h33.9047651853474
V61.02857733362532 H61.02857733362532 zm-30.51428866681266,-6.78095303706948 H13.56190607413896 v6.78095303706948 h16.9523825926737 v-6.78095303706948
zM13.56190607413896,74.59048340776428 h16.9523825926737 v-6.78095303706948 H13.56190607413896 v6.78095303706948 zm61.02857733362532,6.78095303706948 h6.78095303706948
v13.56190607413896 c-0.10595239120421063,1.9071430416757913 -0.7416667384294744,3.4964289097389507 -2.013095432880002,4.767857604189478 s-
2.860714562513687,1.9071430416757913 -4.767857604189478,2.013095432880002 H6.78095303706948 c-3.708333692147372,0 -6.78095303706948,-3.072619344922108
-6.78095303706948,-6.78095303706948 V20.34285911120844 c0,-3.708333692147372 3.072619344922108,-6.78095303706948 6.78095303706948,-6.78095303706948 h20.34285911120844
C27.12381214827792,6.039286298640006 33.16309844691793,0 40.68571822241688,0 s13.56190607413896,6.039286298640006 13.56190607413896,13.56190607413896 h20.34285911120844
c3.708333692147372,0 6.78095303706948,3.072619344922108 6.78095303706948,6.78095303706948 v33.9047651853474 h-6.78095303706948 V33.9047651853474 H6.78095303706948
v61.02857733362532 h67.8095303706948 V81.37143644483376 zM13.56190607413896,27.12381214827792 h54.24762429655584 c0,-3.708333692147372
-3.072619344922108,-6.78095303706948 -6.78095303706948,-6.78095303706948 h-6.78095303706948 c-3.708333692147372,0 -6.78095303706948,-3.072619344922108
-6.78095303706948,-6.78095303706948 s-3.072619344922108,-6.78095303706948 -6.78095303706948,-6.78095303706948 s-6.78095303706948,3.072619344922108
-6.78095303706948,6.78095303706948 s-3.072619344922108,6.78095303706948 -6.78095303706948,6.78095303706948 h-6.78095303706948 c-3.708333692147372,0
-6.78095303706948,3.072619344922108 -6.78095303706948,6.78095303706948 z" id="svg_1" class=""></path></svg></div></div><div class="downloadBtns"><div class="svgImg yashe_deletedBtn"
id="deleteBtn" title="Delete content"><svg xmlns="http://www.w3.org/2000/svg" width="24" height="24" viewBox="0 0 24 24"><path fill="none" d="M0 0h24v24H0V0z"></path><path d="M6 19c0 1.1.9 2
2 2h8c1.1 0 2-.9 2-2V9c0-1.1-.9-2-2-2H8c-1.1 0-2 .9-2 2v10zM9 9h6c.55 0 1 .45 1 1v8c0 .55-.45 1-1 1H9c-.55 0-1-.45-1-1v-8c0-.55.45-1 1-1zm6.5-5l-.71-.71c-.18-.18-.44-.29-.7-.29H9.91c-.26
0-.52.11-.7.29L8.5 4H6c-.55 0-1 .45-1 1s.45 1 1 1h12c.55 0 1-.45 1-1s-.45-1-1-1h-2.5z"></path></svg></div></div><div class="downloadBtns"><div class="svgImg yashe_themeBtn" id="themeBtn"
title="Change the theme"><svg xmlns="http://www.w3.org/2000/svg" width="24" height="24" viewBox="0 0 24 24"><path fill="none" d="M24 0H0v24h24V0z"></path><path d="M6.34 7.93c-3.12 3.12-
3.12 8.19 0 11.31C7.9 20.8 9.95 21.58 12 21.58s4.1-.78 5.66-2.34c3.12-3.12 3.12-8.19 0-11.31l-4.95-4.95c-.39-.39-1.02-.39-1.41 0L6.34 7.93zM12 19.59c-1.6 0-3.11-.62-4.24-1.76C6.62 16.69 6 15.19
6 13.59s.62-3.11 1.76-4.24L12 5.1v14.49z"></path></svg></div></div><div class="fullscreenToggleBtns"><div class="svgImg yashe_fullscreenBtn" title="Set editor full screen" id="fullBtn"><svg
xmlns="http://www.w3.org/2000/svg" width="24" height="24" viewBox="0 0 24 24"><path fill="none" d="M0 0h24v24H0V0z"></path><path d="M6 14c-.55 0-1 .45-1 1v3c0 .55.45 1 1 1h3c.55 0 1-.45 1-
1s-.45-1-1-1H7v-2c0-.55-.45-1-1-1zm0-4c.55 0 1-.45 1-1V7h2c.55 0 1-.45 1-1s-.45-1-1-1H6c-.55 0-1 .45-1 1v3c0 .55.45 1 1 1zm11 7h-2c-.55 0-1 .45-1 1s.45 1 1 1h3c.55 0 1-.45 1-1v-3c0-.55-.45-1-1-
1s-1 .45-1 1v2zM14 6c0 .55.45 1 1 1h2v2c0 .55.45 1 1 1s1-.45 1-1V6c0-.55-.45-1-1-1h-3c-.55 0-1 .45-1 1z"></path></svg></div><div class="svgImg yashe_smallscreenBtn" title="Set editor to normal
size" id="smallBtn"><svg xmlns="http://www.w3.org/2000/svg" width="24" height="24" viewBox="0 0 24 24"><path fill="none" d="M0 0h24v24H0V0z"></path><path d="M6 16h2v2c0 .55.45 1 1 1s1-.45
1-1v-3c0-.55-.45-1-1-1H6c-.55 0-1 .45-1 1s.45 1 1 1zm2-8H6c-.55 0-1 .45-1 1s.45 1 1 1h3c.55 0 1-.45 1-1V6c0-.55-.45-1-1-1s-1 .45-1 1v2zm7 11c.55 0 1-.45 1-1v-2h2c.55 0 1-.45 1-1s-.45-1-1-1h-
3c-.55 0-1 .45-1 1v3c0 .55.45 1 1 1zm1-11V6c0-.55-.45-1-1-1s-1 .45-1 1v3c0 .55.45 1 1 1h3c.55 0 1-.45 1-1s-.45-1-1-1h-2z"></path></svg></div></div></div></div></section></main><footer>
<p>Made by Iván Álvarez López. Contact: <a href="https://linktr.ee/ialnavy">link tree</a>.</p></footer></body></html>
```

Código fuente : http://shareyashe.sytes.net/shexDoc/648e2b1459e950bb1b49c4ab

```
1. width: 29px;
```

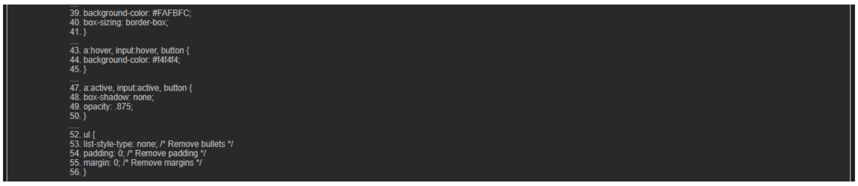
Figure 108. Results of TAW test for ShEx document editing view (11 of 13).



Código fuente : http://shareyashe.sytes.net/stylesheets/style.css

```
1. body {
2. font-family: Helvetica Neue, Helvetica, Arial, sans-serif;
3. color: black;
4. }
....
6. header, main, aside, footer {
7. background-color: #2A5A95;
8.    color: #FFFFFF;
....
10. border: solid 1.25px #e0e0e0;
11. box-shadow: 0 0.75px 1px #c0c0c0;
12. border-radius: 6px;
....
14. box-sizing: border-box;
15. }
....
17. header > *, header > nav > *,
18. main > *, aside > *, aside > ul > *,
19. footer > * {
20. margin: 0.5em;
21. padding: 0.2em;
22. }
....
24. header > p:nth-of-type(2) {
25. font-weight: bold;
26. font-size: 1.25rem;
27. text-decoration: underline overline;
28. }
....
30. aside > figure > figcaption {
31. text-decoration: underline;
32. }
....
34. a, input, button {
35. border: solid 1.25px #e0e0e0;
36. box-shadow: 0 0.75px 1px #c0c0c0;
37. border-radius: 6px;
```

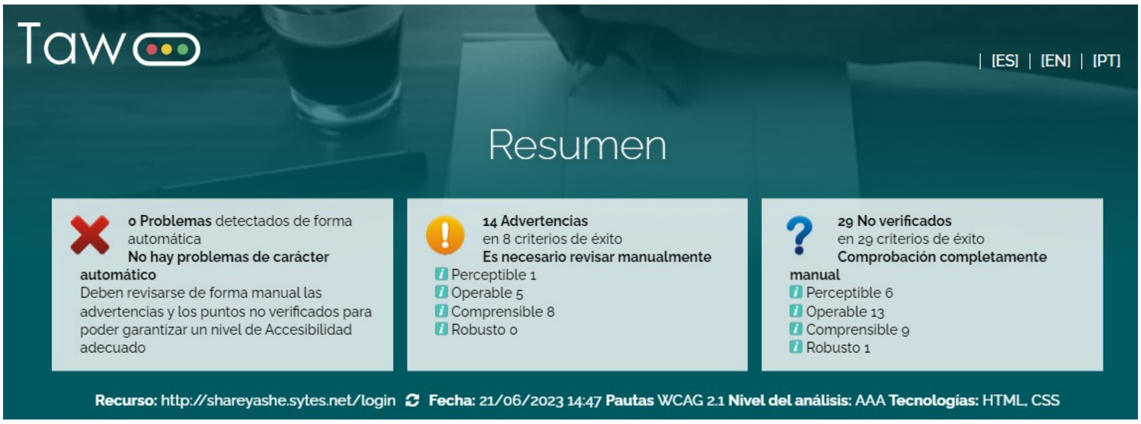
Figure 109. Results of TAW test for ShEx document editing view (12 of 13).



```
....
39. background-color: #FAFBFC;
40. box-sizing: border-box;
41. }
....
43. a:hover, input:hover, button {
44. background-color: #f4f4f4;
45. }
....
47. a:active, input:active, button {
48. box-shadow: none;
49. opacity: .875;
50. }
....
52. ul {
53. list-style-type: none; /* Remove bullets */
54. padding: 0; /* Remove padding */
55. margin: 0; /* Remove margins */
56. }
```

Figure 110. Results of TAW test for ShEx document editing view (13 of 13).

6.3.2.6.3 Log-in view



Resumen

0 **Problemas** detectados de forma automática
No hay problemas de carácter automático
Deben revisarse de forma manual las advertencias y los puntos no verificados para poder garantizar un nivel de Accesibilidad adecuado

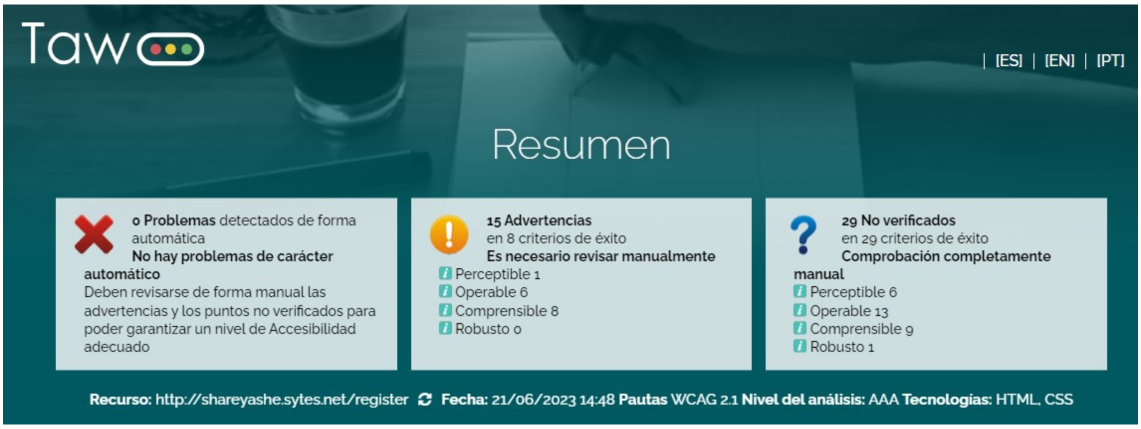
14 **Advertencias**
en 8 criterios de éxito
Es necesario revisar manualmente
Perceptible 1
Operable 5
Comprensible 8
Robusto 0

29 **No verificados**
en 29 criterios de éxito
Comprobación completamente manual
Perceptible 6
Operable 13
Comprensible 9
Robusto 1

Recurso: http://shareyashe.sytes.net/login **Fecha:** 21/06/2023 14:47 **Pautas** WCAG 2.1 **Nivel del análisis:** AAA **Tecnologías:** HTML, CSS

Figure 111. Results of TAW test for log-in view.

6.3.2.6.4 Register view



Resumen

0 **Problemas** detectados de forma automática
No hay problemas de carácter automático
Deben revisarse de forma manual las advertencias y los puntos no verificados para poder garantizar un nivel de Accesibilidad adecuado

15 **Advertencias**
en 8 criterios de éxito
Es necesario revisar manualmente
Perceptible 1
Operable 6
Comprensible 8
Robusto 0

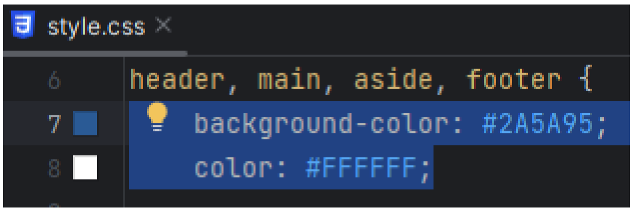
29 **No verificados**
en 29 criterios de éxito
Comprobación completamente manual
Perceptible 6
Operable 13
Comprensible 9
Robusto 1

Recurso: http://shareyashe.sytes.net/register **Fecha:** 21/06/2023 14:48 **Pautas** WCAG 2.1 **Nivel del análisis:** AAA **Tecnologías:** HTML, CSS

Figure 112. Results of TAW test for register view.

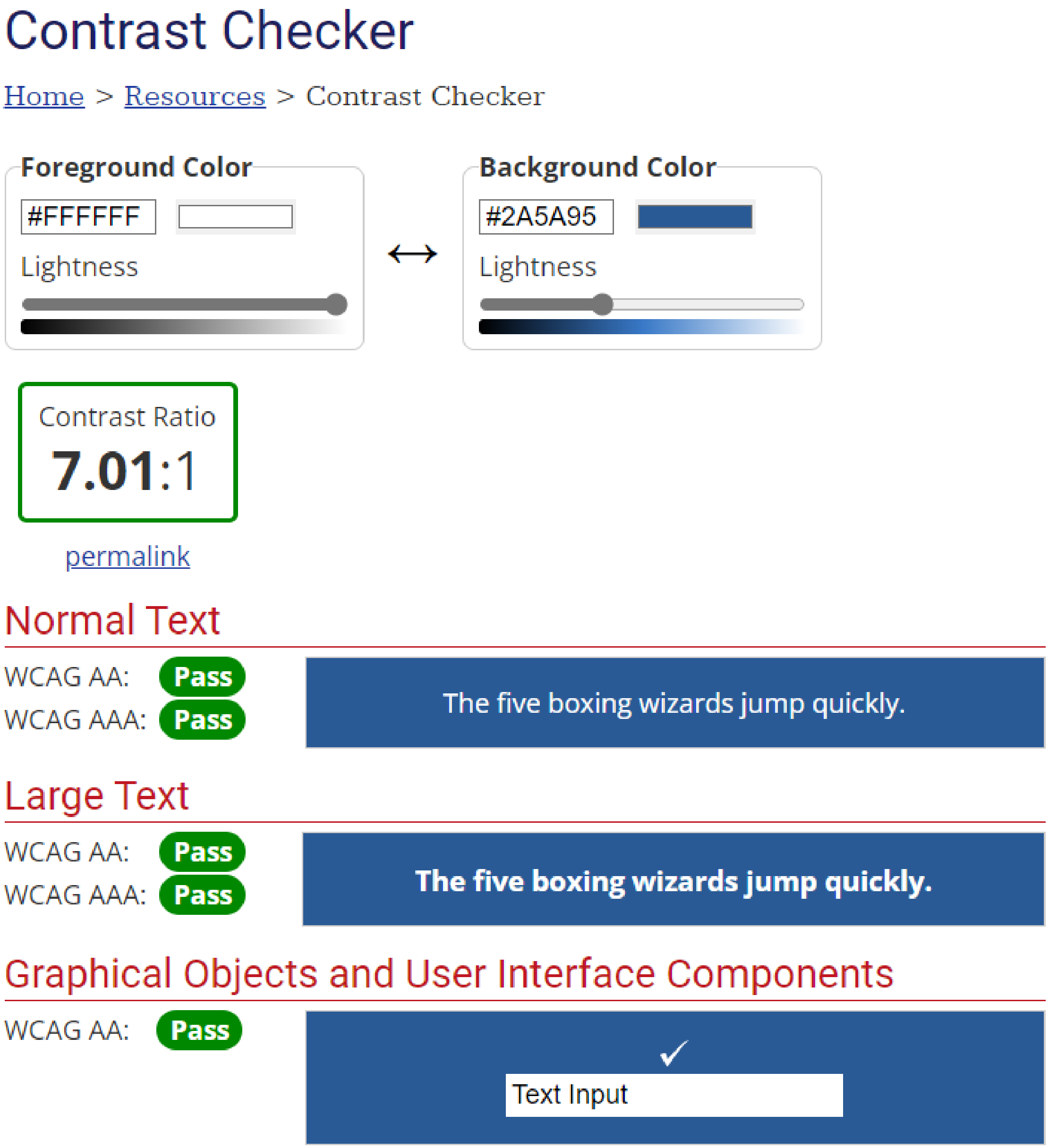
6.3.2.7 WebAIM Contrast Check

6.3.2.7.1 Main style



```css
header, main, aside, footer {
    background-color: #2A5A95;
    color: #FFFFFF;
```

Figure 113. Main style colours.



Contrast Checker

Home > Resources > Contrast Checker

Foreground Color: #FFFFFF
Lightness

↔

Background Color: #2A5A95
Lightness

Contrast Ratio
7.01:1

permalink

Normal Text

WCAG AA: Pass
WCAG AAA: Pass

The five boxing wizards jump quickly.

Large Text

WCAG AA: Pass
WCAG AAA: Pass

The five boxing wizards jump quickly.

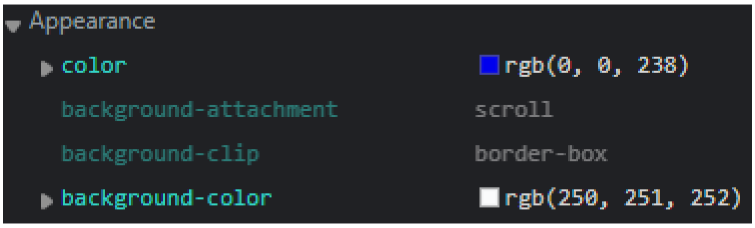
Graphical Objects and User Interface Components

WCAG AA: Pass

✓
Text Input

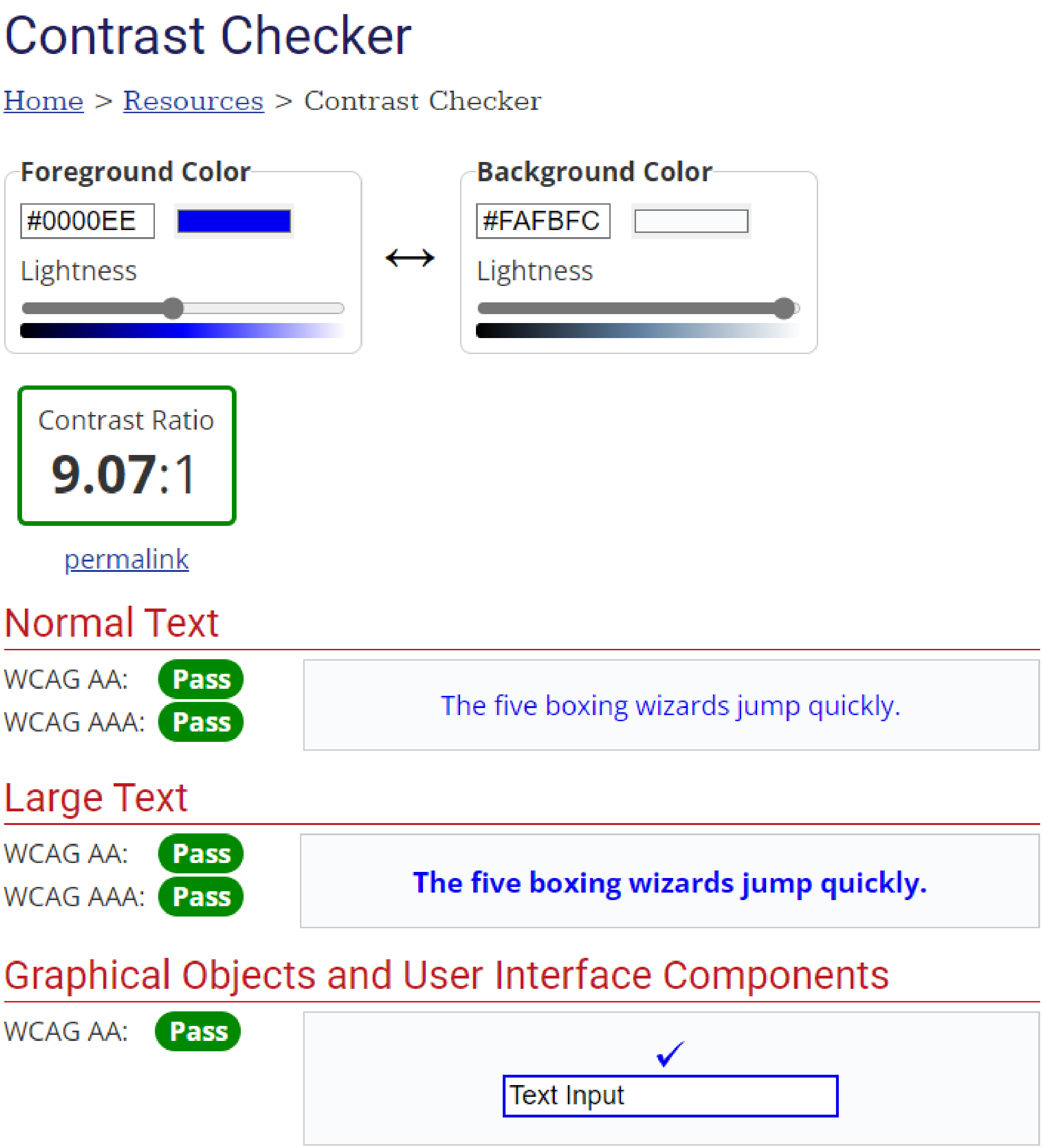
Figure 114. Main style contrast check.

6.3.2.7.2 Unused links



