

#### Instituto Politécnico de Viana do Castelo





#### Graduation in Informatics Engineering

#### Project III Course Unit

**Portal de Turismo**

**José Machado, n.º 22344**

**Paulo Macedo, n.º 21603**

#### 2020 - 2021

#### Coordinator: Prof. Doutor. António Cruz | miguel.cruz@estg.ipvc.pt

Index

[1. Introduction and objectives 3](#_Toc52123307)

[2. Technologies, tools, libraries, methodology and project management 4](#_Toc52123308)

[2.1 Programming Environment 4](#_Toc52123309)

[2.2 Methodology and Project Management 5](#_Toc52123310)

[2.3 Technologies and Architecture 5](#_Toc52123311)

[2.3.1 HTML 6](#_Toc52123312)

[2.3.2 Javascript 7](#_Toc52123313)

[2.3.3 CSS 8](#_Toc52123314)

[2.3.4 Bootstrap 8](#_Toc52123315)

[2.3.5 Visual Studio Code 9](#_Toc52123316)

[2.3.6 GitHub 10](#_Toc52123317)

[2.3.7 NodeJS 10](#_Toc52123318)

[2.3.8 Angular 11](#_Toc52123319)

[2.3.9 MongoDB 12](#_Toc52123320)

[2.3.10 ExpressJS 13](#_Toc52123321)

[2.3.11 Mongoose 13](#_Toc52123322)

[3. Use Cases, Users Analysis, Class Diagram 14](#_Toc52123323)

[3.1 Users Analysis 14](#_Toc52123324)

[3.2 Use Case Diagram 14](#_Toc52123325)

[3.3. Class Diagram 15](#_Toc52123326)

[3.4. Database Schema 16](#_Toc52123327)

[4. Developed features 17](#_Toc52123328)

[5. Practical Case/Project Developed 17](#_Toc52123329)

[6. Difficulties & future features 18](#_Toc52123330)

[6.1 Difficulties 18](#_Toc52123331)

[6.2 Future Features 18](#_Toc52123332)

[6.3 Final thoughts 18](#_Toc52123333)

[7. Conlusions and Future Work 19](#_Toc52123334)

[8. Bibliography and Web References 19](#_Toc52123335)

[8.1 Bibliography: 19](#_Toc52123336)

[8.2 Web References: 19](#_Toc52123337)

[9. Appendix 19](#_Toc52123338)

[9.1 Example of Code 19](#_Toc52123339)

[9.2 WebServices API Schema 20](#_Toc52123340)

[9.3 Installation Manual 20](#_Toc52123341)

## Illustration List

[Figure 1 - Illustration of the MEAN Full Stack Developer Framework. 5](#_Toc52123700)

[Figure 2 - Illustration of the project Architecture. 7](#_Toc52123701)

[Figure 3 - Illustration of the users analysis. 15](#_Toc52123702)

[Figure 4 - Illustration of the use case diagram. 16](#_Toc52123703)

[Figure 5 - Illustration of the class diagram. 17](#_Toc52123704)

[Figure 6 - Illustration of the database diagram. 18](#_Toc52123705)

[Figure 7 - Illustration of the Web Services Diagram. 19](#_Toc52123706)

[Figure 8 - Illustration of the web page of the functionality X. 20](#_Toc52123707)

Introduction and objectives

This project was developed associated with the curricular unit of Project III of the 3rd year of the Graduation Degree in Informatics’ Engineering (Computer Science) at the School of Technology and Management of the Polytechnique institute of Viana do Castelo.

The **main objective** of this project is the development of a web portal that allows the sale of tourist products or the reservation of tourist services.

The **scope of this project** is a development of a web platform at IPVC and was proposed by Professor António Cruz.

We build an e-commerce web application that is able to have an authentication system that supports 3 types of users: administrator, shopkeeper or supplier and costumer. There is a fourth user called anonymous who is not part of the authentication system.

The administrator is able to register and deactivate other administrators as well as deactivate shopkeepers and suppliers. The shopkeeper or supplier can create his online store, add his services and products and also view his reservations and orders, respectively. The customer has his own area where he can view the products or services and evaluate them. The customer has a shopping cart where he can order products, however he has to login to do this.