```
▼ Appearance
  ▶ color                      ■ rgb(0, 0, 238)
    background-attachment       scroll
    background-clip             border-box
  ▶ background-color           □ rgb(250, 251, 252)
```

Figure 115. Unused links colours.



Contrast Checker

Home > Resources > Contrast Checker

Foreground Color

#0000EE

Lightness

Background Color

#FAFBFC

Lightness

Contrast Ratio

9.07:1

permalink

Normal Text

WCAG AA: **Pass**

WCAG AAA: **Pass**

The five boxing wizards jump quickly.

Large Text

WCAG AA: **Pass**

WCAG AAA: **Pass**

The five boxing wizards jump quickly.

Graphical Objects and User Interface Components

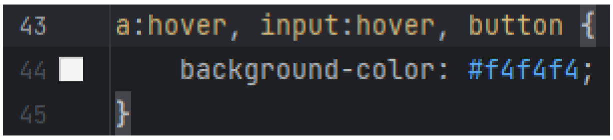
WCAG AA: **Pass**

✓

Text Input

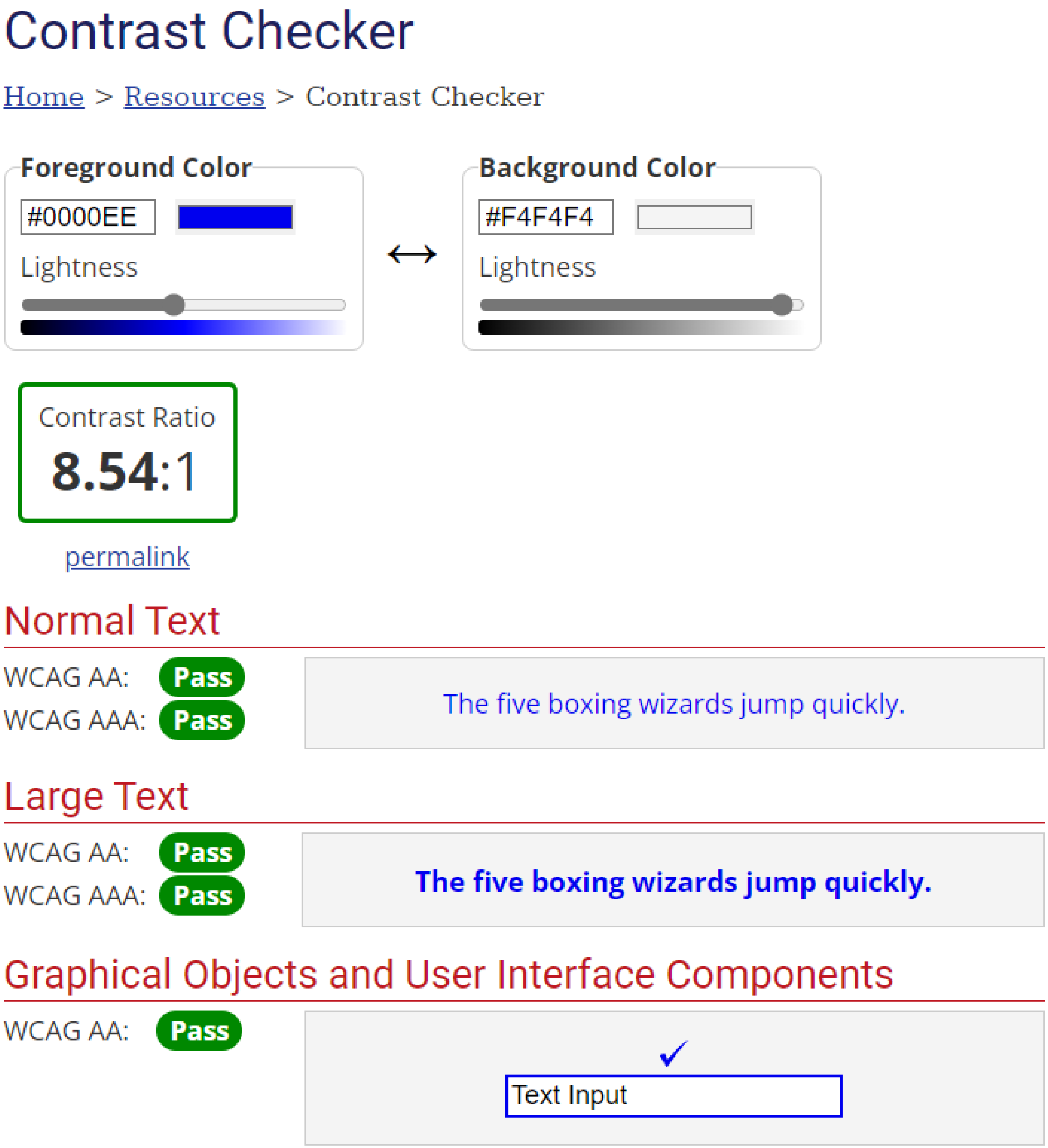
Figure 116. Unused links contrast check.

6.3.2.7.3 Unused links when hover



```
a:hover, input:hover, button {
    background-color: #f4f4f4;
}
```

Figure 117. Unused links when hover colours.



Contrast Checker

Home > Resources > Contrast Checker

Foreground Color: #0000EE

Lightness

Background Color: #F4F4F4

Lightness

Contrast Ratio
8.54:1

permalink

Normal Text

WCAG AA: Pass
WCAG AAA: Pass

The five boxing wizards jump quickly.

Large Text

WCAG AA: Pass
WCAG AAA: Pass

The five boxing wizards jump quickly.

Graphical Objects and User Interface Components

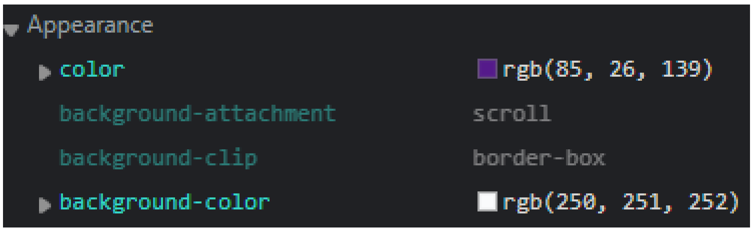
WCAG AA: Pass

✓

Text Input

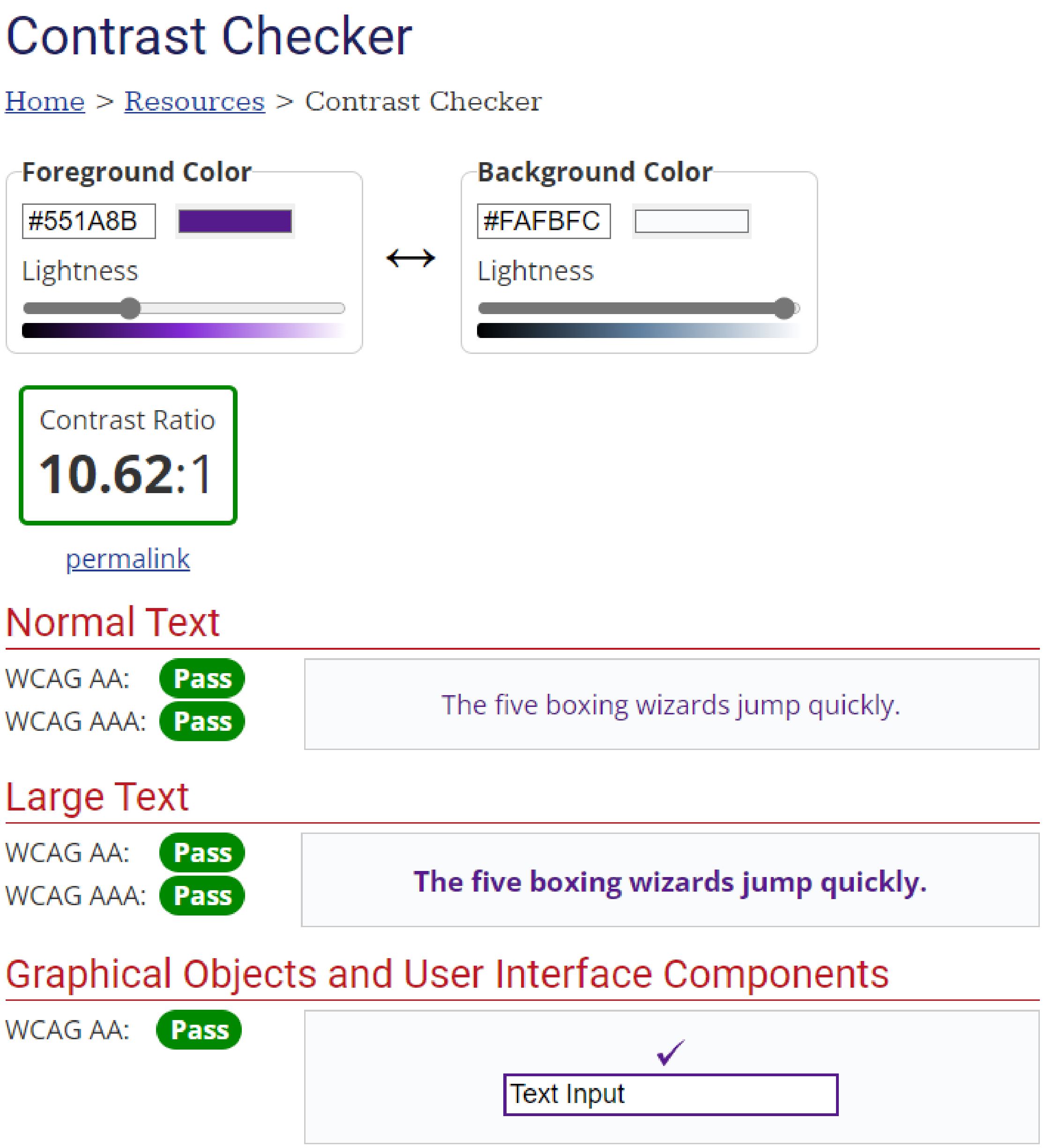
Figure 118. Unused links when hover contrast check.

6.3.2.7.4 Used links



```
▾ Appearance
  ▸ color                       ■ rgb(85, 26, 139)
    background-attachment       scroll
    background-clip             border-box
  ▸ background-color            □ rgb(250, 251, 252)
```

Figure 119. Used links colours.



Contrast Checker

Home > Resources > Contrast Checker

Foreground Color: #551A8B

Lightness

Background Color: #FAFBFC

Lightness

Contrast Ratio
10.62:1

permalink

Normal Text

WCAG AA: Pass
WCAG AAA: Pass

The five boxing wizards jump quickly.

Large Text

WCAG AA: Pass
WCAG AAA: Pass

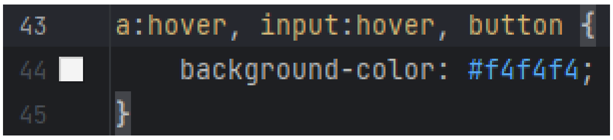
The five boxing wizards jump quickly.

Graphical Objects and User Interface Components

WCAG AA: Pass

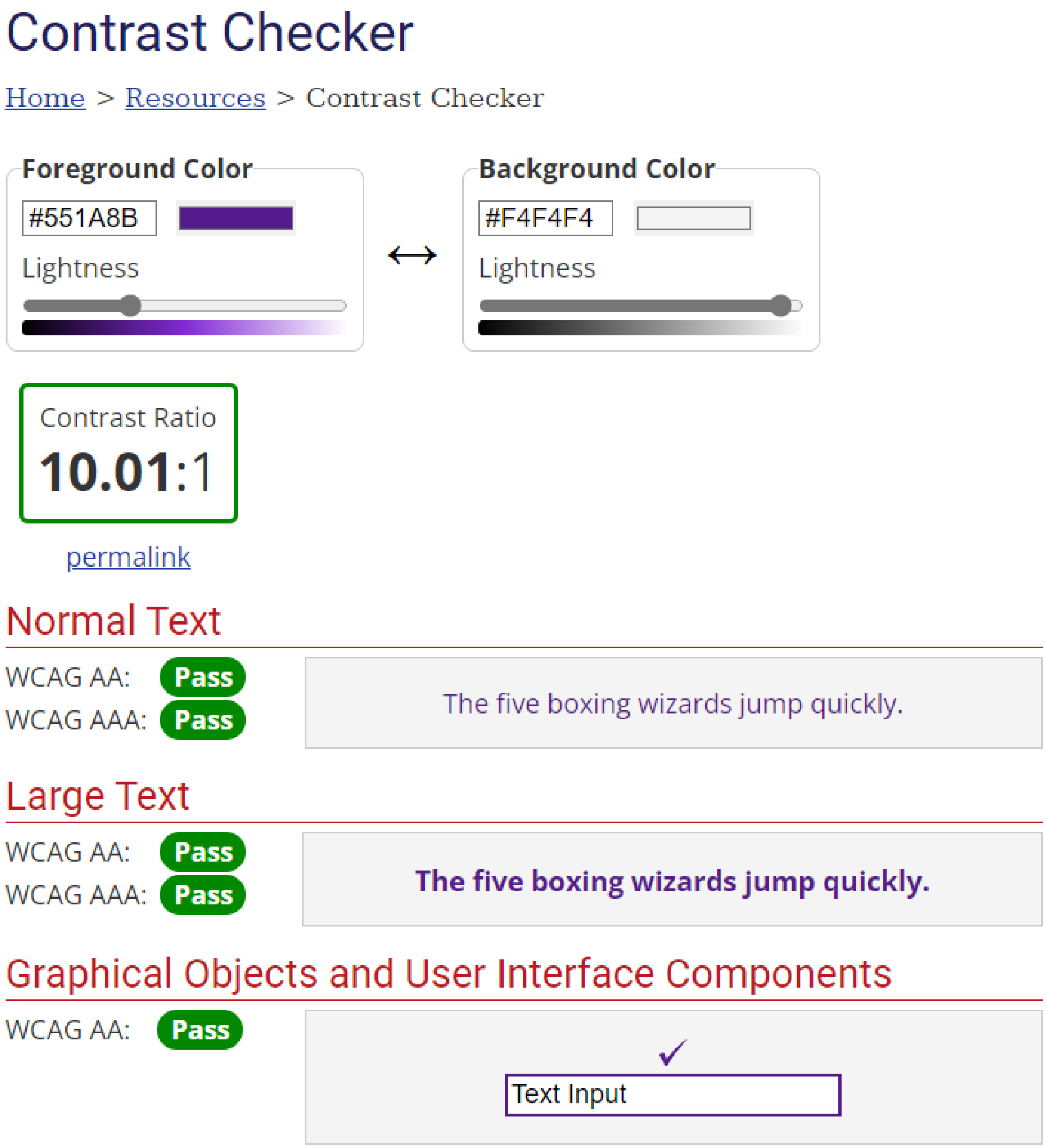
Figure 120. Used links contrast check.

6.3.2.7.5 Used links when hover



```
a:hover, input:hover, button {
    background-color: #f4f4f4;
}
```

Figure 121. Used links when hover colours.



Contrast Checker

Home > Resources > Contrast Checker

Foreground Color: #551A8B
Lightness

Background Color: #F4F4F4
Lightness

Contrast Ratio
10.01:1

permalink

Normal Text

WCAG AA: Pass
WCAG AAA: Pass

The five boxing wizards jump quickly.

Large Text

WCAG AA: Pass
WCAG AAA: Pass

The five boxing wizards jump quickly.

Graphical Objects and User Interface Components

WCAG AA: Pass

✓

Text Input

Figure 122. Used links when hover contrast check.

7 System documentation

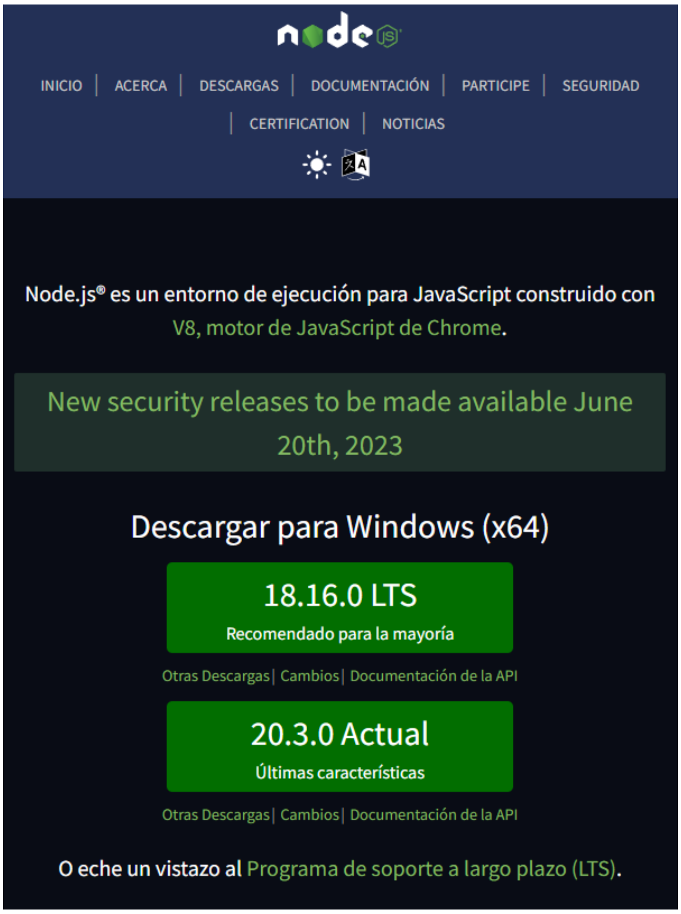
7.1 Manual of deployment on premises with Windows

ShareYASHE can be deployed in any machine capable of running Node.js. This includes Windows, Linux, macOS, and any other Unix-like Systems. Node.js is designed to be a cross-platform. In this chapter, we will see how to deploy ShareYASHE both in Windows and Linux.

In this manual, we will be using a Windows 11 Pro 22H2 machine on premises.

7.1.1 Install Node.js and NPM

The first step for deploying ShareYASHE is installing Node.js, and its package manager: NPM. Be can obtain the installable file from the Node.js Website[Node23]. At the moment of the development of ShareYASHE, Node.js v18 was used. However, you can try installing another version, but it is unsure that ShareYASHE would run correctly.



node js
INICIO | ACERCA | DESCARGAS | DOCUMENTACIÓN | PARTICIPE | SEGURIDAD
| CERTIFICATION | NOTICIAS

Node.js® es un entorno de ejecución para JavaScript construido con V8, motor de JavaScript de Chrome.

New security releases to be made available June 20th, 2023

Descargar para Windows (x64)

18.16.0 LTS
Recomendado para la mayoría

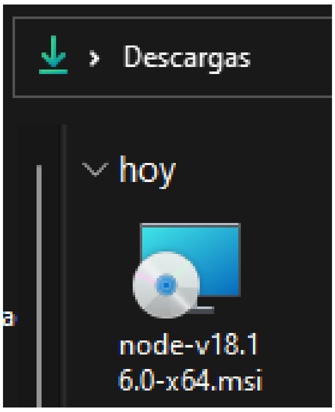
Otras Descargas | Cambios | Documentación de la API

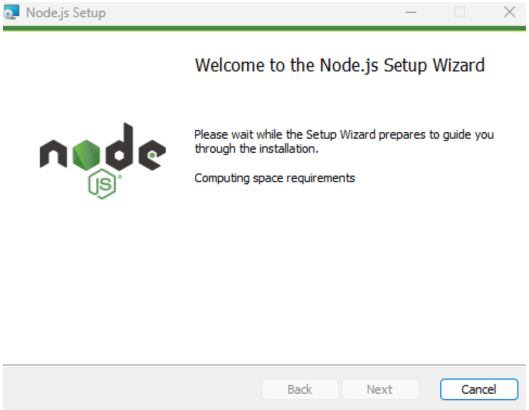
20.3.0 Actual
Últimas características

Otras Descargas | Cambios | Documentación de la API

O eche un vistazo al Programa de soporte a largo plazo (LTS).

When the installable file of Node.js is downloaded, we must execute it in order to install Node.js in the machine.

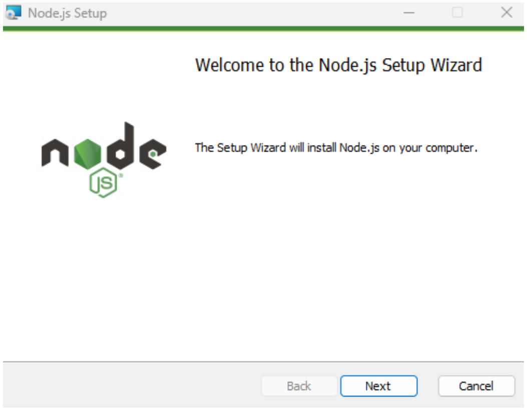




Welcome to the Node.js Setup Wizard

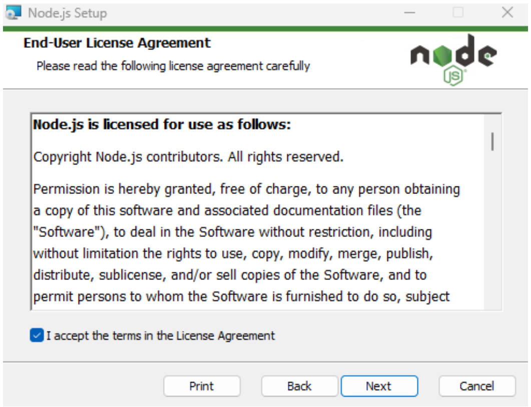
Please wait while the Setup Wizard prepares to guide you through the installation.

Computing space requirements



Welcome to the Node.js Setup Wizard

The Setup Wizard will install Node.js on your computer.



End-User License Agreement

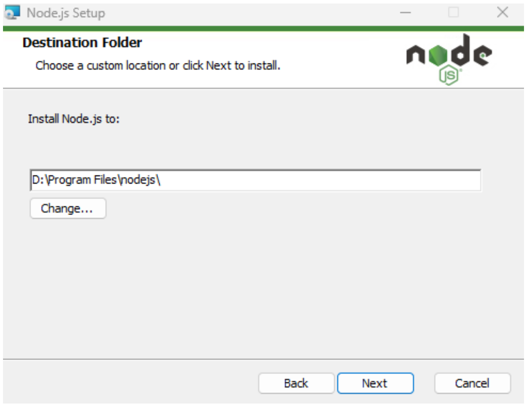
Please read the following license agreement carefully

Node.js is licensed for use as follows:

Copyright Node.js contributors. All rights reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject

☑ I accept the terms in the License Agreement

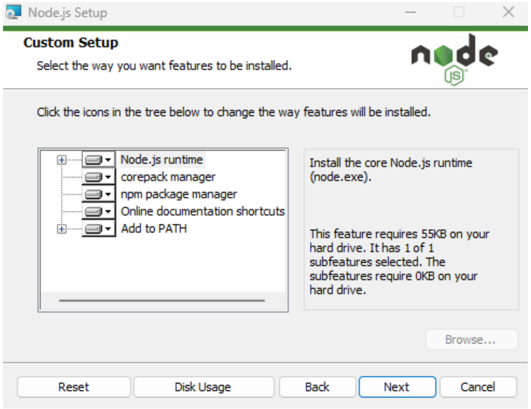


Destination Folder

Choose a custom location or click Next to install.

Install Node.js to:

D:\Program Files\nodejs\



Custom Setup

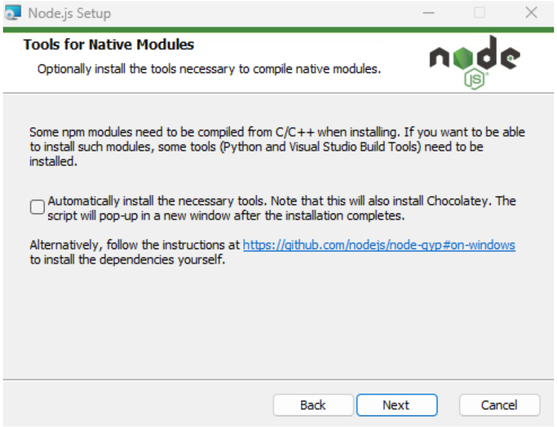
Select the way you want features to be installed.

Click the icons in the tree below to change the way features will be installed.

- Node.js runtime
- corepack manager
- npm package manager
- Online documentation shortcuts
- Add to PATH

Install the core Node.js runtime (node.exe).

This feature requires 55KB on your hard drive. It has 1 of 1 subfeatures selected. The subfeatures require 0KB on your hard drive.



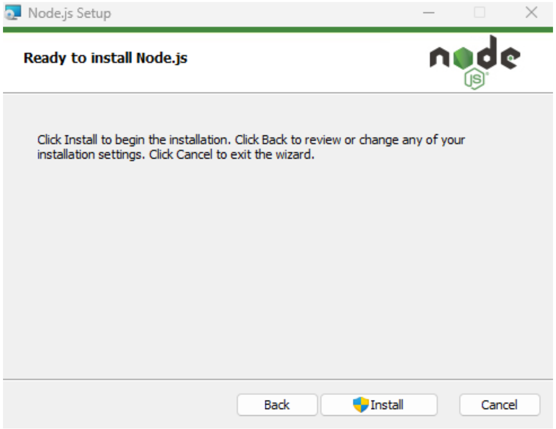
Tools for Native Modules

Optionally install the tools necessary to compile native modules.

Some npm modules need to be compiled from C/C++ when installing. If you want to be able to install such modules, some tools (Python and Visual Studio Build Tools) need to be installed.

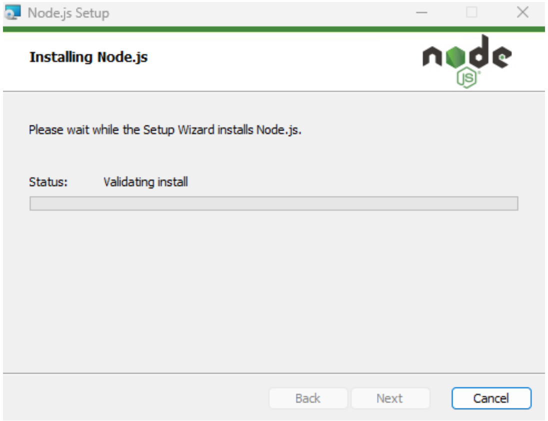
□ Automatically install the necessary tools. Note that this will also install Chocolatey. The script will pop-up in a new window after the installation completes.

Alternatively, follow the instructions at https://github.com/nodejs/node-gyp#on-windows to install the dependencies yourself.



Ready to install Node.js

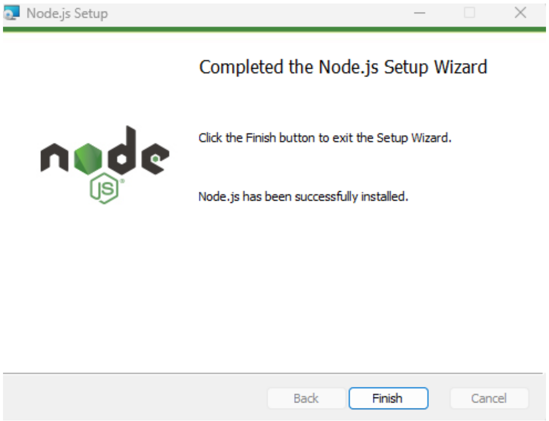
Click Install to begin the installation. Click Back to review or change any of your installation settings. Click Cancel to exit the wizard.



Installing Node.js

Please wait while the Setup Wizard installs Node.js.

Status: Validating install

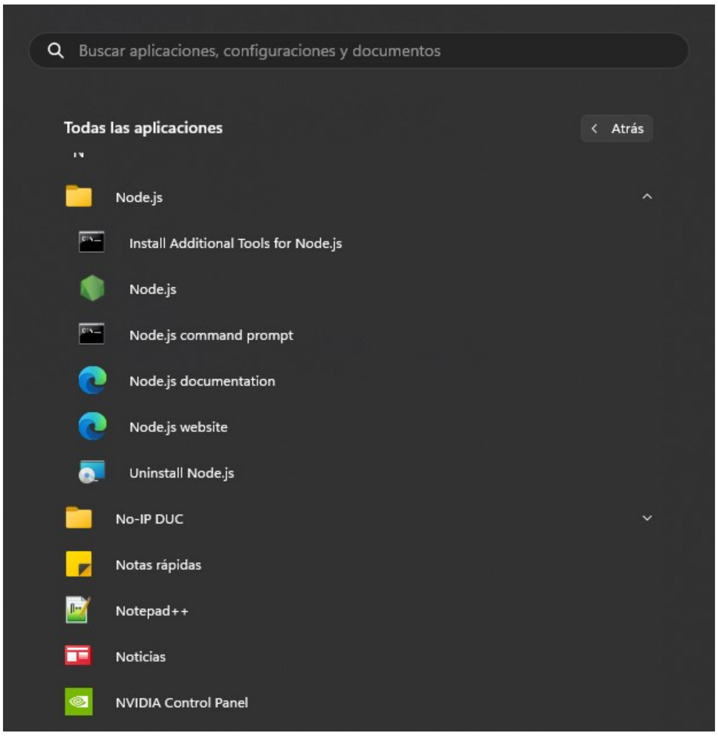


Completed the Node.js Setup Wizard

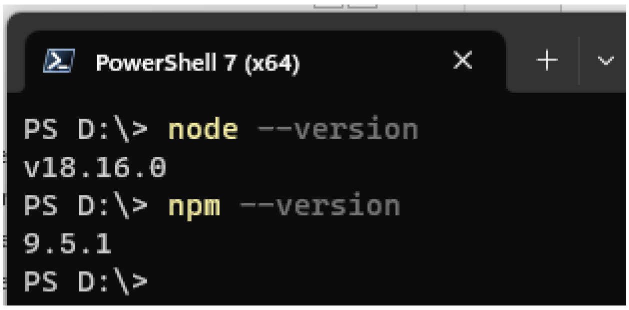
Click the Finish button to exit the Setup Wizard.

Node.js has been successfully installed.

At this point, Node.js would be installed and the command prompt could be accessible from the programs list.



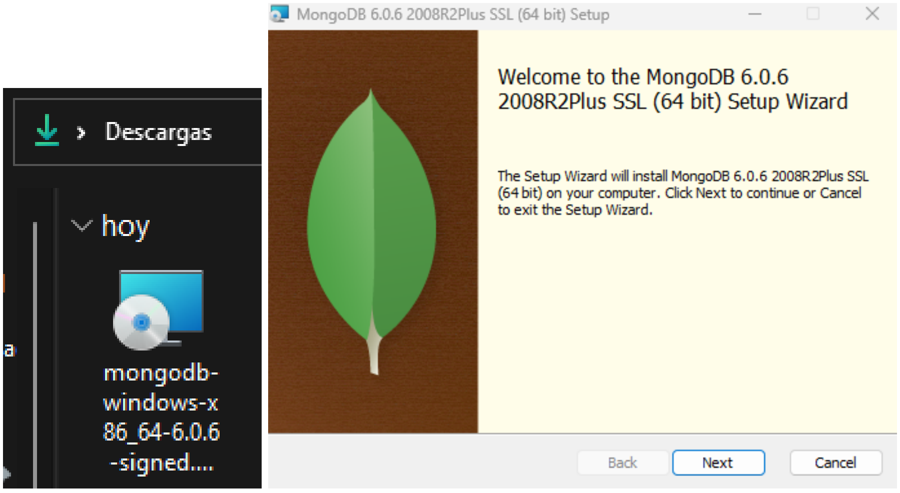
We may prefer access the Node.js functionality either through the native command prompt of Windows, or through the PowerShell 7. We can check the installation of Node.js and NPM with the following commands.