In order to make this project, our **methodology** was to search for related information in GitHub projects that use similar technologies and on YouTube watching tutorials and step-by-step guides. When something just didn’t work on our development, we user Stack Overflow to find answers or try different approaches.

**In the following report** is possible to find an introduction to this project, the explanation of the project architecture and information about MEAN stack. After that, we have the section which has a description of the main technologies, tools, and libraries used. In the next title we show the class diagram with all the entities and relations represented in our web application, and the uses-cases diagram.

# 2. Technologies, tools, libraries, methodology and project management

## 2.1 Programming Environment

MEAN is a collection of technologies (**MongoDB**, **Express.js**, **Angular.js**, and **Node.js**) associated with JavaScript, which is used to build web-based applications. MEAN stack is responsible for the development of each component of website development from client-side/server-side to database handling, and all these are based on one technology, i.e., **JavaScript**.

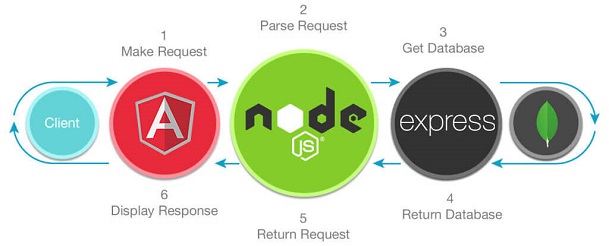




Figure 1 - Illustration of the MEAN Full Stack Developer Framework.

**Technologies That Come Under MEAN:**

* MongoDB: It handles the database of web applications.
* Express: It is a Node.js framework used to build APIs for web applications.
* Angular: It is a frontend development framework majorly maintained by Google.
* Node: It is used to handle the server-side.

…….

…….

## 2.2 Methodology and Project Management

…….

…….

…….

In this project I used trello to keep track of what i needed to do, or plan what to do next, and i also shared the board with the teacher, so he could add some cards as well. For this project, i decided to divide my Trello Board, in five lists: To Do, Doing, Done, Documentation, Repository. Thus, The “To Do”, “Doing” and “Done” lists are pretty explanatory. The “Documentation” list had the link to a Google Doc containing the bibliography. The “Repository” list had the link to an OneDrive shared folder, so i could share all the work that I was doing (ex: mockups, mini-examples, final project, etc…).

…….

## 2.3 Technologies and Architecture

The architecture of the project is composed by a set of frameworks and technologies…….

…….

…….

The next Figure illustrates the general scheme.

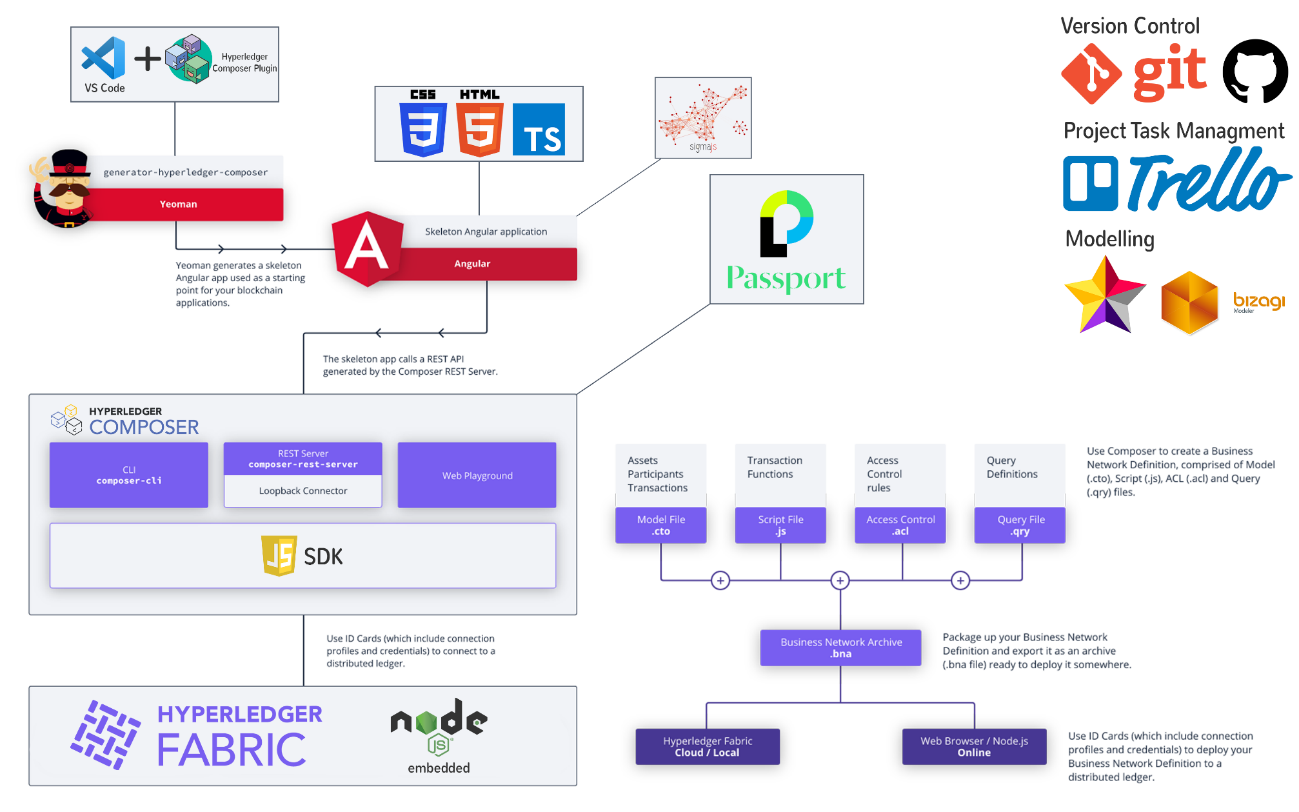


Figure 2 - Illustration of the project Architecture.

The architecture is divided in layers and is interconnects the hyperledger Fabric with nodejs to the Composer SDK………. Continuar a explicar a interigação de cada componente da framework.

Para a modelação do project foi utilizada a framework xyx, o trello para registo de tarefas e o GIT para controlo de versões…

…….

……

### 2.3.1 HTML

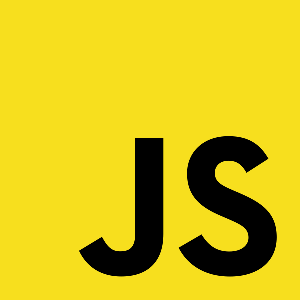


HTML is a computer language devised to allow website creation. These websites can then be viewed by anyone else connected to the Internet. It is relatively easy to learn, with the basics being accessible to most people in one sitting; and quite powerful in what it allows you to create. It is constantly undergoing revision and evolution to meet the demands and requirements of the growing Internet audience under the direction of the W3C, the organization charged with designing and maintaining the language.

Some important web links:

* Wek Link 1; <http://www.xpto.com>
* Web Link 2: <http://www.xpto.com>
* ….
* Web Link n: <http://www.xpto.com>

### 2.3.2 Javascript



JavaScript is a scripting or programming language that allows you to implement complex features on web pages — every time a web page does more than just sit there and display static information for you to look at — displaying timely content updates, interactive maps, animated 2D/3D graphics, scrolling video jukeboxes, etc. — you can bet that JavaScript is probably involved. It is the third layer of the layer cake of standard web technologies, two of which (HTML and CSS) we have covered in much more detail in other parts of the Learning Area.

Some important web links:

* Wek Link 1; <http://www.xpto.com>
* Web Link 2: <http://www.xpto.com>
* ….
* Web Link n: <http://www.xpto.com>

### 2.3.3 CSS



CSS stands for Cascading Style Sheets with an emphasis placed on “Style.” While HTML is used to structure a web document (defining things like headlines and paragraphs, and allowing you to embed images, video, and other media), CSS comes through and specifies your document’s style—page layouts, colors, and fonts are all determined with CSS. Think of HTML as the foundation (every house has one), and CSS as the aesthetic choices (there’s a big difference between a Victorian mansion and a mid-century modern home).