```
PS D:\> node --version
v18.16.0
PS D:\> npm --version
9.5.1
PS D:\>
```

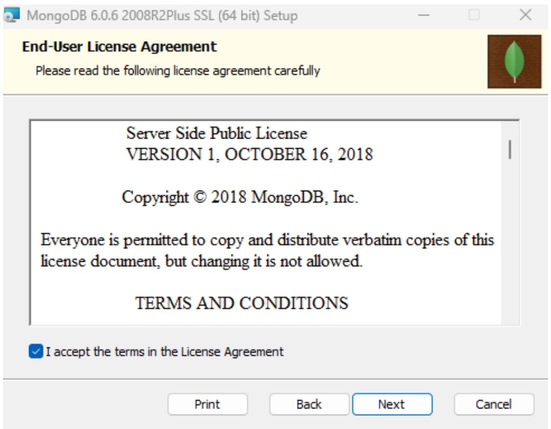
7.1.2 Install MongoDB and run it

ShareYASHE needs a MongoDB for its persistence. In this chapter, we will be deploying a local instance of MongoDB. Firstly, we may download the MongoDB Community Edition installable from its Website[Mongo23].


Descargas
hoy
mongodb-windows-x86_64-6.0.6-signed....
MongoDB 6.0.6 2008R2Plus SSL (64 bit) Setup
Back
Next
Cancel

Welcome to the MongoDB 6.0.6 2008R2Plus SSL (64 bit) Setup Wizard

The Setup Wizard will install MongoDB 6.0.6 2008R2Plus SSL (64 bit) on your computer. Click Next to continue or Cancel to exit the Setup Wizard.



End-User License Agreement

Please read the following license agreement carefully

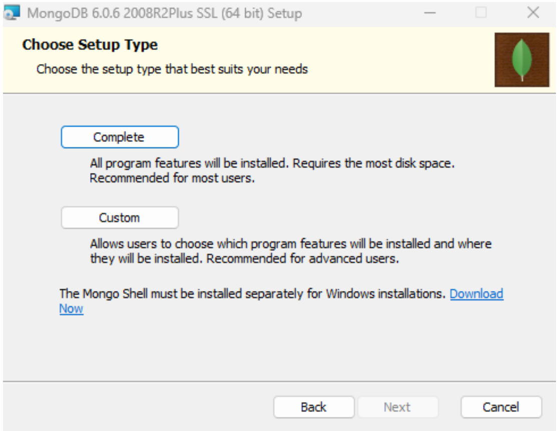
Server Side Public License
VERSION 1, OCTOBER 16, 2018

Copyright © 2018 MongoDB, Inc.

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

TERMS AND CONDITIONS

☑ I accept the terms in the License Agreement



Choose Setup Type

Choose the setup type that best suits your needs

Complete

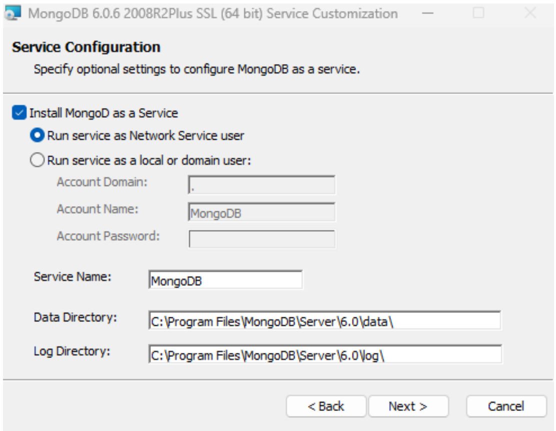
All program features will be installed. Requires the most disk space. Recommended for most users.

Custom

Allows users to choose which program features will be installed and where they will be installed. Recommended for advanced users.

The Mongo Shell must be installed separately for Windows installations. Download Now

We select the "Complete" option.



MongoDB 6.0.6 2008R2Plus SSL (64 bit) Service Customization

Service Configuration

Specify optional settings to configure MongoDB as a service.

☑ Install MongoD as a Service

● Run service as Network Service user

○ Run service as a local or domain user:

Account Domain: .

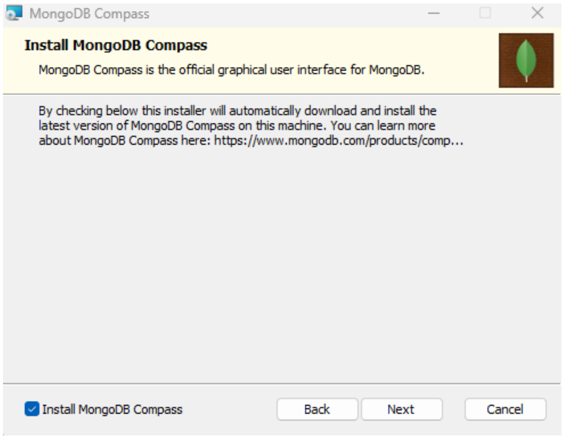
Account Name: MongoDB

Account Password:

Service Name: MongoDB

Data Directory: C:\Program Files\MongoDB\Server\6.0\data\

Log Directory: C:\Program Files\MongoDB\Server\6.0\log\



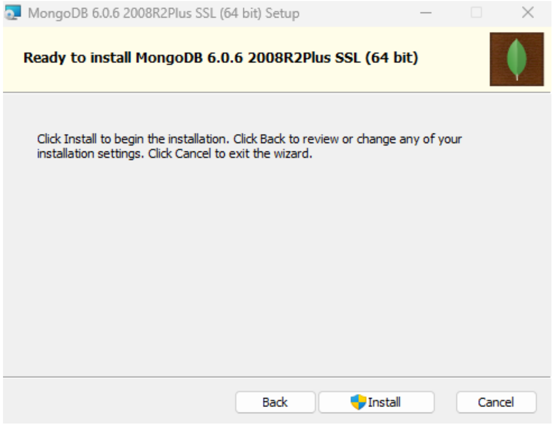
MongoDB Compass

Install MongoDB Compass

MongoDB Compass is the official graphical user interface for MongoDB.

By checking below this installer will automatically download and install the latest version of MongoDB Compass on this machine. You can learn more about MongoDB Compass here: https://www.mongodb.com/products/comp...

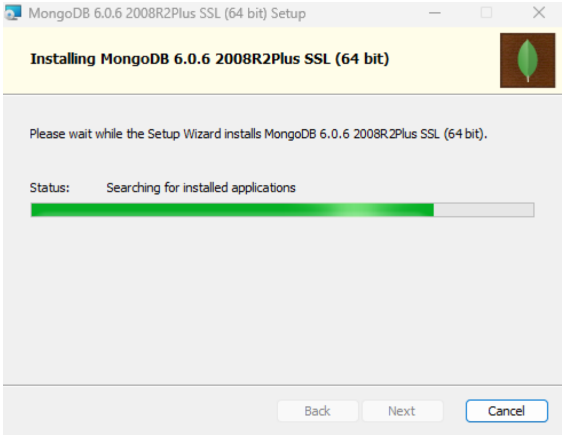
☑ Install MongoDB Compass



MongoDB 6.0.6 2008R2Plus SSL (64 bit) Setup

Ready to install MongoDB 6.0.6 2008R2Plus SSL (64 bit)

Click Install to begin the installation. Click Back to review or change any of your installation settings. Click Cancel to exit the wizard.

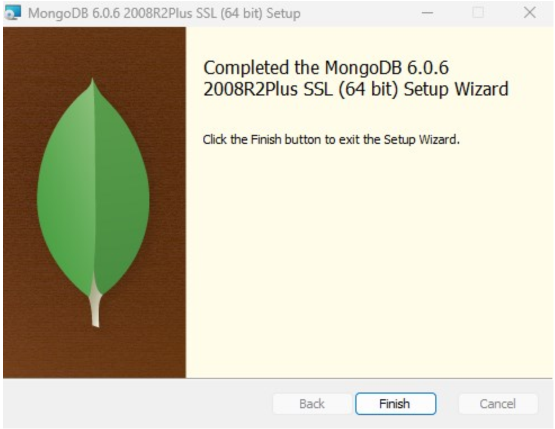


MongoDB 6.0.6 2008R2Plus SSL (64 bit) Setup

Installing MongoDB 6.0.6 2008R2Plus SSL (64 bit)

Please wait while the Setup Wizard installs MongoDB 6.0.6 2008R2Plus SSL (64 bit).

Status: Searching for installed applications



MongoDB 6.0.6 2008R2Plus SSL (64 bit) Setup

Completed the MongoDB 6.0.6 2008R2Plus SSL (64 bit) Setup Wizard

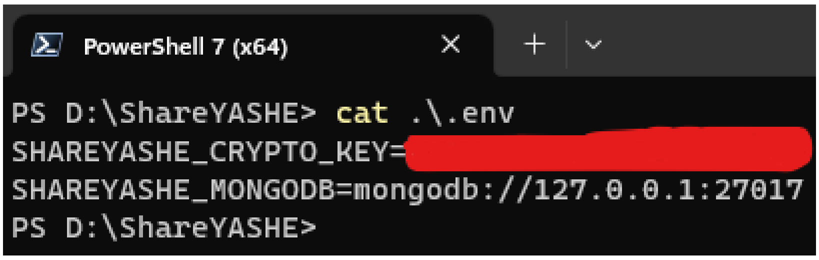
Click the Finish button to exit the Setup Wizard.

At this point, MongoDB would be fully installed, and its service would start with each System startup. Then, we can refer to the MongoDB database with the loopback address.

7.1.3 Obtain the ShareYASHE source code, customise its environment parameters, and run it

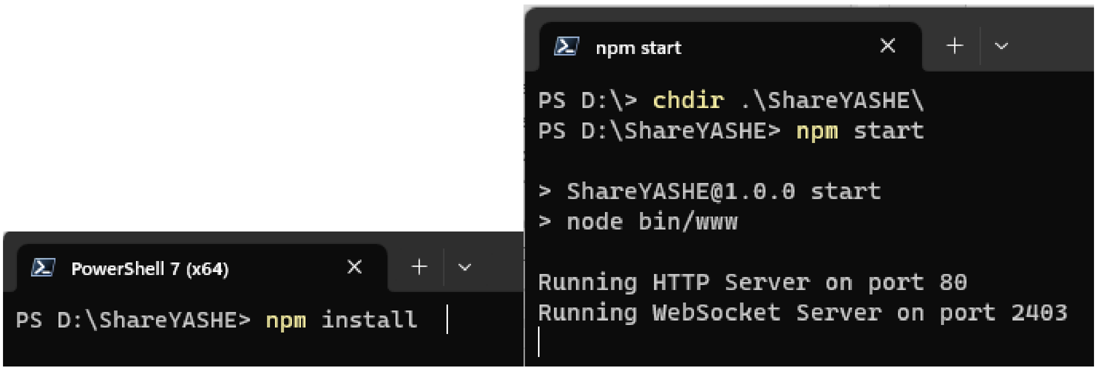
The code of ShareYASHE is published in a GitHub repository[Alv23] as an open-source project. We can simply download it and place it in a folder.

In the ".env" file, in the root of the project, we need to specify two parameters: SHAREYASHE_CRYPTO_KEY (an entropy word, for passwords to be encrypted), and SHAREYASHE_MONGODB (the reference to the MongoDB database).



```
PS D:\ShareYASHE> cat .\.env
SHAREYASHE_CRYPTO_KEY=[redacted]
SHAREYASHE_MONGODB=mongodb://127.0.0.1:27017
PS D:\ShareYASHE>
```

After installing the NPM packages, we just call the script "npm start", after calling "npm install", whose steps are declared in the package.json file. "npm install" only needs to be called the first time you deploy, or every time that you change the packages.json file.



```
PS D:\ShareYASHE> npm install
```

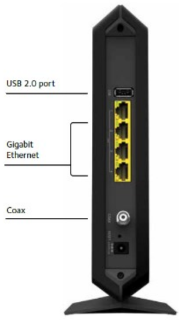
```
PS D:\> chdir .\ShareYASHE\
PS D:\ShareYASHE> npm start

> ShareYASHE@1.0.0 start
> node bin/www

Running HTTP Server on port 80
Running WebSocket Server on port 2403
```

At this point, ShareYASHE would be running on your local network.

7.1.4 Access from the external network

In pursue of users from the external network accessing your ShareYASHE deployment, you must ensure that the ports are opened, both 80 and 2403, both UDP and TCP. You must also know your public IP, and you may use a DNS for making the URI of you ShareYASHE deployment to be friendlier.

7.1.5 Ensure external projection capability

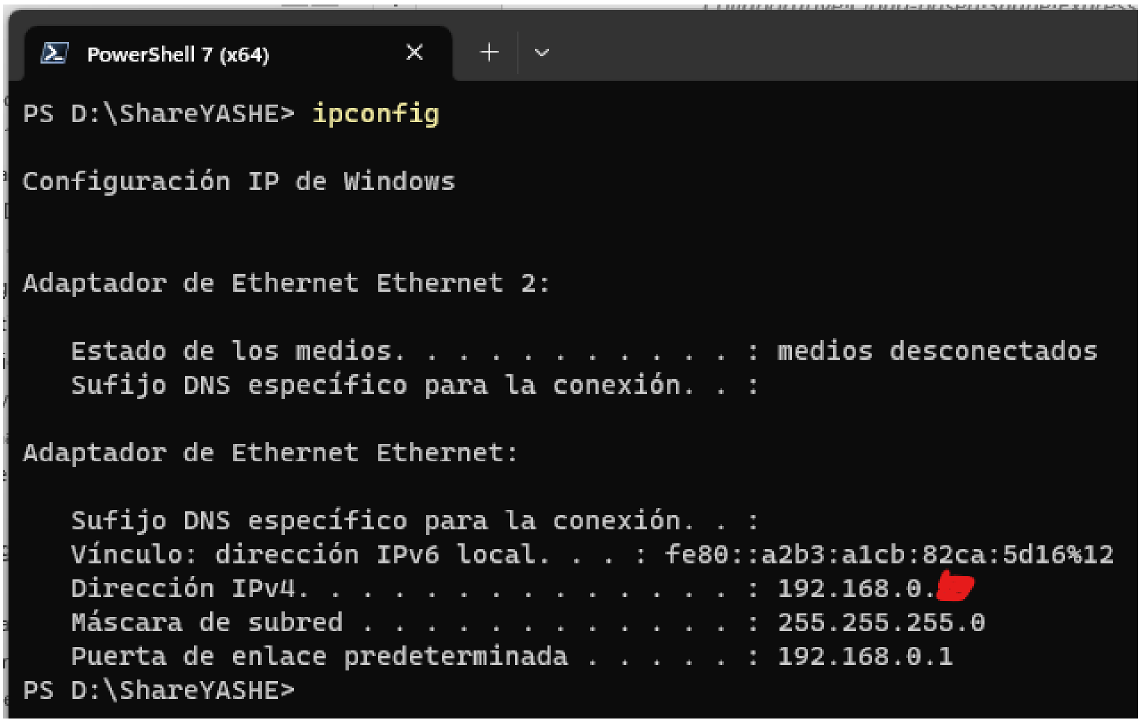


Deployment tools, or cloud-based machines, offer a variety of ways to open ports. Nevertheless, here we will be doing it as if we were deploying ShareYASHE on premises.

First of all, we must ensure that the machine, in which ShareYASHE is deployed, is connected via an Ethernet cable to a router which is directly connected to the Internet. Your router may be connected to the Internet through a coaxial cable, like that of the picture below, or maybe through a satellite antenna.

If the machine is connected in series to a chain of routers, until the last router is connected directly to the Internet, we must open the 80 and 2403 ports of each of those routers.

We may know the default access to the router configuration through the “ipconfig” command.



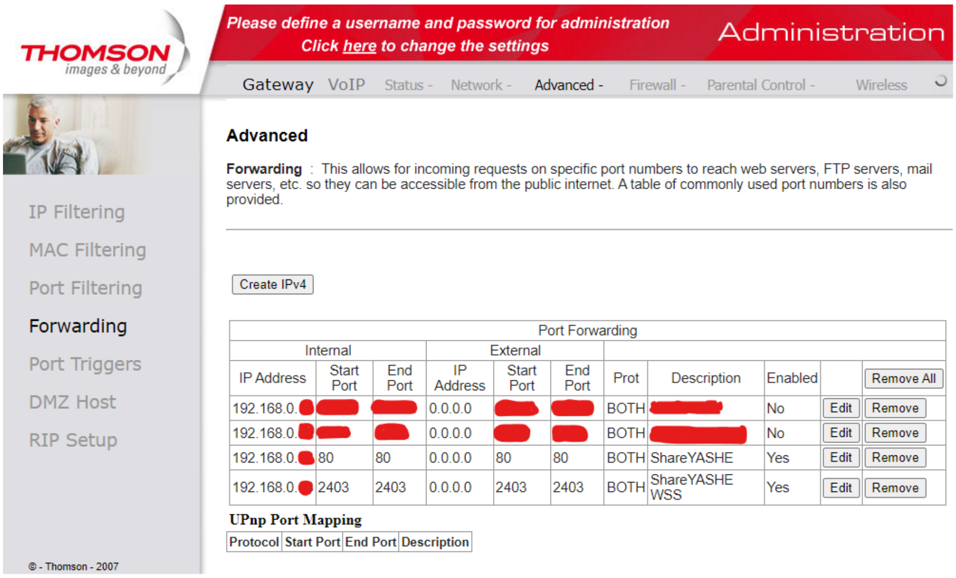
```
PS D:\ShareYASHE> ipconfig

Configuración IP de Windows


Adaptador de Ethernet Ethernet 2:

   Estado de los medios. . . . . . . . . . . : medios desconectados
   Sufijo DNS específico para la conexión. . :

Adaptador de Ethernet Ethernet:

   Sufijo DNS específico para la conexión. . :
   Vínculo: dirección IPv6 local. . . : fe80::a2b3:a1cb:82ca:5d16%12
   Dirección IPv4. . . . . . . . . . . . . . : 192.168.0.[redacted]
   Máscara de subred . . . . . . . . . . . . : 255.255.255.0
   Puerta de enlace predeterminada . . . . . : 192.168.0.1
PS D:\ShareYASHE>
```

We can then access to that configuration using the address http://192.168.0.1/, because that is the address that the ipconfig command threw to me, which may be different in your case.

It is probable that a login form may be shown to us. In order to access the configuration, we must know the credentials, which may be specified in the router’s manual.

7.1.6 Open ports

Then, we must find the forwarding configuration and, using the internal IPv4 address of the machine in which ShareYASHE is running (the address that I censored in the screenshot of the ipconfig command), stablish both ports to be forwarded.



Advanced

Forwarding : This allows for incoming requests on specific port numbers to reach web servers, FTP servers, mail servers, etc. so they can be accessible from the public internet. A table of commonly used port numbers is also provided.

Port Forwarding

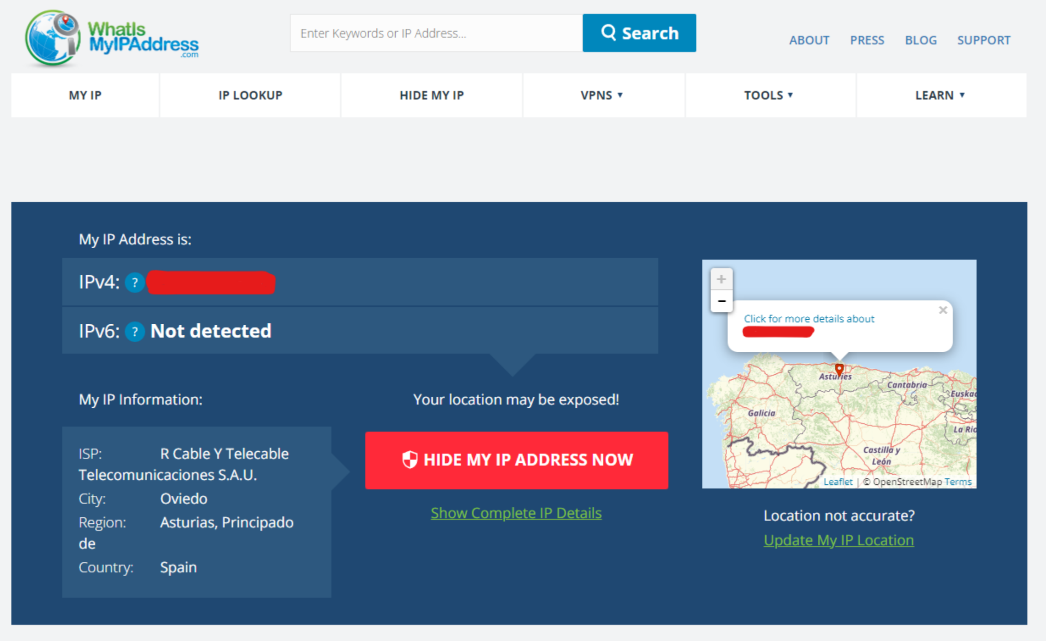
Internal			External							
IP Address	Start Port	End Port	IP Address	Start Port	End Port	Prot	Description	Enabled		Remove All
192.168.0.[redacted]	[redacted]	[redacted]	0.0.0.0	[redacted]	[redacted]	BOTH	[redacted]	No	Edit	Remove
192.168.0.[redacted]	[redacted]	[redacted]	0.0.0.0	[redacted]	[redacted]	BOTH	[redacted]	No	Edit	Remove
192.168.0.[redacted]	80	80	0.0.0.0	80	80	BOTH	ShareYASHE	Yes	Edit	Remove
192.168.0.[redacted]	2403	2403	0.0.0.0	2403	2403	BOTH	ShareYASHE WSS	Yes	Edit	Remove

UPnp Port Mapping

Protocol	Start Port	End Port	Description

7.1.7 Get your external address

We must also find our external IPv4 address, so users can use it to connect to our ShareYASHE deployment. We can do it with several tools, such as WhatIsMyIPAddress[WIMIPA23].



My IP Address is:

IPv4: [redacted]

IPv6: **Not detected**

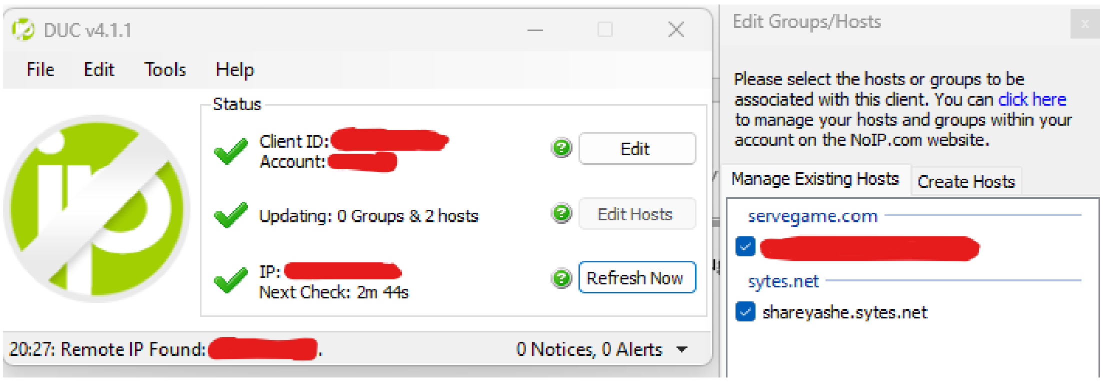
My IP Information:

ISP:	R Cable Y Telecable Telecomunicaciones S.A.U.
City:	Oviedo
Region:	Asturias, Principado de
Country:	Spain

For security reasons, I must censor my external IPv4 address. Imagine that it is: 55.55.55.55. Users can then access to you ShareYASHE deployment through this link: http://55.55.55.55/.

7.1.8 Set up a DNS

Furthermore, we can use a DNS tool in order to hide our external IPv4 address. In my case, I am using No-IP DUC, so I can use the following domain name for my ShareYASHE deployment on premises: http://shareyashe.sytes.net.



DUC v4.1.1

File Edit Tools Help

Status

Client ID: [redacted]
Account: [redacted]

Edit

Updating: 0 Groups & 2 hosts

Edit Hosts

IP: [redacted]
Next Check: 2m 44s

Refresh Now

20:27: Remote IP Found: [redacted].

0 Notices, 0 Alerts

Edit Groups/Hosts

Please select the hosts or groups to be associated with this client. You can click here to manage your hosts and groups within your account on the NoIP.com website.

Manage Existing Hosts | Create Hosts

servegame.com

☑ [redacted]

sytes.net

☑ shareyashe.sytes.net

This does not prevent hackers to know your external IPv4 address, because they can ping the domain name and obtain your address, but it makes the domain name more friendly for the users to remind it.

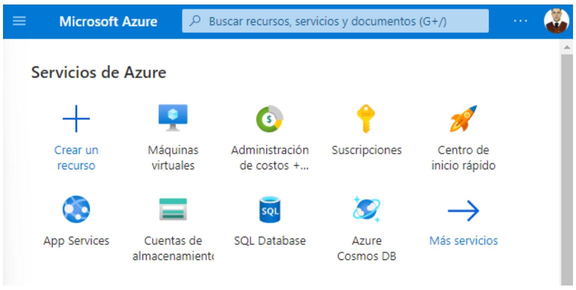
7.2 Manual of deployment on the cloud with Linux

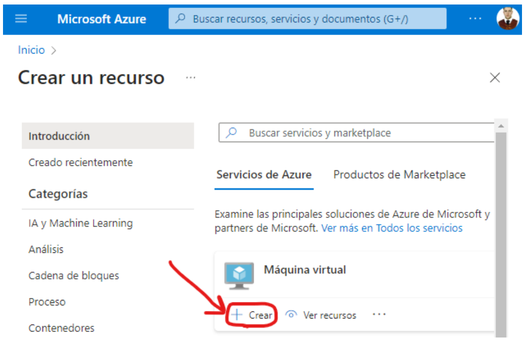
ShareYASHE can be deployed in any machine capable of running Node.js. This includes Windows, Linux, macOS, and any other Unix-like Systems. Node.js is designed to be a cross-platform. In this chapter, we will see how to deploy ShareYASHE both in Windows and Linux.

In this manual, we will be using an Ubuntu Server 20.04 machine on a Microsoft Azure cloud.

7.2.1 Set up an account in a cloud Website

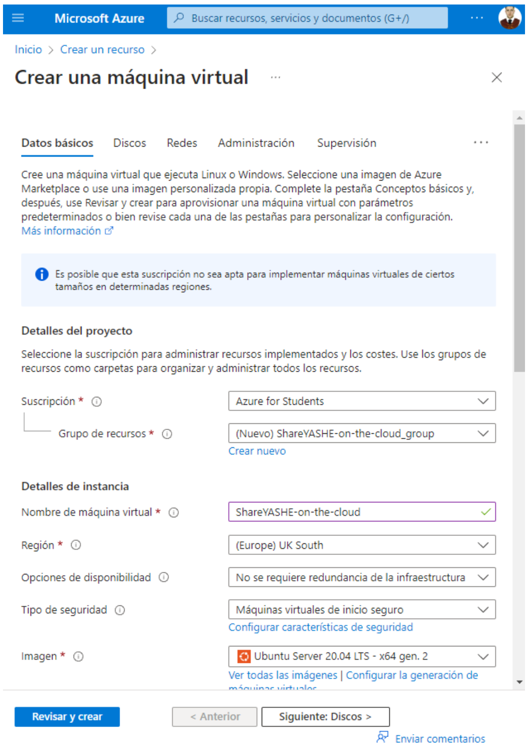
Firstly, we need an account in a cloud management Website. My account from the University of Oviedo grants me some credits for Microsoft Azure, so I will be using this one. Having received the necessary credits, we must access the portal for managing cloud machines[Azure23]. We create a cloud machine with the option for creating a resource (in the screenshot: "Crear un recurso").





7.2.2 Create a cloud machine

Microsoft Azure lets us create a cloud machine with a simple form. In this case, I will put a name for the virtual machine: “ShareYASHE-on-the-cloud”, and I will press the blue button at the bottom.



Crear una máquina virtual

Cree una máquina virtual que ejecuta Linux o Windows. Seleccione una imagen de Azure Marketplace o use una imagen personalizada propia. Complete la pestaña Conceptos básicos y, después, use Revisar y crear para aprovisionar una máquina virtual con parámetros predeterminados o bien revise cada una de las pestañas para personalizar la configuración. Más información

Es posible que esta suscripción no sea apta para implementar máquinas virtuales de ciertos tamaños en determinadas regiones.

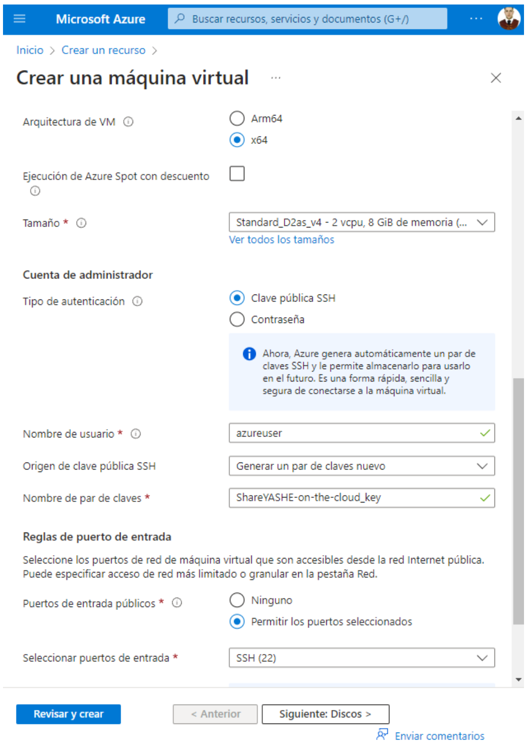
Detalles del proyecto

Seleccione la suscripción para administrar recursos implementados y los costes. Use los grupos de recursos como carpetas para organizar y administrar todos los recursos.

Suscripción * Azure for Students
Grupo de recursos * (Nuevo) ShareYASHE-on-the-cloud_group
Crear nuevo

Detalles de instancia

Nombre de máquina virtual * ShareYASHE-on-the-cloud ✓
Región * (Europe) UK South
Opciones de disponibilidad No se requiere redundancia de la infraestructura
Tipo de seguridad Máquinas virtuales de inicio seguro
Configurar características de seguridad
Imagen * Ubuntu Server 20.04 LTS - x64 gen. 2
Ver todas las imágenes | Configurar la generación de máquinas virtuales



Crear una máquina virtual

Arquitectura de VM ○ Arm64 ● x64
Ejecución de Azure Spot con descuento □
Tamaño * Standard_D2as_v4 - 2 vcpu, 8 GiB de memoria (...
Ver todos los tamaños

Cuenta de administrador

Tipo de autenticación ● Clave pública SSH ○ Contraseña

Ahora, Azure genera automáticamente un par de claves SSH y le permite almacenarlo para usarlo en el futuro. Es una forma rápida, sencilla y segura de conectarse a la máquina virtual.

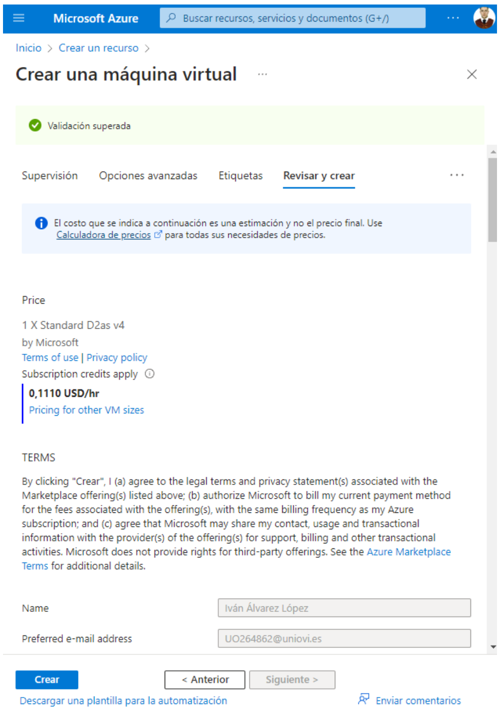
Nombre de usuario * azureuser ✓
Origen de clave pública SSH Generar un par de claves nuevo
Nombre de par de claves * ShareYASHE-on-the-cloud_key ✓

Reglas de puerto de entrada

Seleccione los puertos de red de máquina virtual que son accesibles desde la red Internet pública. Puede especificar acceso de red más limitado o granular en la pestaña Red.

Puertos de entrada públicos * ○ Ninguno ● Permitir los puertos seleccionados
Seleccionar puertos de entrada * SSH (22)

As we can see, running a machine on the cloud is not free. After clicking on “Create”, we must download the private SSH key, for accessing to the virtual machine.


Microsoft Azure
Buscar recursos, servicios y documentos (G+/)
Inicio > Crear un recurso >
Supervisión
Opciones avanzadas
Etiquetas
Revisar y crear
Crear
< Anterior
Siguiente >
Descargar una plantilla para la automatización
Enviar comentarios

Crear una máquina virtual

Validación superada

El costo que se indica a continuación es una estimación y no el precio final. Use Calculadora de precios para todas sus necesidades de precios.

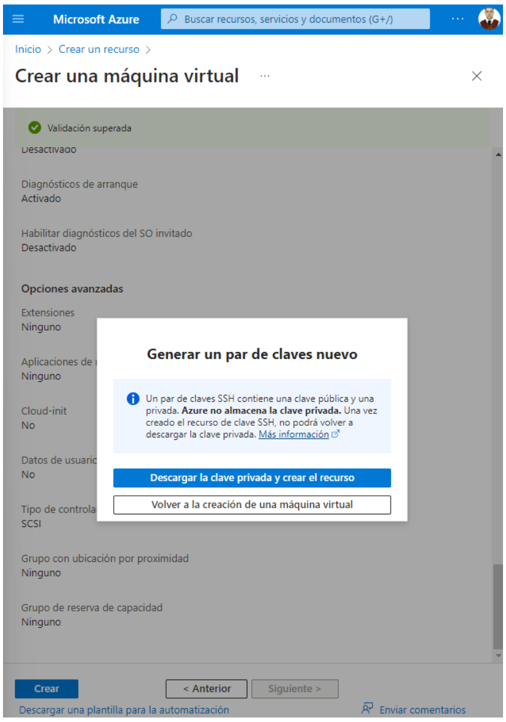
Price

1 X Standard D2as v4
by Microsoft
Terms of use | Privacy policy
Subscription credits apply
0,1110 USD/hr
Pricing for other VM sizes

TERMS

By clicking "Crear", I (a) agree to the legal terms and privacy statement(s) associated with the Marketplace offering(s) listed above; (b) authorize Microsoft to bill my current payment method for the fees associated with the offering(s), with the same billing frequency as my Azure subscription; and (c) agree that Microsoft may share my contact, usage and transactional information with the provider(s) of the offering(s) for support, billing and other transactional activities. Microsoft does not provide rights for third-party offerings. See the Azure Marketplace Terms for additional details.

Name Iván Álvarez López
Preferred e-mail address UO264862@uniovi.es


Microsoft Azure
Buscar recursos, servicios y documentos (G+/)
Inicio > Crear un recurso >
Descargar la clave privada y crear el recurso
Volver a la creación de una máquina virtual
Crear
< Anterior
Siguiente >
Descargar una plantilla para la automatización
Enviar comentarios

Crear una máquina virtual

Validación superada

Desactivado

Diagnósticos de arranque
Activado

Habilitar diagnósticos del SO invitado
Desactivado

Opciones avanzadas

Extensiones
Ninguno

Aplicaciones de
Ninguno

Cloud-init
No

Datos de usuario
No

Tipo de controla
SCSI

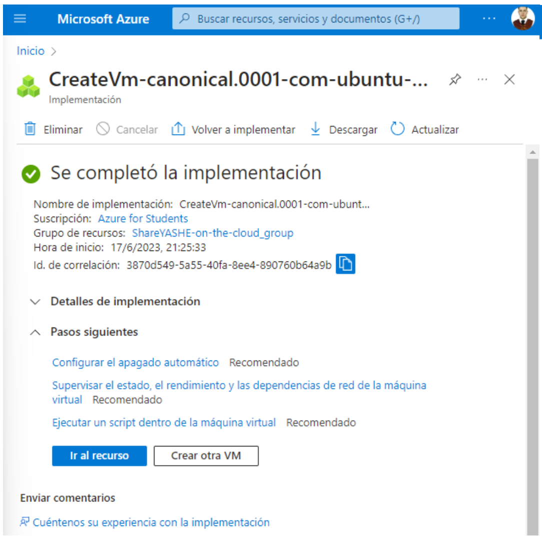
Grupo con ubicación por proximidad
Ninguno

Grupo de reserva de capacidad
Ninguno

Generar un par de claves nuevo

Un par de claves SSH contiene una clave pública y una privada. **Azure no almacena la clave privada.** Una vez creado el recurso de clave SSH, no podrá volver a descargar la clave privada. Más información

After a few minutes, the cloud would be finally created.



CreateVm-canonical.0001-com-ubuntu-...
Implementación

Se completó la implementación

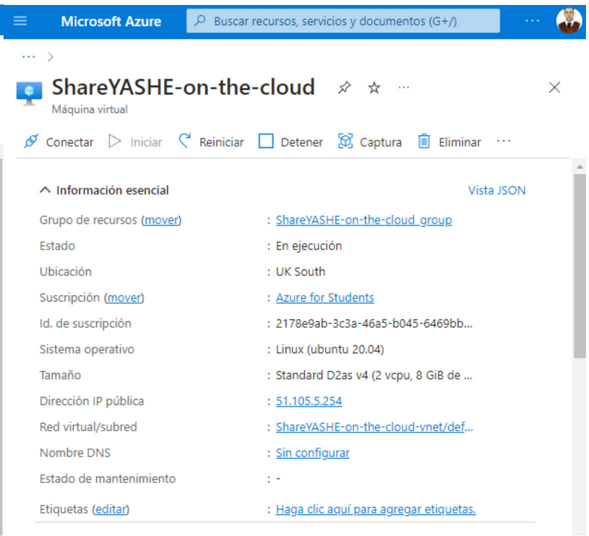
Nombre de implementación: CreateVm-canonical.0001-com-ubunt...
Suscripción: Azure for Students
Grupo de recursos: ShareYASHE-on-the-cloud_group
Hora de inicio: 17/6/2023, 21:25:33
Id. de correlación: 3870d549-5a55-40fa-8ee4-890760b64a9b

Detalles de implementación

Pasos siguientes

Configurar el apagado automático Recomendado
Supervisar el estado, el rendimiento y las dependencias de red de la máquina virtual Recomendado
Ejecutar un script dentro de la máquina virtual Recomendado

Enviar comentarios
Cuéntenos su experiencia con la implementación



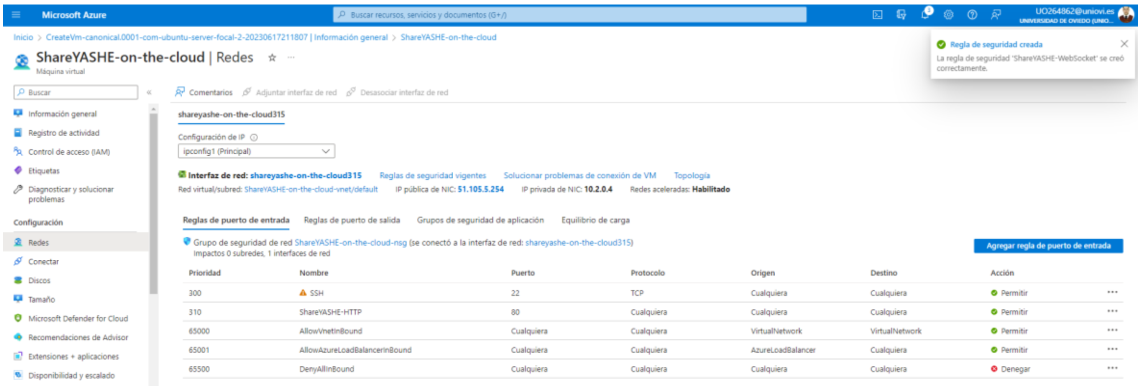
ShareYASHE-on-the-cloud
Máquina virtual

Información esencial

Grupo de recursos (mover)	: ShareYASHE-on-the-cloud_group
Estado	: En ejecución
Ubicación	: UK South
Suscripción (mover)	: Azure for Students
Id. de suscripción	: 2178e9ab-3c3a-46a5-b045-6469bb...
Sistema operativo	: Linux (ubuntu 20.04)
Tamaño	: Standard D2as v4 (2 vcpu, 8 GiB de ...
Dirección IP pública	: 51.105.5.254
Red virtual/subred	: ShareYASHE-on-the-cloud-vnet/def...
Nombre DNS	: Sin configurar
Estado de mantenimiento	: -
Etiquetas (editar)	: Haga clic aquí para agregar etiquetas.

7.2.3 Open ports

Before leaving the Microsoft Azure portal, we will open the ports needed for ShareYASHE: 80 and 2403. This purpose can be achieved in the "Networks" section, in the configuration of our cloud machine.


Microsoft Azure
Buscar recursos, servicios y documentos (G+/)
UO264862@uniovi.es
UNIVERSIDAD DE OVIEDO (UNIO...
Inicio > CreateVm-canonical.0001-com-ubuntu-server-focal-2-20230617211807 | Información general > ShareYASHE-on-the-cloud
Buscar
Información general
Registro de actividad
Control de acceso (IAM)
Etiquetas
Diagnosticar y solucionar problemas
Configuración
Redes
Conectar
Discos
Tamaño
Microsoft Defender for Cloud
Recomendaciones de Advisor
Extensiones + aplicaciones
Disponibilidad y escalado
Comentarios
Adjuntar interfaz de red
Desasociar interfaz de red
shareyashe-on-the-cloud315
Reglas de puerto de entrada
Reglas de puerto de salida
Grupos de seguridad de aplicación
Equilibrio de carga
Agregar regla de puerto de entrada

Regla de seguridad creada
La regla de seguridad 'ShareYASHE-WebSocket' se creó correctamente.

ShareYASHE-on-the-cloud | Redes
Máquina virtual

Configuración de IP: ipconfig1 (Principal)

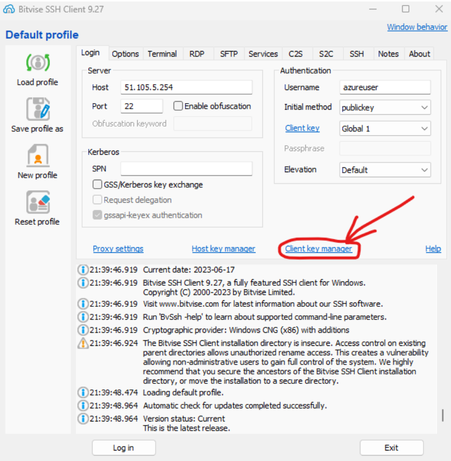
Interfaz de red: shareyashe-on-the-cloud315 Reglas de seguridad vigentes Solucionar problemas de conexión de VM Topología
Red virtual/subred: ShareYASHE-on-the-cloud-vnet/default IP pública de NIC: 51.105.5.254 IP privada de NIC: 10.2.0.4 Redes aceleradas: Habilitado

Grupo de seguridad de red ShareYASHE-on-the-cloud-nsg (se conectó a la interfaz de red: shareyashe-on-the-cloud315)
Impactos 0 subredes, 1 interfaces de red

Prioridad	Nombre	Puerto	Protocolo	Origen	Destino	Acción
300	⚠ SSH	22	TCP	Cualquiera	Cualquiera	✓ Permitir
310	ShareYASHE-HTTP	80	Cualquiera	Cualquiera	Cualquiera	✓ Permitir
65000	AllowVnetInBound	Cualquiera	Cualquiera	VirtualNetwork	VirtualNetwork	✓ Permitir
65001	AllowAzureLoadBalancerInBound	Cualquiera	Cualquiera	AzureLoadBalancer	Cualquiera	✓ Permitir
65500	DenyAllInBound	Cualquiera	Cualquiera	Cualquiera	Cualquiera	✗ Denegar

7.2.4 Access the cloud machine

This cloud machine can be accessed with tools such as Bitvise SSH Client[Bitvise23]. Having installed it on our local machine, we must import the private key that we downloaded while creating the cloud machine.

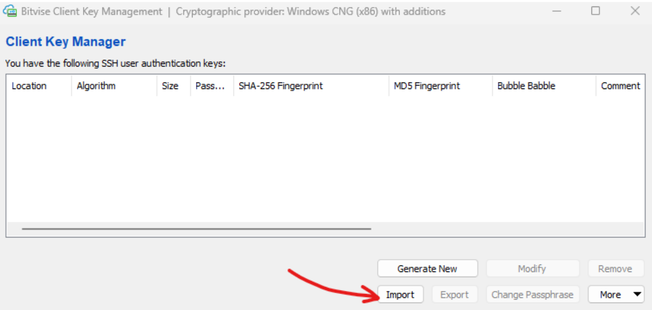


Server
Host: 51.105.5.254
Port: 22
☐ Enable obfuscation
Obfuscation keyword

Kerberos
SPN
☐ GSS/Kerberos key exchange
☐ Request delegation
☑ gssapi-keyex authentication

Authentication
Username: azureuser
Initial method: publickey
Client key: Global 1
Passphrase
Elevation: Default

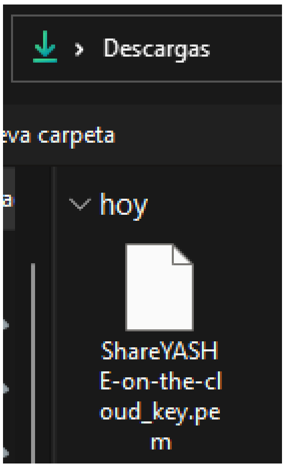
```
21:39:46.919  Current date: 2023-06-17
21:39:46.919  Bitvise SSH Client 9.27, a fully featured SSH client for Windows.
              Copyright (C) 2000-2023 by Bitvise Limited.
21:39:46.919  Visit www.bitvise.com for latest information about our SSH software.
21:39:46.919  Run 'BvSsh -help' to learn about supported command-line parameters.
21:39:46.919  Cryptographic provider: Windows CNG (x86) with additions
21:39:46.924  The Bitvise SSH Client installation directory is insecure. Access control on existing
              parent directories allows unauthorized rename access. This creates a vulnerability
              allowing non-administrative users to gain full control of the system. We highly
              recommend that you secure the ancestors of the Bitvise SSH Client installation
              directory, or move the installation to a secure directory.
21:39:48.474  Loading default profile.
21:39:48.964  Automatic check for updates completed successfully.
21:39:48.964  Version status: Current
              This is the latest release.
```

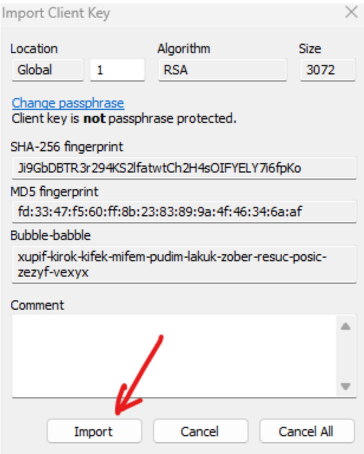

Bitvise Client Key Management | Cryptographic provider: Windows CNG (x86) with additions
Generate New
Modify
Remove
Import
Export
Change Passphrase
More

Client Key Manager

You have the following SSH user authentication keys:

Location	Algorithm	Size	Pass...	SHA-256 Fingerprint	MD5 Fingerprint	Bubble Babble	Comment





Location: Global 1
Algorithm: RSA
Size: 3072

Client key is **not** passphrase protected.

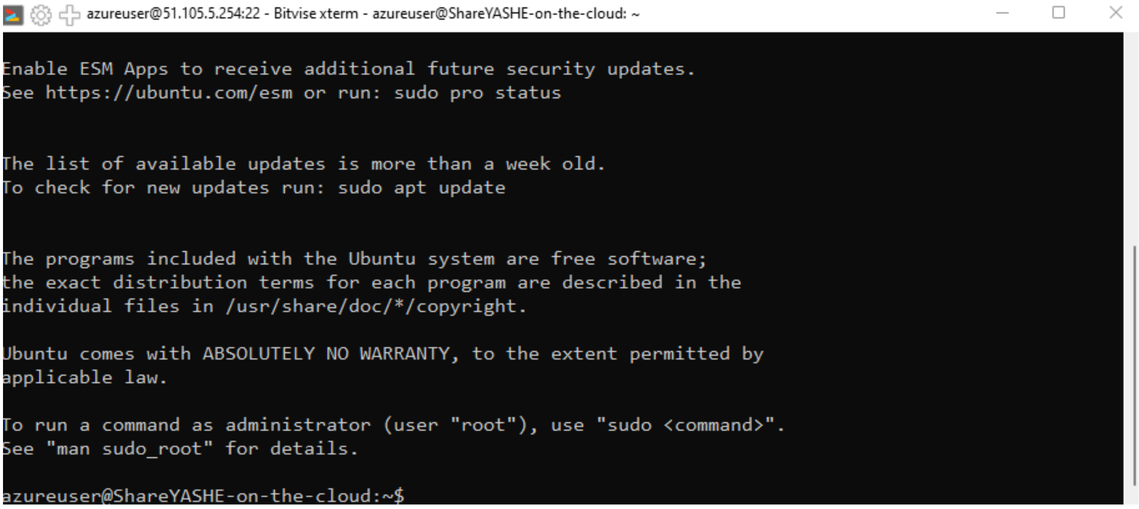
SHA-256 fingerprint: Ji9GbDBTR3r294KS2lfatwtCh2H4sOIFYELY7i6fpKo

MD5 fingerprint: fd:33:47:f5:60:ff:8b:23:83:89:9a:4f:46:34:6a:af

Bubble-babble: xupif-kirok-kifek-mifem-pudim-lakuk-zober-resuc-posic-zezyf-vexyx

Comment

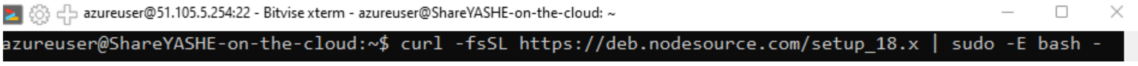
We can then click on “Log in”, and then in “Accept and save”. After this, we are able to open a new terminal window. We have then accessed to our cloud machine.


azureuser@51.105.5.254:22 - Bitvise xterm - azureuser@ShareYASHE-on-the-cloud: ~

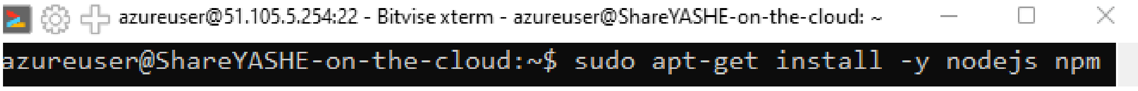
```
Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status