Some important web links:

* Wek Link 1; <http://www.xpto.com>
* Web Link 2: <http://www.xpto.com>
* ….
* Web Link n: <http://www.xpto.com>

### 2.3.4 Bootstrap

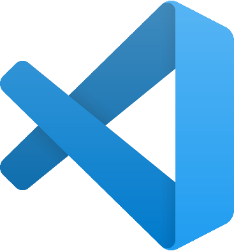


Bootstrap is a free and open-source CSS framework directed at responsive, mobile-first front-end web development. It contains CSS- and (optionally) JavaScript-based design templates for typography, forms, buttons, navigation and other interface components. Bootstrap is a web framework that focuses on simplifying the development of informative web pages (as opposed to web apps). The primary purpose of adding it to a web project is to apply Bootstrap's choices of color, size, font and layout to that project. As such, the primary factor is whether the developers in charge find those choices to their liking. Once added to a project, Bootstrap provides basic style definitions for all HTML elements. The result is a uniform appearance for prose, tables and form elements across web browsers. In addition, developers can take advantage of CSS classes defined in Bootstrap to further customize the appearance of their contents. For example, Bootstrap has provisioned for light- and dark-colored tables, page headings, more prominent pull quotes, and text with a highlight.

Some important web links:

* Wek Link 1; <http://www.xpto.com>
* Web Link 2: <http://www.xpto.com>
* ….
* Web Link n: <http://www.xpto.com>

### 2.3.5 Visual Studio Code



Visual Studio Code is a source-code editor developed by Microsoft for Windows, Linux and macOS. It includes support for debugging, embedded Git control and GitHub, syntax highlighting, intelligent code completion, snippets, and code refactoring. It is highly customizable, allowing users to change the theme, keyboard shortcuts, preferences, and install extensions that add additional functionality. The source code is free and open source. The compiled binaries are freeware and free for private or commercial use.

Some important web links:

* Wek Link 1; <http://www.xpto.com>
* Web Link 2: <http://www.xpto.com>
* ….
* Web Link n: <http://www.xpto.com>

### 2.3.6 GitHub

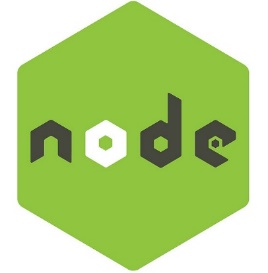


GitHub is an application you can use to store your code on the web. You may use it for something as simple as free cloud storage for your projects, or you may use it to show your code to potential employers interested in assessing your coding skills. However, GitHub is much more than simple code storage; it is a tool used by individual developers and teams alike all across the world to collaborate with each other on virtually any kind of project imaginable.

Some important web links:

* Wek Link 1; <http://www.xpto.com>
* Web Link 2: <http://www.xpto.com>
* ….
* Web Link n: <http://www.xpto.com>

### 2.3.7 NodeJS



The Node.js run-time environment includes everything you need to execute a program written in JavaScript. Node.js came into existence when the original developers of JavaScript extended it from something you could only run in the browser to something you could run on your machine as a standalone application. Now you can do much more with JavaScript than just making websites interactive. JavaScript now has the capability to do things that other scripting languages like Python can do. Both your browser JavaScript and Node.js run on the V8 JavaScript runtime engine. This engine takes your JavaScript code and converts it into a faster machine code. Machine code is low-level code which the computer can run without needing to first interpret it.

Some important web links:

* Wek Link 1; <http://www.xpto.com>
* Web Link 2: <http://www.xpto.com>
* ….
* Web Link n: <http://www.xpto.com>

### 2.3.8 Angular

Uma imagem com símbolo, paragem, exterior, vermelho

Descrição gerada automaticamente

AngularJS is a JavaScript-based open-source front-end web framework mainly maintained by Google and by a community of individuals and corporations to address many of the challenges encountered in developing single-page applications. It aims to simplify both the development and the testing of such applications by providing a framework for client-side model–view–controller (MVC) and model–view–viewmodel (MVVM) architectures, along with components commonly used in rich Internet applications. The AngularJS framework works by first reading the Hypertext Markup Language (HTML) page, which has an additional custom HTML attributes embedded into it. Angular interprets those attributes as directives to bind input or output parts of the page to a model that is represented by standard JavaScript variables. The values of those JavaScript variables can be manually set within the code or retrieved from static or dynamic JSON resources.