The list of available updates is more than a week old.
To check for new updates run: sudo apt update


The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

azureuser@ShareYASHE-on-the-cloud:~$
```

7.2.5 Install Node.js and NPM

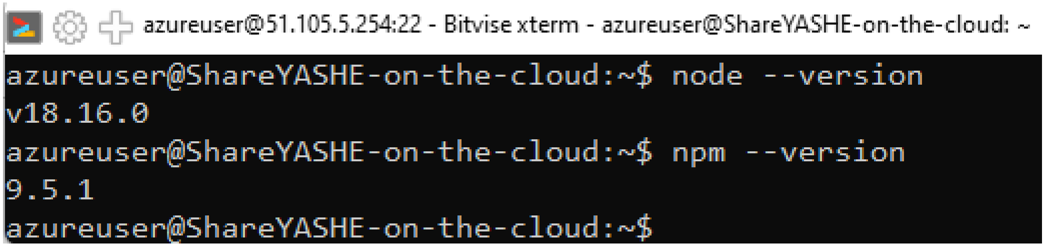
At the moment of this writing, Node.js v18 is not on the Ubuntu repository. Thus, we have to add it with the following command. Then, we will install Node.js and NPM. We can check the installation calling the commands with the "--version" option.


azureuser@51.105.5.254:22 - Bitvise xterm - azureuser@ShareYASHE-on-the-cloud: ~

```
azureuser@ShareYASHE-on-the-cloud:~$ curl -fsSL https://deb.nodesource.com/setup_18.x | sudo -E bash -
```


azureuser@51.105.5.254:22 - Bitvise xterm - azureuser@ShareYASHE-on-the-cloud: ~

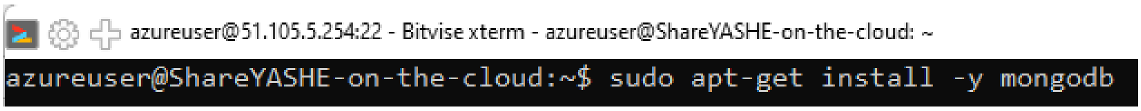
```
azureuser@ShareYASHE-on-the-cloud:~$ sudo apt-get install -y nodejs npm
```


azureuser@51.105.5.254:22 - Bitvise xterm - azureuser@ShareYASHE-on-the-cloud: ~

```
azureuser@ShareYASHE-on-the-cloud:~$ node --version
v18.16.0
azureuser@ShareYASHE-on-the-cloud:~$ npm --version
9.5.1
azureuser@ShareYASHE-on-the-cloud:~$
```

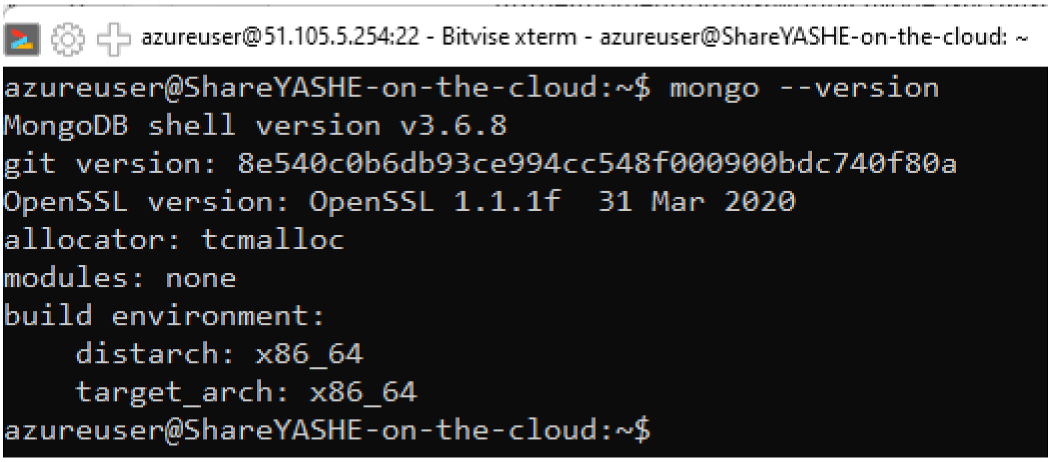
7.2.6 Install MongoDB and run it

For setting up the MongoDB database, we need to install its package with the following package.


azureuser@51.105.5.254:22 - Bitvise xterm - azureuser@ShareYASHE-on-the-cloud: ~

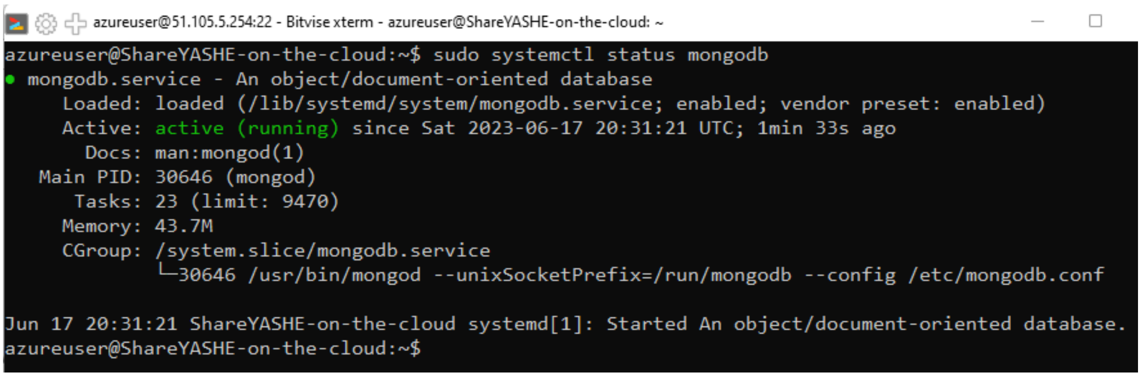
```
azureuser@ShareYASHE-on-the-cloud:~$ sudo apt-get install -y mongodb
```

We can then check its installation with the "--version" option.


azureuser@51.105.5.254:22 - Bitvise xterm - azureuser@ShareYASHE-on-the-cloud: ~

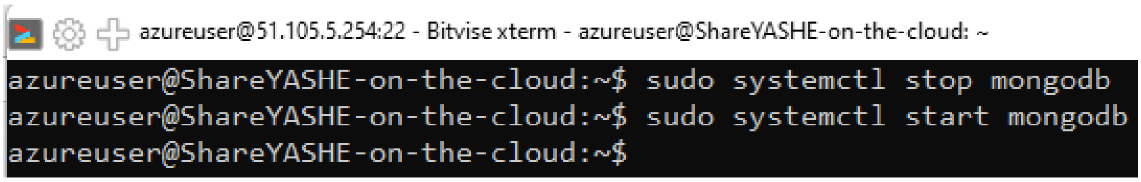
```
azureuser@ShareYASHE-on-the-cloud:~$ mongo --version
MongoDB shell version v3.6.8
git version: 8e540c0b6db93ce994cc548f000900bdc740f80a
OpenSSL version: OpenSSL 1.1.1f  31 Mar 2020
allocator: tcmalloc
modules: none
build environment:
    distarch: x86_64
    target_arch: x86_64
azureuser@ShareYASHE-on-the-cloud:~$
```

We must also check that the MongoDB service is running, with the following command.


azureuser@51.105.5.254:22 - Bitvise xterm - azureuser@ShareYASHE-on-the-cloud: ~

```
azureuser@ShareYASHE-on-the-cloud:~$ sudo systemctl status mongodb
● mongodb.service - An object/document-oriented database
     Loaded: loaded (/lib/systemd/system/mongodb.service; enabled; vendor preset: enabled)
     Active: active (running) since Sat 2023-06-17 20:31:21 UTC; 1min 33s ago
       Docs: man:mongod(1)
   Main PID: 30646 (mongod)
      Tasks: 23 (limit: 9470)
     Memory: 43.7M
     CGroup: /system.slice/mongodb.service
             └─30646 /usr/bin/mongod --unixSocketPrefix=/run/mongodb --config /etc/mongodb.conf

Jun 17 20:31:21 ShareYASHE-on-the-cloud systemd[1]: Started An object/document-oriented database.
azureuser@ShareYASHE-on-the-cloud:~$
```

If it is not running, we can restarting it with the following commands.

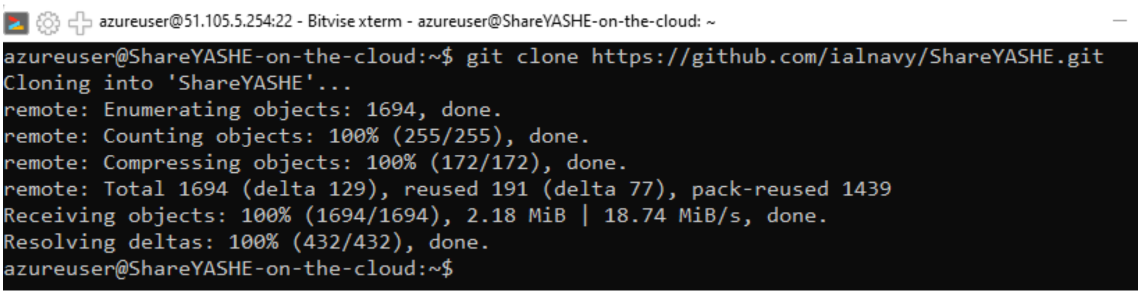

azureuser@51.105.5.254:22 - Bitvise xterm - azureuser@ShareYASHE-on-the-cloud: ~

```
azureuser@ShareYASHE-on-the-cloud:~$ sudo systemctl stop mongodb
azureuser@ShareYASHE-on-the-cloud:~$ sudo systemctl start mongodb
azureuser@ShareYASHE-on-the-cloud:~$
```

The MongoDB database would be running by now.

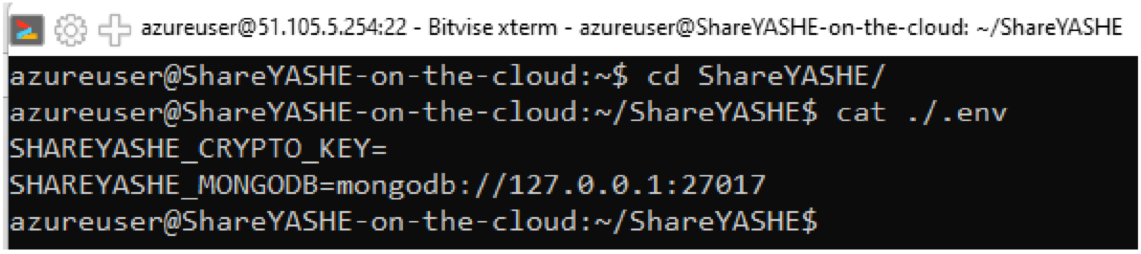
7.2.7 Obtain the ShareYASHE source code, customise its environment parameters, and run it

The code of ShareYASHE is published in a GitHub repository[Alv23] as an open-source project. We can download it by cloning it with the Git command.


azureuser@51.105.5.254:22 - Bitvise xterm - azureuser@ShareYASHE-on-the-cloud: ~

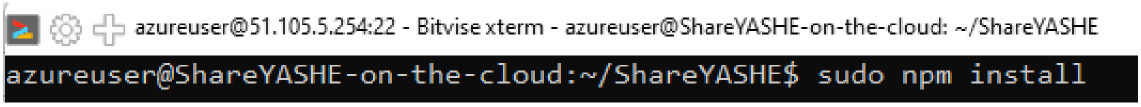
```
azureuser@ShareYASHE-on-the-cloud:~$ git clone https://github.com/ialnavy/ShareYASHE.git
Cloning into 'ShareYASHE'...
remote: Enumerating objects: 1694, done.
remote: Counting objects: 100% (255/255), done.
remote: Compressing objects: 100% (172/172), done.
remote: Total 1694 (delta 129), reused 191 (delta 77), pack-reused 1439
Receiving objects: 100% (1694/1694), 2.18 MiB | 18.74 MiB/s, done.
Resolving deltas: 100% (432/432), done.
azureuser@ShareYASHE-on-the-cloud:~$
```

In the ".env" file, in the root of the project, we need to specify two parameters: SHAREYASHE_CRYPTO_KEY (an entropy word, for passwords to be encrypted), and SHAREYASHE_MONGODB (the reference to the MongoDB database). We can do it with the command: "nano ./.env".

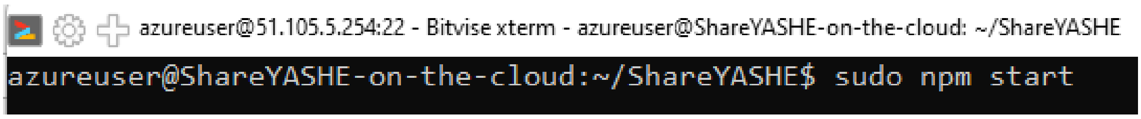

azureuser@51.105.5.254:22 - Bitvise xterm - azureuser@ShareYASHE-on-the-cloud: ~/ShareYASHE

```
azureuser@ShareYASHE-on-the-cloud:~$ cd ShareYASHE/
azureuser@ShareYASHE-on-the-cloud:~/ShareYASHE$ cat ./.env
SHAREYASHE_CRYPTO_KEY=
SHAREYASHE_MONGODB=mongodb://127.0.0.1:27017
azureuser@ShareYASHE-on-the-cloud:~/ShareYASHE$
```

After installing the NPM packages, we just call the script "npm start", whose steps are declared in the package.json file.


azureuser@51.105.5.254:22 - Bitvise xterm - azureuser@ShareYASHE-on-the-cloud: ~/ShareYASHE

```
azureuser@ShareYASHE-on-the-cloud:~/ShareYASHE$ sudo npm install
```


azureuser@51.105.5.254:22 - Bitvise xterm - azureuser@ShareYASHE-on-the-cloud: ~/ShareYASHE

```
azureuser@ShareYASHE-on-the-cloud:~/ShareYASHE$ sudo npm start
```

At this point, ShareYASHE would be running publicly.

We can then set up a DNS, like we did on premises, but we will not talk about this again here.

7.3 Developer’s guide

In this chapter, we will review specific questions that must be taken into consideration by any developer which encounters this project, for its fixing, or for its extension.

ShareYASHE possesses a robust and well-defined architecture that serves as the foundation for its functionality and future expansion. It follows a five-layer architecture that serves as the bedrock for its expandability and future development. This thoughtfully crafted architecture adheres to industry best practices and enables seamless integration of new features and modules.

Understanding and comprehending this intricate architecture empowers developers to extend ShareYASHE by effortlessly integrating new modules, refining existing functionalities, and guaranteeing the application's overall stability, scalability, and success. Thus, I strongly recommend reading the Software design and architecture section.

However, in this section I will be giving some overall steps for extending the ShareYASHE functionality.

7.3.1 Define a new route

The most common extension could involve adding a new route. In ShareYASHE, this process is carried out in the routes layer. The developer must consider the best routes file which will handle the new route, and add it as follows:

```
app.<HTTP_METHOD> ('/NEW_ROUTE', async function (req, res) {
    await appLayerFactory.for<NEW_ROUTE>Command(req, res);
});
```

7.3.2 Define a new application command

The command classes of the application layer implement every route that is defined in the routes layer. We must create a factory method for every new route.

```
    for<NEW_ROUTE>Command: async function (req, res) {
        let CommandExecutor = (await
import('./CommandExecutor.mjs')).CommandExecutor;
        let <NEW_ROUTE>Command = (await import('./
<NEW_ROUTE>Command.mjs')). <NEW_ROUTE>Command;

        let commandExecutor = new CommandExecutor(this.app,
this.mongoClient, ObjectId);
        let <NEW_ROUTE>Command = new <NEW_ROUTE>Command(this.app,
this.mongoClient, ObjectId);

        await commandExecutor.execute(req, res, <NEW_ROUTE>Command);
    },
```

Then, we must implement that command class.

```
import {BusinessFactory} from "../../businessLayer/BusinessFactory.mjs";
import {AbstractAppLayerCommand} from "../AbstractAppLayerCommand.mjs";

class <NEW_ROUTE>Command extends AbstractAppLayerCommand {
    async execute(req, res) {
        // Do the implementation
        // Call BusinessFactory for invoking high level actions
    }
}

export {<NEW_ROUTE>Command};
```

7.3.3 Define new business methods

The more we dig into the architecture, the less probable is that changes must be made there. It is possible that a new functionality only requires creating a new application command, restricting itself to only calling existing business methods. However, it is also possible that we may need to create new business methods.

I will give you an example of how to create a new method related to the authentication business logic. Furthermore, you may need to create a new business service class.

```
import {AbstractBusiness} from "../AbstractBusiness.mjs";
import {PersistenceFactory} from
"../../persistenceLayer/PersistenceFactory.mjs";

class AuthBusiness extends AbstractBusiness {
    /* New method */
    async howeverYouMayCallIt() {
        // Functionality
    }

    /* ... */

}
export {AuthBusiness};
```

Note that the method must conveniently be noted as asynchronous. This prevents confusions, because the core of the architecture, the persistence layer, requires calling methods from the database API asynchronously. If one method of those is called, the whole chain of responsibility must be asynchronous, waiting for methods down below with the “await” reserved word.

7.3.4 Instantiate new presentation service objects

If a new view is created, the first step is creating a Pug.js view. Then, we must declare it in the interface of the business factory.

```
class BusinessFactory {
    /* New view method */
    static forRender<name_of_the_new_view>(app, mongoClient, req, res) {
        return new RenderBusiness(app, mongoClient, req, res, "- <Title
of the new view>", "howeverYouCalledThePugFile.pug");
    }
    /* ... */
}
export {BusinessFactory};
```

The “render” method has the following signature:

```
render(message, username, ownedShExDocs, shExDoc)
```

If the user is logged in, you may call this render method from the application command passing the “username” and “ownedShExDocs” objects; you retrieve them from the service objects business layer, in the application layer.

Exceptionally, the view for editing a Shape Expressions document takes the “shExDoc” object, which is the Yjs CRDT type object. Yours may use it too.

7.3.5 Define a new entity

The least likely thing that you may need is having to define a new persistent entity. If that is the case, I have good news for you: the whole CRUD functionality is implemented in a superclass: AbstractRepository.

```
class <NEW_ENTITY>Repository extends AbstractRepository {
    constructor(app, mongoClient) {
        super(app, mongoClient, "<however_the_new_entity_is_called>");
    }
}

export {<NEW_ENTITY>Repository};
```

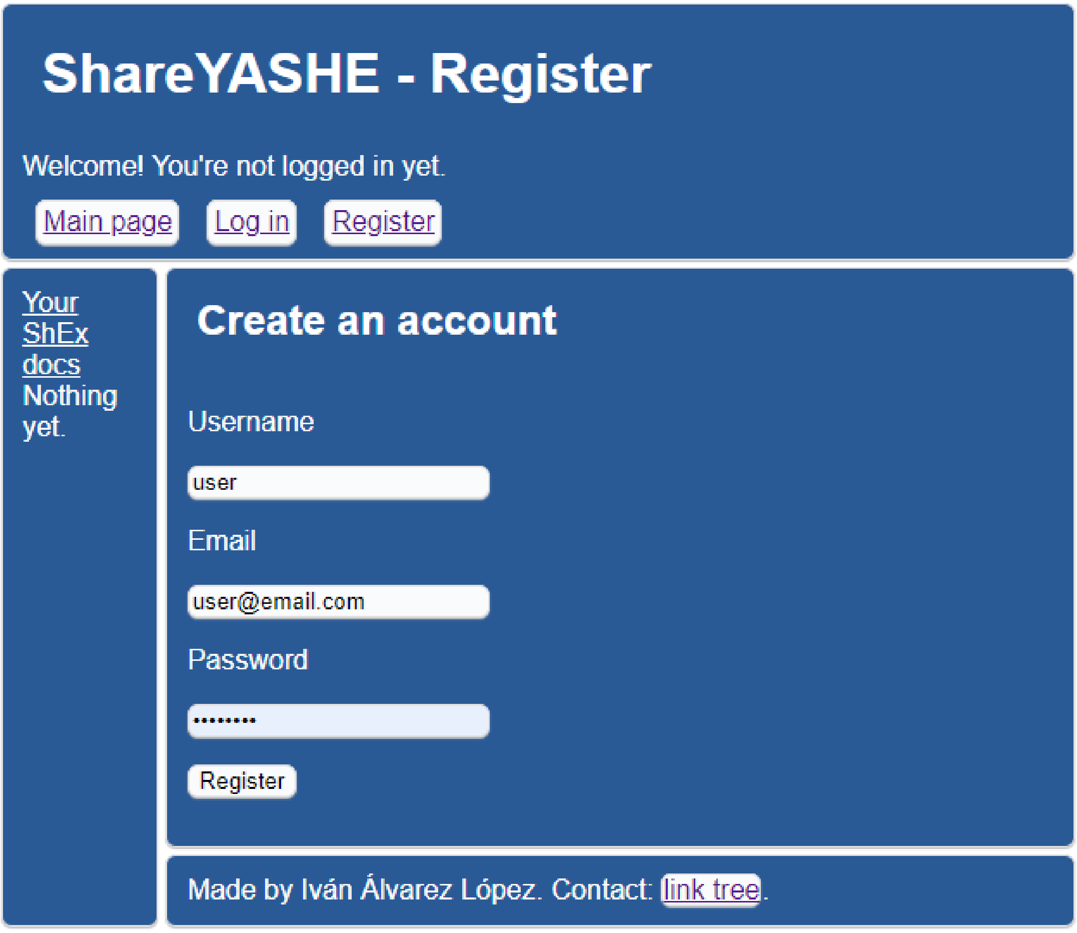
The last step is adding a new factory method in the persistence factory, and you would have a new entity added to ShareYASHE.

```
class PersistenceFactory {
    static for<NEW_ENTITY>(app, mongoClient) {
        return new <NEW_ENTITY>Repository(app, mongoClient);
    }
    /* ... */
}
export {PersistenceFactory};
```

7.4 User's manual

7.4.1 Create an account

If you want to create an account in ShareYASHE, you must click in the "Register" button and fill the formulary. You must specify a username and password, longer than three characters, which could not be already taken by another user. You also must write a correct email. Then, you click the button at the bottom of the formulary, and, if no message informs you on the contrary, you would be registered.



ShareYASHE - Register

Welcome! You're not logged in yet.

Nothing yet.

Create an account

Username: user

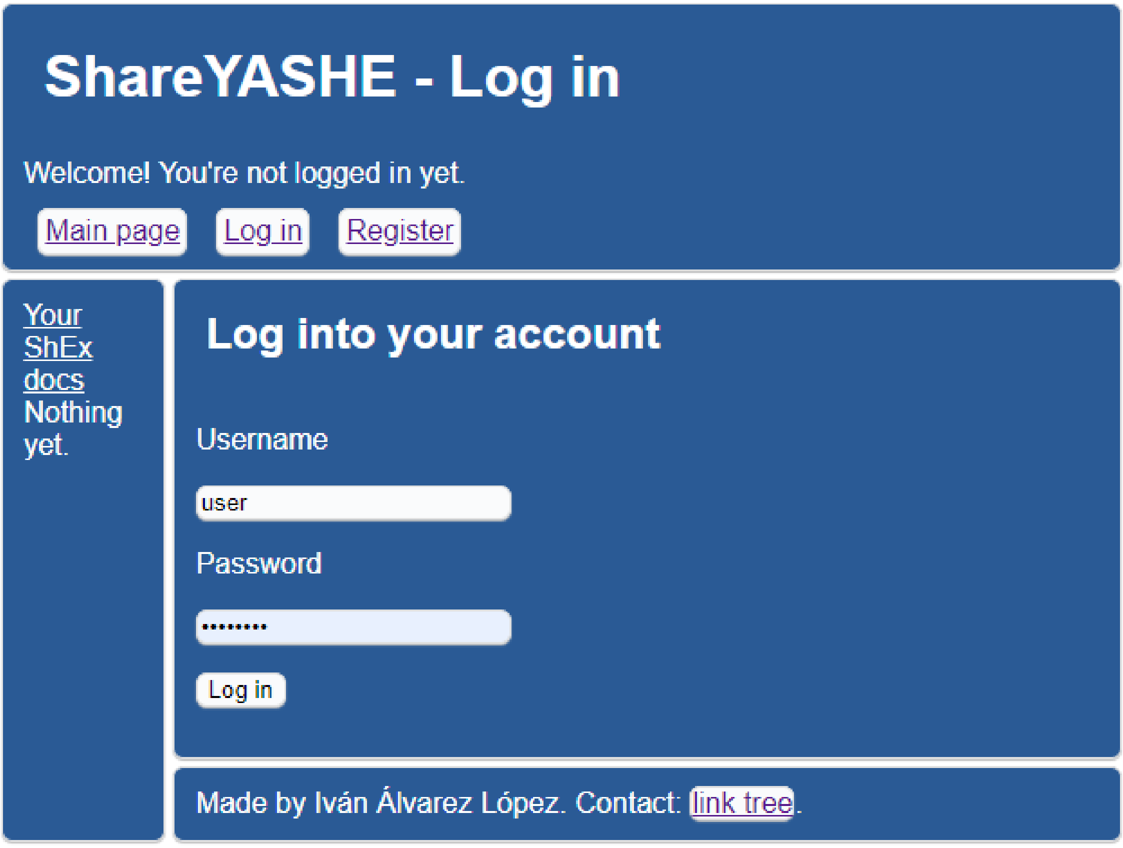
Email: user@email.com

Password: ••••••••

Made by Iván Álvarez López. Contact: link tree.

7.4.2 Login into your account

The login formulary needs you to write the information that you specified in the register form, the username and password must match with those that you used in the registration. Then, you click the button at the bottom of the formulary, and, if no message informs you on the contrary, you would be logged in.


ShareYASHE - Log in

Welcome! You're not logged in yet.

Your ShEx docs
Nothing yet.

Log into your account

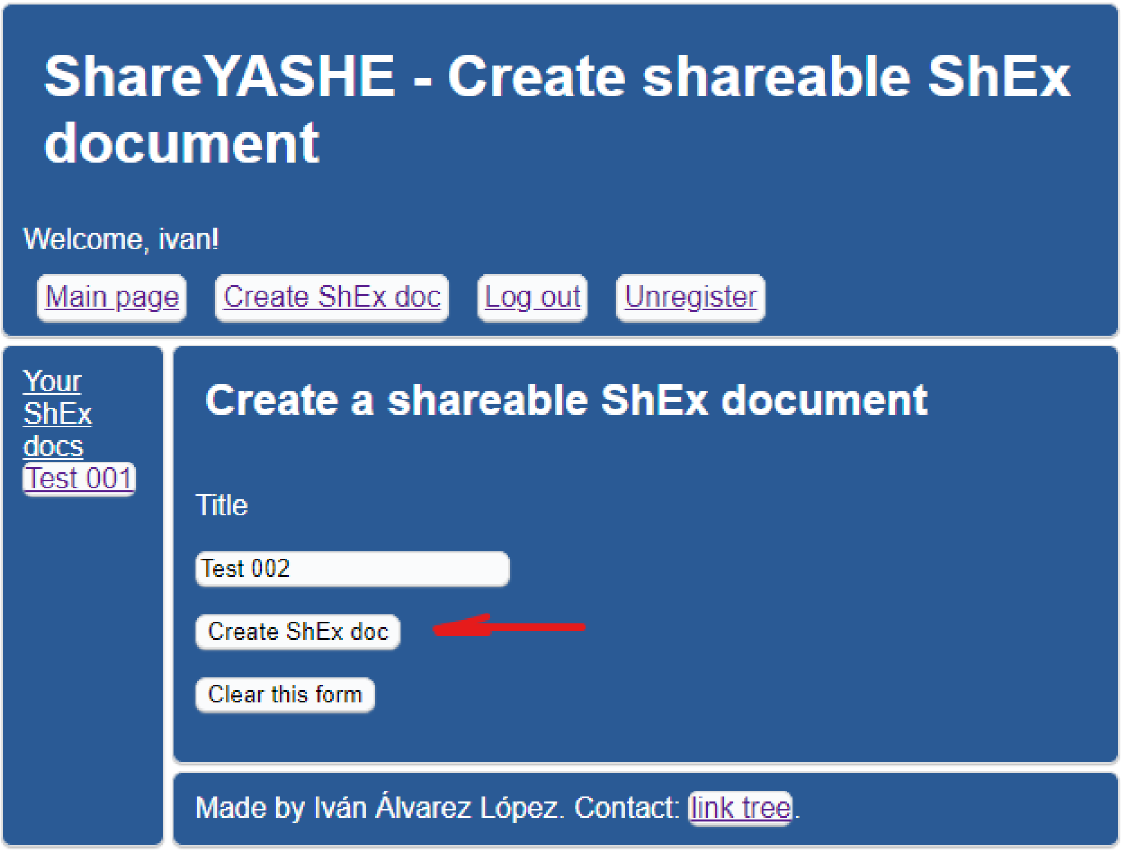
Username
user

Password
••••••••

Made by Iván Álvarez López. Contact: link tree.

7.4.3 Create a collaborative ShEx document

For this action, you must have made the login process. You must click on the “Create ShEx doc” option, write the title of the new ShEx document, and click the button of the formulary for its creation.



ShareYASHE - Create shareable ShEx document

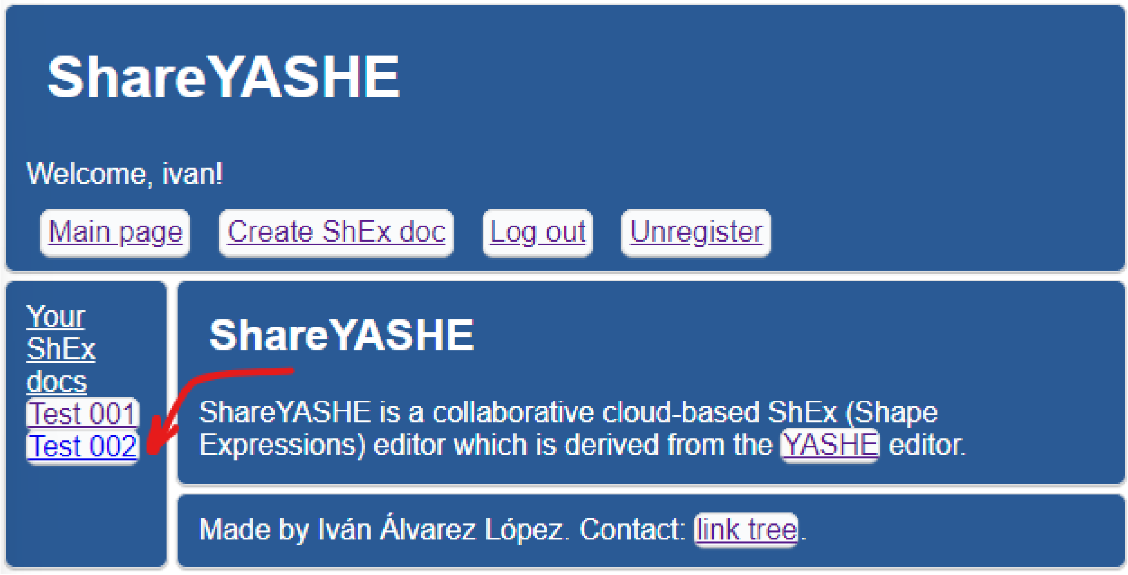
Welcome, ivan!

Create a shareable ShEx document

Title: Test 002

Made by Iván Álvarez López. Contact: link tree.

You will be redirected to the index page, and you will be able to see the new collaborative ShEx document on the left. Clicking it, you will access its edition view.



ShareYASHE

Welcome, ivan!

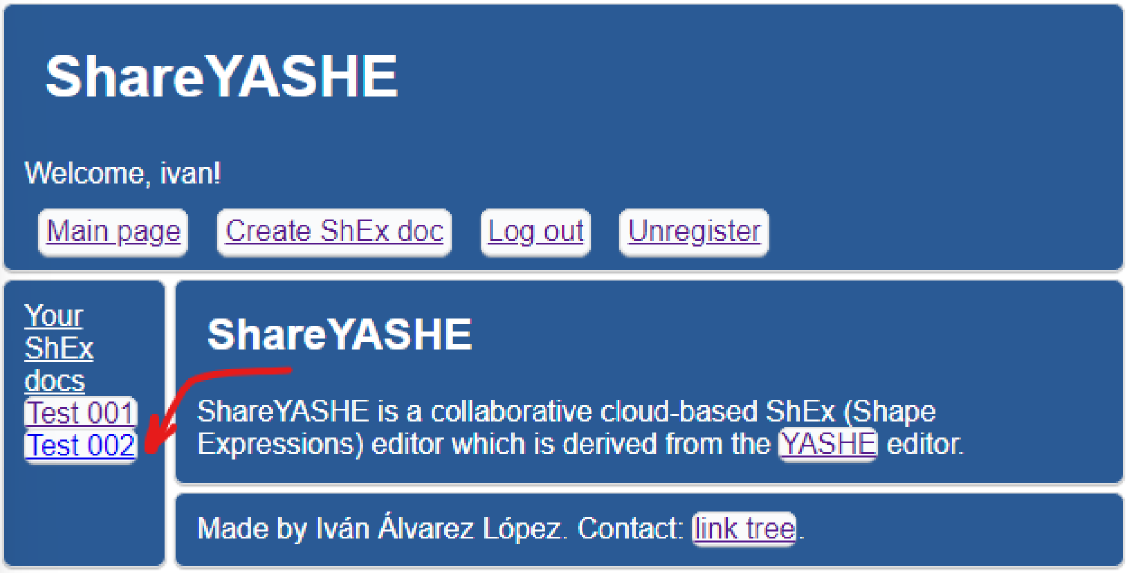
ShareYASHE

ShareYASHE is a collaborative cloud-based ShEx (Shape Expressions) editor which is derived from the YASHE editor.

Made by Iván Álvarez López. Contact: link tree.

7.4.4 Access your ShEx document

For this action, you must have made the login process, and created the collaborative ShEx document. You will always see your ShEx documents on the left of the Website, and access clicking on them.



ShareYASHE

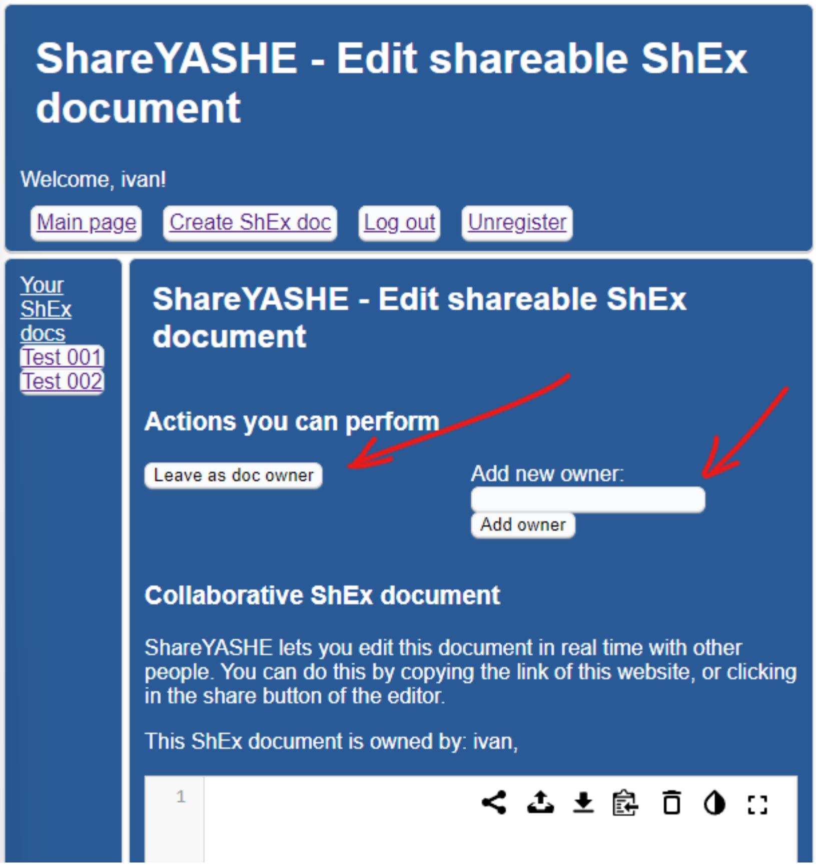
Welcome, ivan!

ShareYASHE

ShareYASHE is a collaborative cloud-based ShEx (Shape Expressions) editor which is derived from the YASHE editor.

Made by Iván Álvarez López. Contact: link tree.

Your view as an owner will provide you some actions, that a non-owner will not see.



ShareYASHE - Edit shareable ShEx document

Welcome, ivan!

ShareYASHE - Edit shareable ShEx document

Actions you can perform

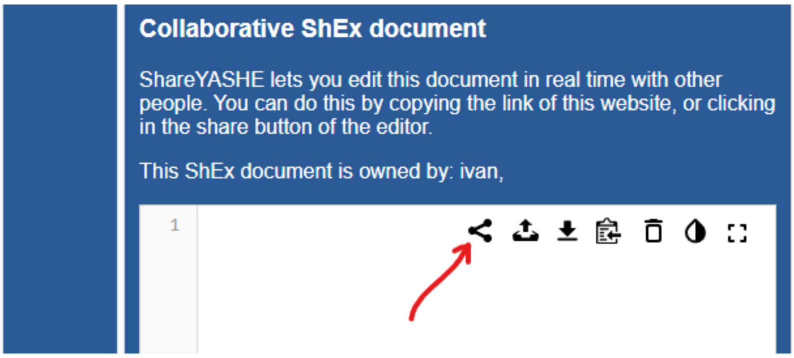
Collaborative ShEx document

ShareYASHE lets you edit this document in real time with other people. You can do this by copying the link of this website, or clicking in the share button of the editor.

This ShEx document is owned by: ivan,

7.4.5 Share your ShEx document

For this action, you must have made the login process, be in possession of any collaborative ShEx document, and accessed it. Sharing your ShEx document means inviting a person, who we will name as the guest, to edit the file, but not conserving it in his list at the left. By either copying the URL link, or clicking the sharing button, anybody can access the edition of your document.

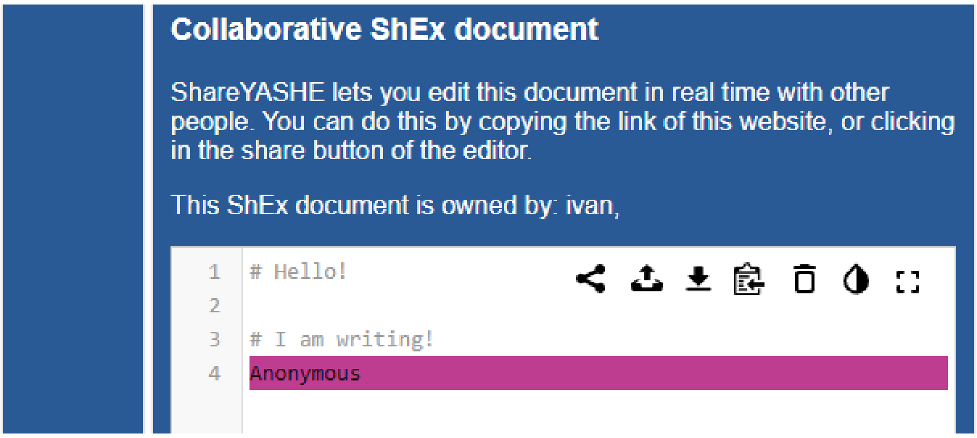


Collaborative ShEx document

ShareYASHE lets you edit this document in real time with other people. You can do this by copying the link of this website, or clicking in the share button of the editor.

This ShEx document is owned by: ivan,

The guest does not need to have made the login process, and he will be able to see and edit the ShEx document in real time. He will appear to the owner as "Anonymous", if he is not logged in, otherwise it will prompt his username.



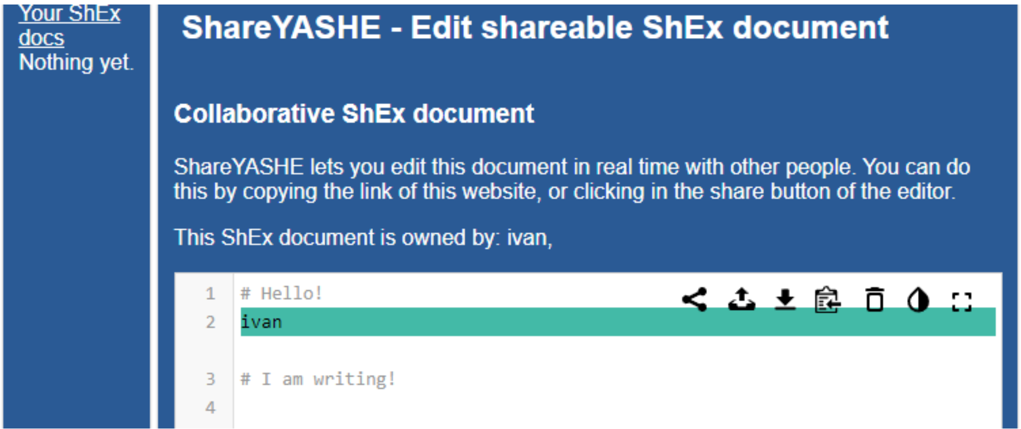
Collaborative ShEx document

ShareYASHE lets you edit this document in real time with other people. You can do this by copying the link of this website, or clicking in the share button of the editor.

This ShEx document is owned by: ivan,