Some important web links:

* Wek Link 1; <http://www.xpto.com>
* Web Link 2: <http://www.xpto.com>
* ….
* Web Link n: <http://www.xpto.com>

### 2.3.9 MongoDB



MongoDB is a document-oriented NoSQL database used for high volume data storage. MongoDB is a database which came into light around the mid-2000s. It falls under the category of a NoSQL database. Some features of the MongoDB are the following: each database contains collections which in turn contains documents. Each document can be different with a varying number of fields. The size and content of each document can be different from each other, the document structure is more in line with how developers construct their classes and objects in their respective programming languages, developers will often say that their classes are not rows and columns but have a clear structure with key-value pairs, as seen in the introduction with NoSQL databases, the rows (or documents as called in MongoDB) doesn't need to have a schema defined beforehand. Instead, the fields can be created on the fly and the data model available within MongoDB allows you to represent hierarchical relationships, to store arrays, and other more complex structures more easily.

Some important web links:

* Wek Link 1; <http://www.xpto.com>
* Web Link 2: <http://www.xpto.com>
* ….
* Web Link n: <http://www.xpto.com>

### 2.3.10 ExpressJS



Express is the most popular Node web framework and is the underlying library for a number of other popular Node web frameworks. It provides mechanisms to many things such as write handlers for requests with different HTTP verbs at different URL paths (routes), integrate with "view" rendering engines in order to generate responses by inserting data into templates, set common web application settings like the port to use for connecting, and the location of templates that are used for rendering the response, add additional request processing "middleware" at any point within the request handling pipeline.

Some important web links:

* Wek Link 1; <http://www.xpto.com>
* Web Link 2: <http://www.xpto.com>
* ….
* Web Link n: <http://www.xpto.com>

### 2.3.11 Mongoose



Mongoose is an Object Data Modeling (ODM) library for MongoDB and Node.js. It manages relationships between data, provides schema validation, and is used to translate between objects in code and the representation of those objects in MongoDB. Mongoose supports both promises and callbacks. Mongoose provides an incredible amount of functionality around creating and working with schemas. Mongoose currently contains eight SchemaTypes that a property is saved as when it is persisted to MongoDB (String, Number, Date, Buffer, Boolean, Mixed, ObjectId, Array).

Some important web links:

# 3. Use Cases, Users Analysis, Class Diagram

## 3.1 Users Analysis

Figure 3 - Illustration of the users analysis.

## 3.2 Use Case Diagram

Diagrama

Descrição gerada automaticamente

Figure 4 - Illustration of the use case diagram.

As previously mentioned, one of the administrator's requirements is to disable other administrators. However, not all admins will be able to perform this task, so there is an admin with more privileges than the others (General Administrator).

An anonymous customer registering as a customer will become a customer for the system, which will allow access to all the customer's features, in case of registering as a supplier or storekeeper, will be able to perform the tasks relating to it.