```
# Hello!

# I am writing!
```

The ShEx document will not be saved in the list of the left side for the guest, and he will see your username, as follows:



Your ShEx docs
Nothing yet.

ShareYASHE - Edit shareable ShEx document

Collaborative ShEx document

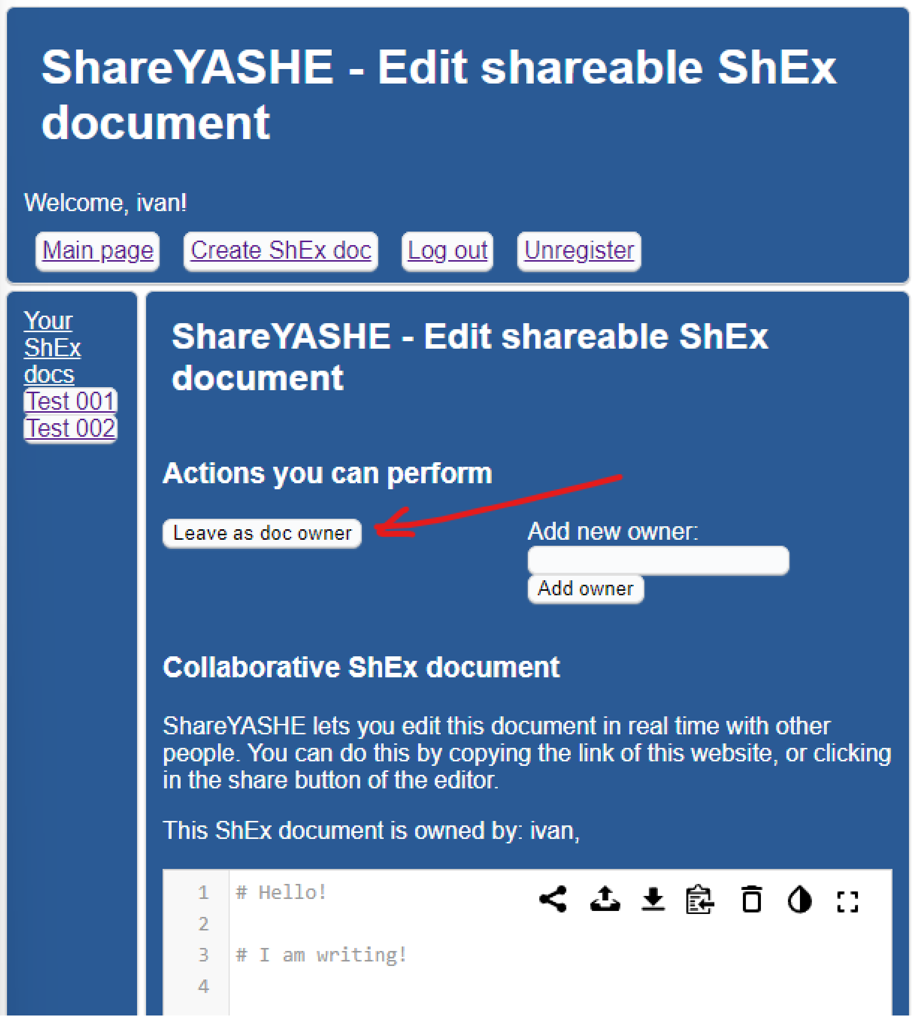
ShareYASHE lets you edit this document in real time with other people. You can do this by copying the link of this website, or clicking in the share button of the editor.

This ShEx document is owned by: ivan,

```
# Hello!

# I am writing!
```

7.4.6 Leave the ownership of a ShEx document

For this action, you must have made the login process, and be in possession of any collaborative ShEx document. All you have to do is clicking the button “Leve as doc owner”.



ShareYASHE - Edit shareable ShEx document

Welcome, ivan!

ShareYASHE - Edit shareable ShEx document

Actions you can perform

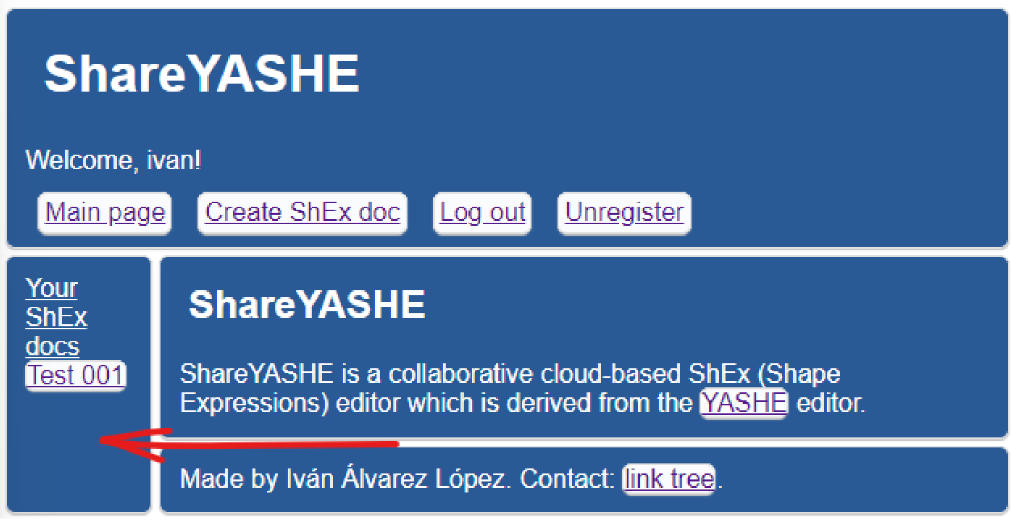
Collaborative ShEx document

ShareYASHE lets you edit this document in real time with other people. You can do this by copying the link of this website, or clicking in the share button of the editor.

This ShEx document is owned by: ivan,

```
# Hello!

# I am writing!

```

This will remove it from your list and, if nobody else owns it, it will be deleted.



ShareYASHE

Welcome, ivan!

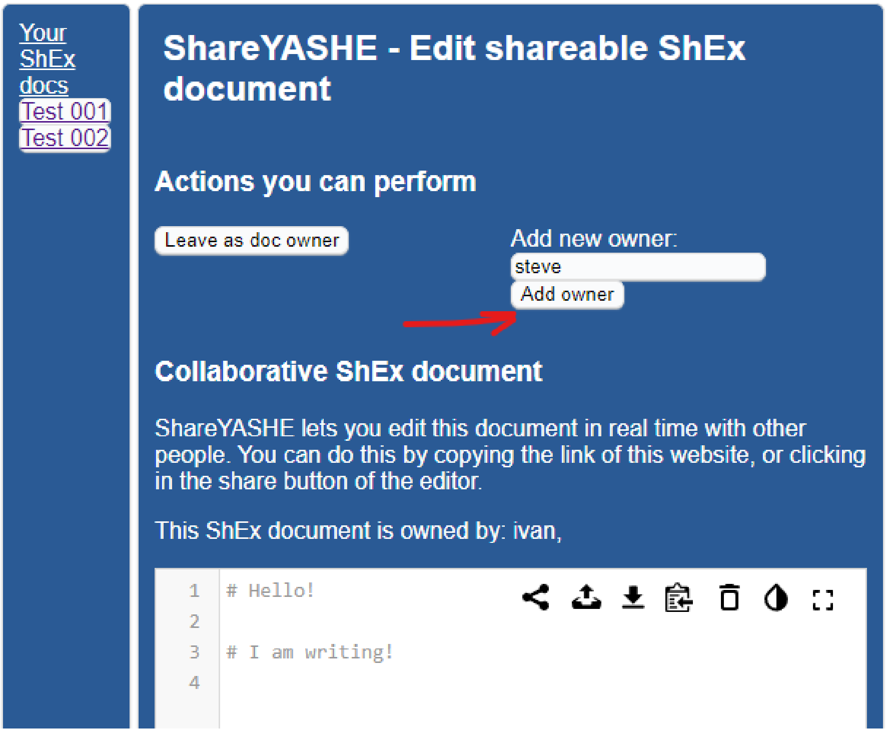
ShareYASHE

ShareYASHE is a collaborative cloud-based ShEx (Shape Expressions) editor which is derived from the YASHE editor.

Made by Iván Álvarez López. Contact: link tree.

7.4.7 Add an owner to a ShEx document

For this action, you must have made the login process, and be in possession of any collaborative ShEx document. For adding a new owner, we must specify an existing username in the formulary marked below.



ShareYASHE - Edit shareable ShEx document

Actions you can perform

Add new owner: steve

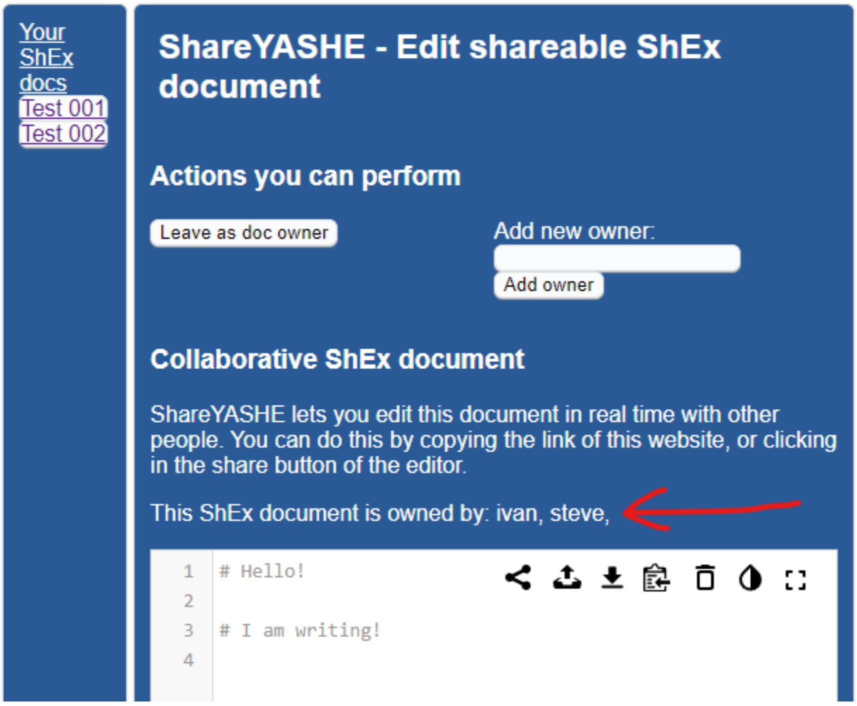
Collaborative ShEx document

ShareYASHE lets you edit this document in real time with other people. You can do this by copying the link of this website, or clicking in the share button of the editor.

This ShEx document is owned by: ivan,

```
# Hello!

# I am writing!

```

If the user exists, it will appear in the owners list.



ShareYASHE - Edit shareable ShEx document

Actions you can perform

Add new owner:

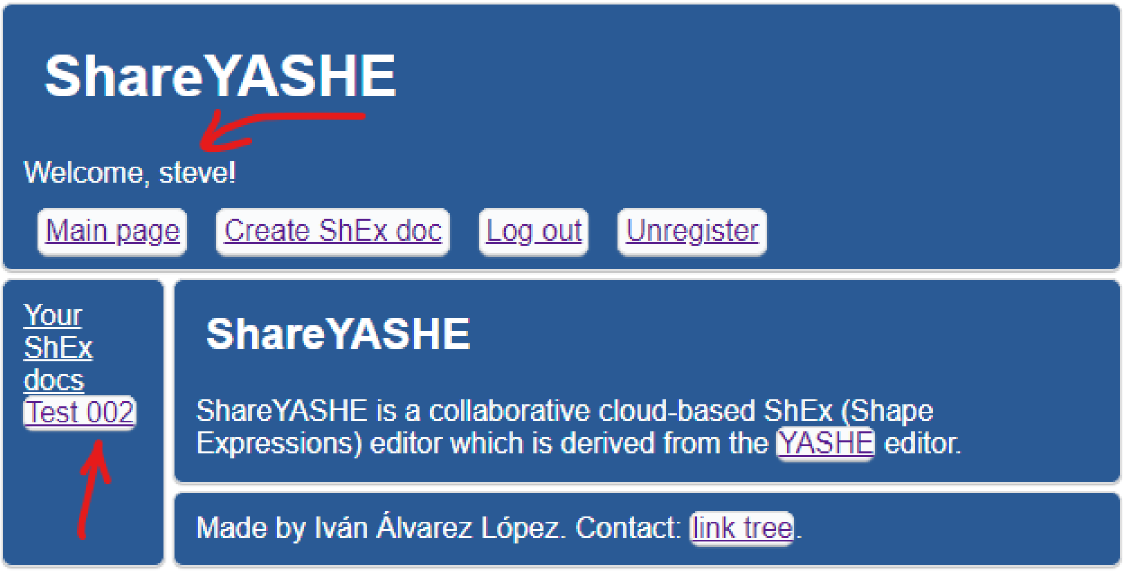
Collaborative ShEx document

ShareYASHE lets you edit this document in real time with other people. You can do this by copying the link of this website, or clicking in the share button of the editor.

This ShEx document is owned by: ivan, steve,

```
# Hello!

# I am writing!

```

The added owner will be able to see the ShEx document in its lists of the left. One ShEx document can have an indefinite number of owners.



ShareYASHE

Welcome, steve!

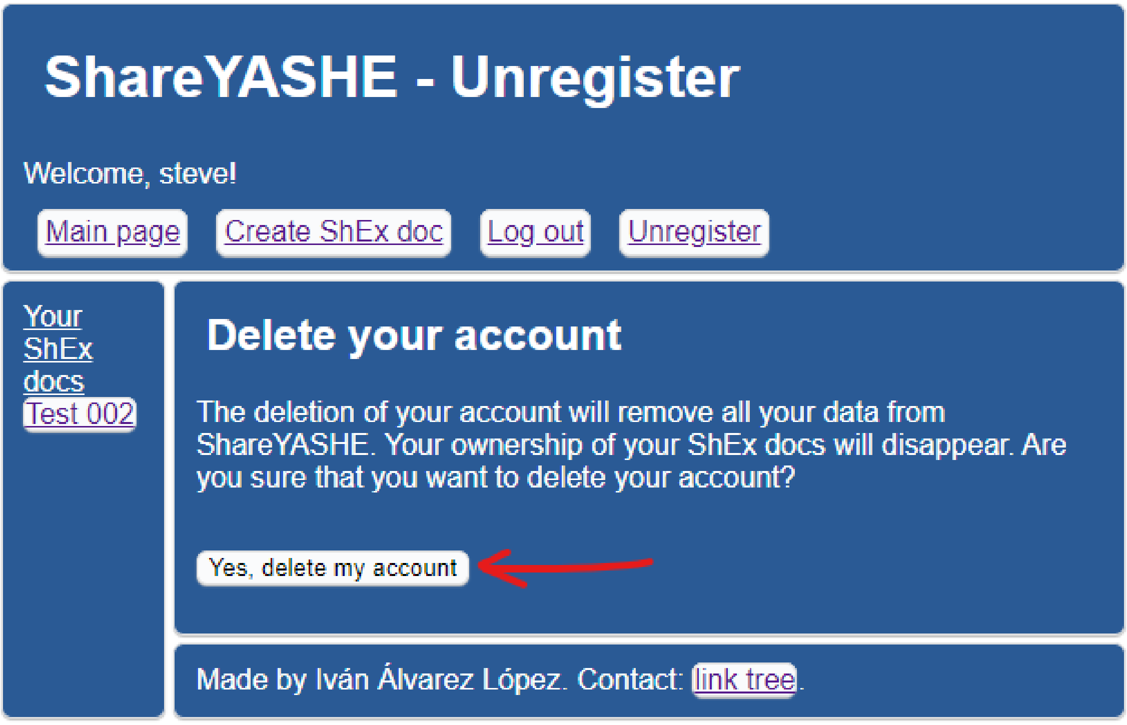
ShareYASHE

ShareYASHE is a collaborative cloud-based ShEx (Shape Expressions) editor which is derived from the YASHE editor.

Made by Iván Álvarez López. Contact: link tree.

7.4.8 Delete your account

For this action, you must have made the login process. Clicking the “Unregister” button, you will be redirected to a confirmation view. This is a terminative action which will imply the deletion of all your user information, as well as the removal of your ownership of any ShEx document, and its erase if it has no more owners left.



ShareYASHE - Unregister

Welcome, steve!

Delete your account

The deletion of your account will remove all your data from ShareYASHE. Your ownership of your ShEx docs will disappear. Are you sure that you want to delete your account?

Made by Iván Álvarez López. Contact: link tree.

8 Concluding remarks

8.1 Conclusions

The result of this project is a remarkable Software application which has been designed for having a flexible and scalable architecture. Despite of the moderate size of the project, the conception of the product, as well as this documentation, evidences my accomplishment of becoming a Software engineer.

In the beginning, I meant to draft a very specific piece of Software as a degree project, which I could emphasise Software architecture with. I am currently interested in fields such as SDV (Software Defined Vehicles), or distributed systems in general. However, I realised that this could have been an ambitious job that I would better leave for my future.

Eventually, my expectations were thoroughly met with ShareYASHE. Carrying out this project, I intended to demonstrate that any good Software development practise, especially object-oriented ones, could be achieved, independent of which programming language or which tool is used. I made the most of standards and recommendations, which were the key to peak this goal.

Not easily have I come to this point of my career. These have been very demanding years, given the pandemic, and other familiar issues that I will not bother you with. Being the beginning of my professional career, this project is not the last milestone of my student life. Software professionals must be constantly learning, and that is what I will do. ShareYASHE is my first creation and will not be the last. Indeed, I will hold all the knowledge that I was provided by this degree, and I will strive to become an excellent Software engineer.

8.2 Contemplated upgrades

Further functionality could be implemented in this project. ShareYASHE has a straightforward purpose, giving support for ShEx editing, so few upgrades may be contemplated. However, they could be added to the application, if necessary, in the future. These following ideas were not taken into consideration because they exceeded the scope of the project for this period, or they were thought after the development of ShareYASHE.

- Automatically send a message to verify user’s email at his registration.
- Creating invitation requests rather than suddenly adding another user as owner of a ShEx document.
- Dynamically represent Shape Expressions using graphs, while ShEx code is written.
- Support for other programming languages besides ShEx.

9 Bibliography

9.1 Books and articles

[PMI21] Project Management Institute. “A guide to the Project Management Body of Knowledge (PMBOK guide) and the Standard for project management”. 2021. ISBN 10: 1628256648. ISBN 13: 978-1628256642.

[IanS11] Sommerville, Ian. “Software Engineering (9th edition)”. Pearson. 2011. ISBN 10: 0-13-703515-2. ISBN 13: 978-0-13-703515-1.

[GoF94] Gamma, Erich; Helm, Richard; Johnson, Ralph; Vlissides, John; Grady, Booch. “Design Patterns: Elements of Reusable Object-Oriented Software”. Addison-Wesley Professional. 1999. ISBN 13: 978-0201633610.

[Fowler02] Fowler, Martin; Rice, Dave; Foemmel, Matthew; Hieatt, Edward; Mee, Robert; Stafford, Randy. “Patterns of Enterprise Application Architecture”. Prentice Hall. 2009. ISBN-10: 0321127420. ISBN-13: 978-0321127426.

[Krug13] Krug, Steve. “Don't Make Me Think, Revisited: A Common Sense Approach to Web Usability (3rd Edition)”. New Riders. 2013. ISBN-10: 9780321965516. ISBN-13: 978-0321965516.

9.2 References on the Internet

[ECMA15] ECMA International. “ECMAScript® 2015 Language Specification”. https://262.ecma-international.org/6.0/. Last seen: 15th June 2023.

[Wiki12] “Document-oriented database”. https://en.wikipedia.org/wiki/Document-oriented_database. Last seen: 16th June 2023.

[WCAG08] “Web Content Accessibility Guidelines (WCAG) 2.0”. https://www.w3.org/TR/WCAG20/. Last seen: 16th June 2023.

[Jahns14] “The Yjs Project”. https://yjs.dev/. Last seen: 17th June 2023.

[Alv23] “ShareYASHE (Git repo)”. https://github.com/ialnavy/ShareYASHE. Last seen: 17th June 2023.

[GanPr20] “What is GanttProject Cloud?”. https://help.ganttproject.biz/t/what-is-ganttproject-cloud/1955. Last seen: 17th June 2023.

[Node23] “Node.js Website”. https://nodejs.org/es. Last seen: 17th June 2023.

[Mongo23] "MongoDB Community Edition". https://www.mongodb.com/try/download/community-kubernetes-operator. Last seen: 17th June 2023.

[WIMIPA23] "What Is My IP Address?". https://whatismyipaddress.com/. Last seen: 17th June 2023.

[Azure23] "Microsoft Azure Portal". https://portal.azure.com/#home. Last seen: 17th June 2023.

[Bitvise23] "Bitvise SSH Client". https://www.bitvise.com/. Last seen: 17th June 2023.

[ShEx13] "Shape Expressions Definition". https://www.w3.org/2013/ShEx/Definition. Last seen: 18th June 2023.

[W3CHTML23] "W3C HTML Markup Validation Service". https://validator.w3.org/. Last seen: 21st June 2023.

[W3CCSS23] "W3C CSS Validation Service". https://jigsaw.w3.org/css-validator/. Last seen: 21st June 2023.

[GAdapCheck23] "Google Mobile Optimization Checker". https://search.google.com/test/mobile-friendly. Last seen: 21st June 2023.

[WAVE23] "Web Accessibility Evaluation Tool". https://wave.webaim.org/. Last seen: 21st June 2023.

[ACH23] "aChecker". https://achecks.ca/achecker. Last seen: 21st June 2023.

[TAW23] "Test of Web Accessibility". https://www.tawdis.net/index. Last seen: 21st June 2023.

[WAIMCC23] "WebAIM Contrast Checker". https://webaim.org/resources/contrastchecker/. Last seen: 21st June 2023.

Appendix I. Contents attached to this submission

Along with this document, it will be submitted a .zip file which contains the source code of ShareYASHE. The content of this .zip file is listed below.

Location	Content
ShareYASHE.gan	Gantt chart containing a scheme of the initial project planning.
README.md	This is a read-me file, used as a Copyright disclaimer.
package.json	This file is used by Node.js for knowing which libraries to install.
package-lock.json	Idem "package.json".
.env	This file contains the definition of the required environment variables.
app.js	This file contains the root source code of the ShareYASHE server application.
bin/www.js	This file is a script which deploys the main ShareYASHE server. The "npm start" command will begin executing this script.
wsServer/	This folder contains some *y-websocket* utilities, which were taken from the Yjs GitHub page, and modified a little bit.
diagrams/	This folder contains the diagrams shown in this document, both in format .svg and .plantuml.
routes/	This folder contains the source code of the routes layer.
applicationLayer/	This folder contains the source code of the application layer.
businessLayer/	This folder contains the source code of the business layer, and the presentation layer.
persistenceLayer/	This folder contains the source code of the persistence layer.

Figure 123. List of contents attached to this submission.

Note that ShareYASHE needs more assets in order to be ran: the Node.js engine, a deployed MongoDB database, and the "node-modules" installed (see any guide of deployment).

Appendix II. GNU Affero General Public License v3.0

GNU GENERAL PUBLIC LICENSE

Version 3, 29 June 2007

Copyright © 2007 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.

Preamble

The GNU General Public License is a free, copyleft license for software and other kinds of works.

The licenses for most software and other practical works are designed to take away your freedom to share and change the works. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change all versions of a program--to make sure it remains free software for all its users. We, the Free Software Foundation, use the GNU General Public License for most of our software; it applies also to any other work released this way by its authors. You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for them if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs, and that you know you can do these things.

To protect your rights, we need to prevent others from denying you these rights or asking you to surrender the rights. Therefore, you have certain responsibilities if you distribute copies of the software, or if you modify it: responsibilities to respect the freedom of others.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must pass on to the recipients the same freedoms that you received. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

Developers that use the GNU GPL protect your rights with two steps: (1) assert copyright on the software, and (2) offer you this License giving you legal permission to copy, distribute and/or modify it.

For the developers' and authors' protection, the GPL clearly explains that there is no warranty for this free software. For both users' and authors' sake, the GPL requires that modified versions be marked as changed, so that their problems will not be attributed erroneously to authors of previous versions.

Some devices are designed to deny users access to install or run modified versions of the software inside them, although the manufacturer can do so. This is fundamentally incompatible with the aim of protecting users' freedom to change the software. The systematic pattern of such abuse occurs in the area of products for individuals to use, which is precisely where it is most unacceptable. Therefore, we have designed this version of the GPL to prohibit the practice for those products. If such problems arise substantially in other domains, we stand ready to extend this provision to those domains in future versions of the GPL, as needed to protect the freedom of users.

Finally, every program is threatened constantly by software patents. States should not allow patents to restrict development and use of software on general-purpose computers, but in those that do, we wish to avoid the special danger that patents applied to a free program could make it effectively proprietary. To prevent this, the GPL assures that patents cannot be used to render the program non-free.

The precise terms and conditions for copying, distribution and modification follow.

Terms and conditions

Definitions

“This License” refers to version 3 of the GNU General Public License.

“Copyright” also means copyright-like laws that apply to other kinds of works, such as semiconductor masks.

“The Program” refers to any copyrightable work licensed under this License. Each licensee is addressed as “you”. “Licensees” and “recipients” may be individuals or organizations.

To “modify” a work means to copy from or adapt all or part of the work in a fashion requiring copyright permission, other than the making of an exact copy. The resulting work is called a “modified version” of the earlier work or a work “based on” the earlier work.

A “covered work” means either the unmodified Program or a work based on the Program.

To “propagate” a work means to do anything with it that, without permission, would make you directly or secondarily liable for infringement under applicable copyright law, except executing it on a computer or modifying a private copy. Propagation includes copying, distribution (with or without modification), making available to the public, and in some countries other activities as well.

To “convey” a work means any kind of propagation that enables other parties to make or receive copies. Mere interaction with a user through a computer network, with no transfer of a copy, is not conveying.

An interactive user interface displays “Appropriate Legal Notices” to the extent that it includes a convenient and prominently visible feature that (1) displays an appropriate copyright notice, and (2) tells the user that there is no warranty for the work (except to the extent that warranties are provided), that licensees may convey the work under this License, and how to view a copy of this License. If the interface presents a list of user commands or options, such as a menu, a prominent item in the list meets this criterion.

Source Code

The “source code” for a work means the preferred form of the work for making modifications to it. “Object code” means any non-source form of a work.

A “Standard Interface” means an interface that either is an official standard defined by a recognized standards body, or, in the case of interfaces specified for a particular programming language, one that is widely used among developers working in that language.

The “System Libraries” of an executable work include anything, other than the work as a whole, that (a) is included in the normal form of packaging a Major Component, but which is not part of that Major Component, and (b) serves only to enable use of the work with that Major Component, or to implement a Standard Interface for which an implementation is available to the public in source code form. A “Major Component”, in this context, means a major essential component (kernel, window system, and so on) of the specific operating system (if any) on which the executable work runs, or a compiler used to produce the work, or an object code interpreter used to run it.

The “Corresponding Source” for a work in object code form means all the source code needed to generate, install, and (for an executable work) run the object code and to modify the work, including scripts to control those activities. However, it does not include the work's System Libraries, or general-purpose tools or generally available free programs which are used unmodified in performing those activities but which are not part of the work. For example, Corresponding Source includes interface definition files associated with source files for the work, and the source code for shared libraries and dynamically linked subprograms that the work is specifically designed to require, such as by intimate data communication or control flow between those subprograms and other parts of the work.

The Corresponding Source need not include anything that users can regenerate automatically from other parts of the Corresponding Source.

The Corresponding Source for a work in source code form is that same work.

Basic Permissions

All rights granted under this License are granted for the term of copyright on the Program, and are irrevocable provided the stated conditions are met. This License explicitly affirms your unlimited permission to run the unmodified Program. The output from running a covered work is covered by this License only if the output, given its content, constitutes a covered work. This License acknowledges your rights of fair use or other equivalent, as provided by copyright law.

You may make, run and propagate covered works that you do not convey, without conditions so long as your license otherwise remains in force. You may convey covered works to others for the sole purpose of having them make modifications exclusively for you, or provide you with facilities for running those works, provided that you comply with the terms of this License in conveying all material for which you do not control copyright. Those thus making or running the covered works for you must do so exclusively on your behalf, under your direction and control, on terms that prohibit them from making any copies of your copyrighted material outside their relationship with you.

Conveying under any other circumstances is permitted solely under the conditions stated below. Sublicensing is not allowed; section 10 makes it unnecessary.

Protecting Users' Legal Rights From Anti-Circumvention Law

No covered work shall be deemed part of an effective technological measure under any applicable law fulfilling obligations under article 11 of the WIPO copyright treaty adopted on 20 December 1996, or similar laws prohibiting or restricting circumvention of such measures.

When you convey a covered work, you waive any legal power to forbid circumvention of technological measures to the extent such circumvention is effected by exercising rights under this License with respect to the covered work, and you disclaim any intention to limit operation or modification of the work as a means of enforcing, against the work's users, your or third parties' legal rights to forbid circumvention of technological measures.

Conveying Verbatim Copies

You may convey verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice; keep intact all notices stating that this License and any non-permissive terms added in accord with section 7 apply to the code; keep intact all notices of the absence of any warranty; and give all recipients a copy of this License along with the Program.

You may charge any price or no price for each copy that you convey, and you may offer support or warranty protection for a fee.

Conveying Modified Source Versions

You may convey a work based on the Program, or the modifications to produce it from the Program, in the form of source code under the terms of section 4, provided that you also meet all of these conditions:

a. The work must carry prominent notices stating that you modified it, and giving a relevant date.
b. The work must carry prominent notices stating that it is released under this License and any conditions added under section 7. This requirement modifies the requirement in section 4 to “keep intact all notices”.
c. You must license the entire work, as a whole, under this License to anyone who comes into possession of a copy. This License will therefore apply, along with any applicable section 7 additional terms, to the whole of the work, and all its parts, regardless of how they are packaged. This License gives no permission to license the work in any other way, but it does not invalidate such permission if you have separately received it.
d. If the work has interactive user interfaces, each must display Appropriate Legal Notices; however, if the Program has interactive interfaces that do not display Appropriate Legal Notices, your work need not make them do so.

A compilation of a covered work with other separate and independent works, which are not by their nature extensions of the covered work, and which are not combined with it such as to form a larger program, in or on a volume of a storage or distribution medium, is called an “aggregate” if the compilation and its resulting copyright are not used to limit the access or legal rights of the compilation's users beyond what the individual works permit. Inclusion of a covered work in an aggregate does not cause this License to apply to the other parts of the aggregate.

Conveying Non-Source Forms

You may convey a covered work in object code form under the terms of sections 4 and 5, provided that you also convey the machine-readable Corresponding Source under the terms of this License, in one of these ways:

a. Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by the Corresponding Source fixed on a durable physical medium customarily used for software interchange.

b. Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by a written offer, valid for at least three years and valid for as long as you offer spare parts or customer support for that product model, to give anyone who possesses the object code either (1) a copy of the Corresponding Source for all the software in the product that is covered by this License, on a durable physical medium customarily used for software interchange, for a price no more than your reasonable cost of physically performing this conveying of source, or (2) access to copy the Corresponding Source from a network server at no charge.
c. Convey individual copies of the object code with a copy of the written offer to provide the Corresponding Source. This alternative is allowed only occasionally and noncommercially, and only if you received the object code with such an offer, in accord with subsection 6b.
d. Convey the object code by offering access from a designated place (gratis or for a charge), and offer equivalent access to the Corresponding Source in the same way through the same place at no further charge. You need not require recipients to copy the Corresponding Source along with the object code. If the place to copy the object code is a network server, the Corresponding Source may be on a different server (operated by you or a third party) that supports equivalent copying facilities, provided you maintain clear directions next to the object code saying where to find the Corresponding Source. Regardless of what server hosts the Corresponding Source, you remain obligated to ensure that it is available for as long as needed to satisfy these requirements.
e. Convey the object code using peer-to-peer transmission, provided you inform other peers where the object code and Corresponding Source of the work are being offered to the general public at no charge under subsection 6d.

A separable portion of the object code, whose source code is excluded from the Corresponding Source as a System Library, need not be included in conveying the object code work.

A “User Product” is either (1) a “consumer product”, which means any tangible personal property which is normally used for personal, family, or household purposes, or (2) anything designed or sold for incorporation into a dwelling. In determining whether a product is a consumer product, doubtful cases shall be resolved in favor of coverage. For a particular product received by a particular user, “normally used” refers to a typical or common use of that class of product, regardless of the status of the particular user or of the way in which the particular user actually uses, or expects or is expected to use, the product. A product is a consumer product regardless of whether the product has substantial commercial, industrial or non-consumer uses, unless such uses represent the only significant mode of use of the product.

“Installation Information” for a User Product means any methods, procedures, authorization keys, or other information required to install and execute modified versions of a covered work in that User Product from a modified version of its Corresponding Source. The information must suffice to ensure that the continued functioning of the modified object code is in no case prevented or interfered with solely because modification has been made.

If you convey an object code work under this section in, or with, or specifically for use in, a User Product, and the conveying occurs as part of a transaction in which the right of possession and use of the User Product is transferred to the recipient in perpetuity or for a fixed term (regardless of how the transaction is characterized), the Corresponding Source conveyed under this section must be accompanied by the Installation Information. But this requirement does not apply if neither you nor any third party retains the ability to install modified object code on the User Product (for example, the work has been installed in ROM).

The requirement to provide Installation Information does not include a requirement to continue to provide support service, warranty, or updates for a work that has been modified or installed by the recipient, or for the User Product in which it has been modified or installed. Access to a network may be denied when the modification itself materially and adversely affects the operation of the network or violates the rules and protocols for communication across the network.

Corresponding Source conveyed, and Installation Information provided, in accord with this section must be in a format that is publicly documented (and with an implementation available to the public in source code form), and must require no special password or key for unpacking, reading or copying.

Additional Terms

“Additional permissions” are terms that supplement the terms of this License by making exceptions from one or more of its conditions. Additional permissions that are applicable to the entire Program shall be treated as though they were included in this License, to the extent that they are valid under applicable law. If additional permissions apply only to part of the Program, that part may be used separately under those permissions, but the entire Program remains governed by this License without regard to the additional permissions.

When you convey a copy of a covered work, you may at your option remove any additional permissions from that copy, or from any part of it. (Additional permissions may be written to require their own removal in certain cases when you modify the work.) You may place additional permissions on material, added by you to a covered work, for which you have or can give appropriate copyright permission.

Notwithstanding any other provision of this License, for material you add to a covered work, you may (if authorized by the copyright holders of that material) supplement the terms of this License with terms:

a. Disclaiming warranty or limiting liability differently from the terms of sections 15 and 16 of this License; or
b. Requiring preservation of specified reasonable legal notices or author attributions in that material or in the Appropriate Legal Notices displayed by works containing it; or
c. Prohibiting misrepresentation of the origin of that material, or requiring that modified versions of such material be marked in reasonable ways as different from the original version; or
d. Limiting the use for publicity purposes of names of licensors or authors of the material; or
e. Declining to grant rights under trademark law for use of some trade names, trademarks, or service marks; or
f. Requiring indemnification of licensors and authors of that material by anyone who conveys the material (or modified versions of it) with contractual assumptions of liability to the recipient, for any liability that these contractual assumptions directly impose on those licensors and authors.

All other non-permissive additional terms are considered “further restrictions” within the meaning of section 10. If the Program as you received it, or any part of it, contains a notice stating that it is governed by this License along with a term that is a further restriction, you may remove that term. If a license document contains a further restriction but permits relicensing or conveying under this License, you may add to a covered work material governed by the terms of that license document, provided that the further restriction does not survive such relicensing or conveying.

If you add terms to a covered work in accord with this section, you must place, in the relevant source files, a statement of the additional terms that apply to those files, or a notice indicating where to find the applicable terms.

Additional terms, permissive or non-permissive, may be stated in the form of a separately written license, or stated as exceptions; the above requirements apply either way.

Termination

You may not propagate or modify a covered work except as expressly provided under this License. Any attempt otherwise to propagate or modify it is void, and will automatically terminate your rights under this License (including any patent licenses granted under the third paragraph of section 11).

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, you do not qualify to receive new licenses for the same material under section 10.

Acceptance Not Required for Having Copies

You are not required to accept this License in order to receive or run a copy of the Program. Ancillary propagation of a covered work occurring solely as a consequence of using peer-to-peer transmission to receive a copy likewise does not require acceptance. However, nothing other than this License grants you permission to propagate or modify any covered work. These actions infringe copyright if you do not accept this License. Therefore, by modifying or propagating a covered work, you indicate your acceptance of this License to do so.

Automatic Licensing of Downstream Recipients

Each time you convey a covered work, the recipient automatically receives a license from the original licensors, to run, modify and propagate that work, subject to this License. You are not responsible for enforcing compliance by third parties with this License.

An “entity transaction” is a transaction transferring control of an organization, or substantially all assets of one, or subdividing an organization, or merging organizations. If propagation of a covered work results from an entity transaction, each party to that transaction who receives a copy of the work also receives whatever licenses to the work the party's predecessor in interest had or could give under the previous paragraph, plus a right to possession of the Corresponding Source of the work from the predecessor in interest, if the predecessor has it or can get it with reasonable efforts.

You may not impose any further restrictions on the exercise of the rights granted or affirmed under this License. For example, you may not impose a license fee, royalty, or other charge for exercise of rights granted under this License, and you may not initiate litigation (including a cross-claim or counterclaim in a lawsuit) alleging that any patent claim is infringed by making, using, selling, offering for sale, or importing the Program or any portion of it.

Patents

A “contributor” is a copyright holder who authorizes use under this License of the Program or a work on which the Program is based. The work thus licensed is called the contributor's “contributor version”.

A contributor's “essential patent claims” are all patent claims owned or controlled by the contributor, whether already acquired or hereafter acquired, that would be infringed by some manner, permitted by this License, of making, using, or selling its contributor version, but do not include claims that would be infringed only as a consequence of further modification of the contributor version. For purposes of this definition, “control” includes the right to grant patent sublicenses in a manner consistent with the requirements of this License.

Each contributor grants you a non-exclusive, worldwide, royalty-free patent license under the contributor's essential patent claims, to make, use, sell, offer for sale, import and otherwise run, modify and propagate the contents of its contributor version.

In the following three paragraphs, a “patent license” is any express agreement or commitment, however denominated, not to enforce a patent (such as an express permission to practice a patent or covenant not to sue for patent infringement). To “grant” such a patent license to a party means to make such an agreement or commitment not to enforce a patent against the party.

If you convey a covered work, knowingly relying on a patent license, and the Corresponding Source of the work is not available for anyone to copy, free of charge and under the terms of this License, through a publicly available network server or other readily accessible means, then you must either (1) cause the Corresponding Source to be so available, or (2) arrange to deprive yourself of the benefit of the patent license for this particular work, or (3) arrange, in a manner consistent with the requirements of this License, to extend the patent license to downstream recipients. “Knowingly relying” means you have actual knowledge that, but for the patent license, your conveying the covered work in a country, or your recipient's use of the covered work in a country, would infringe one or more identifiable patents in that country that you have reason to believe are valid.

If, pursuant to or in connection with a single transaction or arrangement, you convey, or propagate by procuring conveyance of, a covered work, and grant a patent license to some of the parties receiving the covered work authorizing them to use, propagate, modify or convey a specific copy of the covered work, then the patent license you grant is automatically extended to all recipients of the covered work and works based on it.

A patent license is “discriminatory” if it does not include within the scope of its coverage, prohibits the exercise of, or is conditioned on the non-exercise of one or more of the rights that are specifically granted under this License. You may not convey a covered work if you are a party to an arrangement with a third party that is in the business of distributing software, under which you make payment to the third party based on the extent of your activity of conveying the work, and under which the third party grants, to any of the parties who would receive the covered work from you, a discriminatory patent license (a) in connection with copies of the covered work conveyed by you (or copies made from those copies), or (b) primarily for and in connection with specific products or compilations that contain the covered work, unless you entered into that arrangement, or that patent license was granted, prior to 28 March 2007.

Nothing in this License shall be construed as excluding or limiting any implied license or other defenses to infringement that may otherwise be available to you under applicable patent law.

No Surrender of Others' Freedom

If conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot convey a covered work so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not convey it at all. For example, if you agree to terms that obligate you to collect a royalty for further conveying from those to whom you convey the Program, the only way you could satisfy both those terms and this License would be to refrain entirely from conveying the Program.

Use with the GNU Affero General Public License

Notwithstanding any other provision of this License, you have permission to link or combine any covered work with a work licensed under version 3 of the GNU Affero General Public License into a single combined work, and to convey the resulting work. The terms of this License will continue to apply to the part which is the covered work, but the special requirements of the GNU Affero General Public License, section 13, concerning interaction through a network will apply to the combination as such.

Revised Versions of this License

The Free Software Foundation may publish revised and/or new versions of the GNU General Public 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.

Each version is given a distinguishing version number. If the Program specifies that a certain numbered version of the GNU General Public License “or any later version” applies to it, you have the option of following the terms and conditions either of that numbered version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of the GNU General Public License, you may choose any version ever published by the Free Software Foundation.

If the Program specifies that a proxy can decide which future versions of the GNU General Public License can be used, that proxy's public statement of acceptance of a version permanently authorizes you to choose that version for the Program.

Later license versions may give you additional or different permissions. However, no additional obligations are imposed on any author or copyright holder as a result of your choosing to follow a later version.

Disclaimer of Warranty

THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM “AS IS” WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

Limitation of Liability

IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MODIFIES AND/OR CONVEYS THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

Interpretation of Sections 15 and 16

If the disclaimer of warranty and limitation of liability provided above cannot be given local legal effect according to their terms, reviewing courts shall apply local law that most closely approximates an absolute waiver of all civil liability in connection with the Program, unless a warranty or assumption of liability accompanies a copy of the Program in return for a fee.

END OF TERMS AND CONDITIONS