## 3.3. Class Diagram

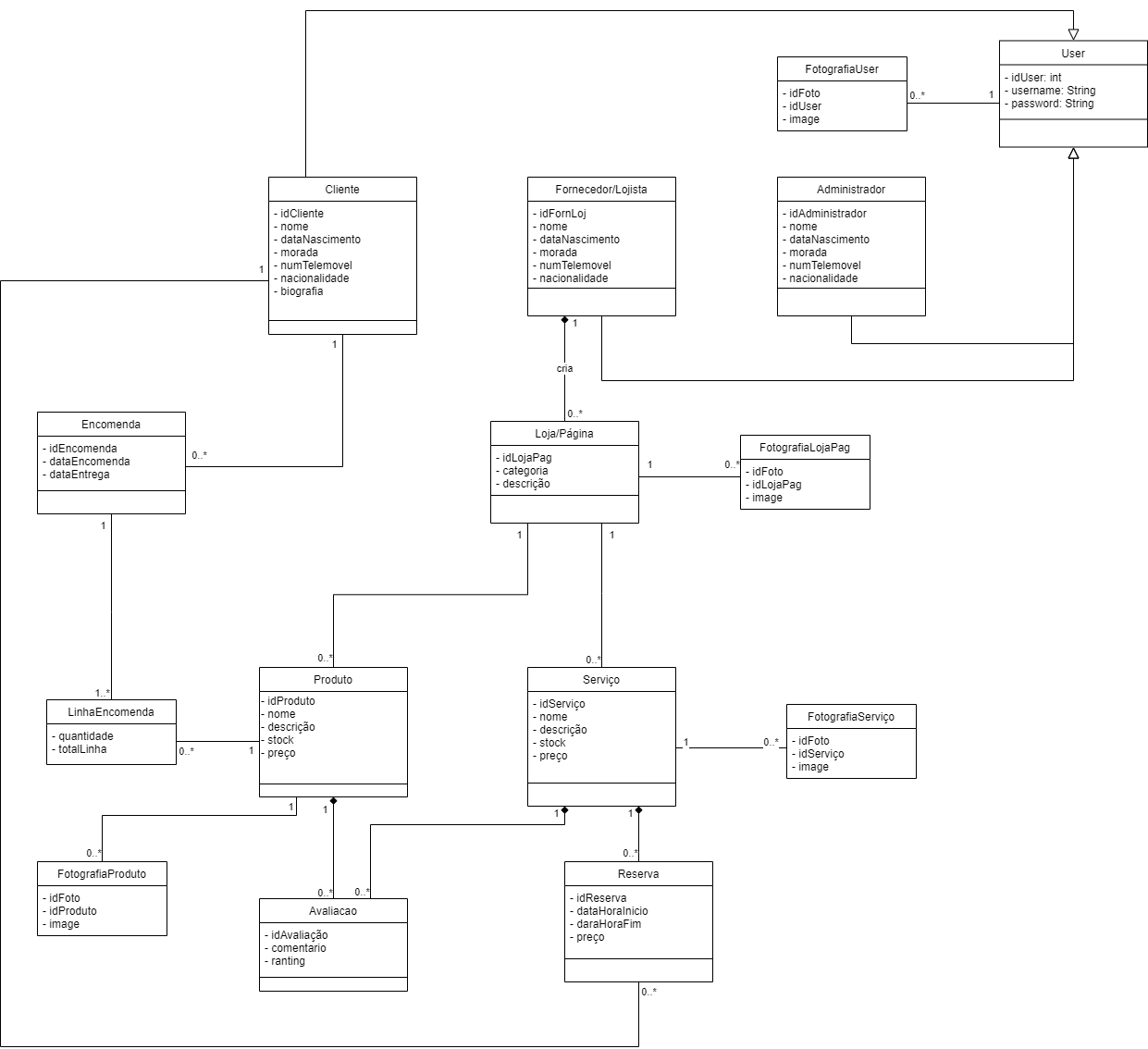


Figure 5 - Illustration of the class diagram.

The anonymous one is not represented in the class diagram because it only performs frontend tasks that do not interfere with the system other than the registration. However, when the registration is done, he will not be an anonymous but a customer or shopkeeper or supplier and they will perform tasks that will interfere with the system.

## 3.4. Database Schema

…….explicar detalhadamente. Ao colocar imagens escrever um parágrafo sobre o que pretende descrever…

Uma imagem com mapa

Descrição gerada automaticamente

Figure 6 - Illustration of the database diagram.

# 4. Developed features

…….explicar detalhadamente. Ao colocar imagens escrever um parágrafo sobre o que pretende descrever…

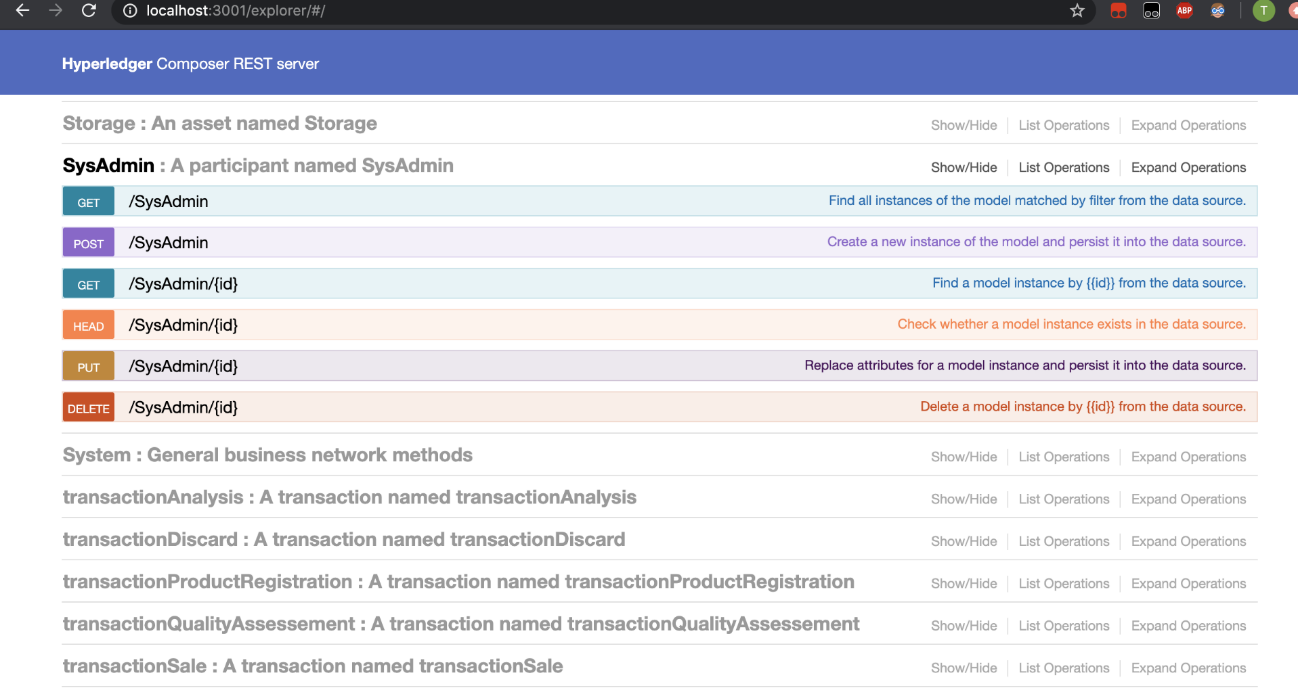


Figure 7 - Illustration of the Web Services Diagram.

…….

# 5. Practical Case/Project Developed

.. Esta secção é muito importante, pois reflecte o que realmente foi iimplementado…

…….explicar detalhadamente. Ao colocar imagens escrever um parágrafo sobre o que pretende descrever…

……. Explicar detalhadamente o que foi implementado no projeto, incluindo o maior número de screenshoots para potenciar o trabalho desenvolvido..

**…..**

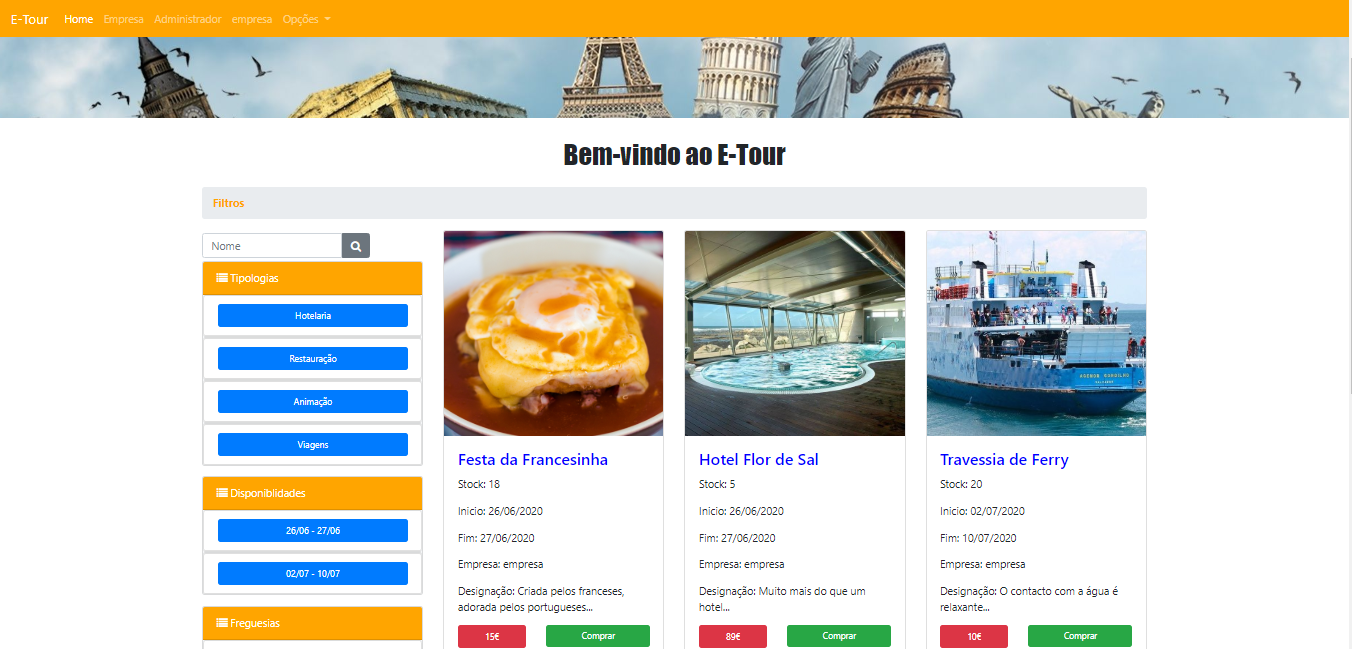
****

Figure 8 - Illustration of the web page of the functionality X.

**..**

**..**

# 6. Difficulties & future features

## 6.1 Difficulties

…….explicar detalhadamente. Ao colocar imagens escrever um parágrafo sobre o que pretende descrever…

…….

## 6.2 Future Features

…….explicar detalhadamente. Ao colocar imagens escrever um parágrafo sobre o que pretende descrever…

…….

## 6.3 Final thoughts

…….explicar detalhadamente. Ao colocar imagens escrever um parágrafo sobre o que pretende descrever…

…….

# 7. Conlusions and Future Work

…….explicar detalhadamente. Ao colocar imagens escrever um parágrafo sobre o que pretende descrever…

…….

# 8. Bibliography and Web References

## 8.1 Bibliography:

* Bronson, J and Martin C. (2020) “Laravel Manual 2.1”. Wrox Editors.
* Kimball, R. (2016) “The Data Warehouse Toolkit”. OReilly editors.

## 8.2 Web References:

……… Na bibliografia e referências WEB colocar uma descrição do objectivo do site e depois o link.

* Hyperledger Composer: <https://hyperledger.github.io/composer/latest/introduction/introduction.html>
* Quality schemes explained: <https://ec.europa.eu/info/food-farming-fisheries/food-safety-and-quality/certification/quality-labels/quality-schemes-explained_en>

………

# 9. Appendix

## 9.1 Example of Code

…….Se aplicável e se fizer sentido…

…….Colocar Screenshoots ou colocar o código numa tabela com forma to Courrier new 10

|  |
| --- |
| Public void procedureXTPO/ar1, arg,2)  {  …  ..  } |

…….

## 9.2 WebServices API Schema

…….Se aplicável e se fizer sentido mediante o tipo de projeto…

…… indicar a especificação da API, que ode ser com prints do uso da framework POSTMAN (ou outra).

**….**

## 9.3 Installation Manual

…….Se aplicável e se fizer sentido mediante o tipo de projeto…

**….**

## 9.4 Mockups

Em conjunto com este relatório, encontra-se um anexo em formato PDF (“Mockups.pdf”), onde são apresentados os mockups da aplicação web.