

SOMMAIRE

INTRODUCTION

PROJET: GESTION DE STOCK ENTRE FOURNISSEUR ET GRANDE SURFACE

stock management platform to facilitate relations and order taking between suppliers and supermarkets.

the objective for

- Each vendor is :
 - to manage its stock -> CRUD
 - to exchange with his customers
 - to organize their delivery in an optimal way

- Each department store is
 - To have the list of their suppliers -> CRUD
 - To manage their stock -> CRUD
 - View and modify future deliveries -> CRUD

Introduction

Within the framework of the installation of a Web application aiming at facilitating the relations and the order taking between the suppliers and the great surfaces, this document gathers important elements which will make it possible to lead to the solution. Thus in the document we will find the functional analysis of the application.

The part dedicated to the functional analysis will detail the different user stories as well as the scenarios linked to them. We will also find the people who will interact with the system as well as the entities participating in the different use cases.

In this functional analysis file the emphasis is on a more detailed presentation of the US.

1. Objective of the document

The objective of this document is to present the functional analysis of the system, of the application.

2. About the project

In view of the needs that were expressed in the specifications and the resulting product backlog, it is important to recall once again that the objective of the application is to propose a computerized solution that will make it easier for any organization wishing to place its order with the supplier to be delivered in a reduced time frame. For the time being, the application will be named Pentachore in all documents.

3. Functional analysis

3 3.1 Roles

The Pentachore application mainly involves the interaction of suppliers and large surfaces. The administrator will have more global actions. Therefore, the roles we will find will be linked to these two actors.

Nom du rôle	Role description
Big surface	The department store is the organization that places an order with the supplier through the
Vendor	The supplier is the organization that delivers orders placed by the department store through the application
Administrateur	The administrator, as his name indicates, has the role of administering the sendMonTchop platform, whether it concerns the management of large surfaces and suppliers

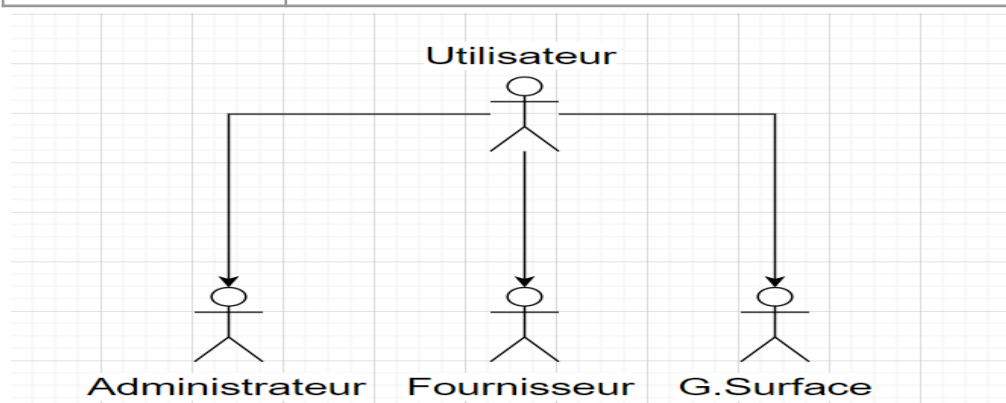


Tableau 1: Rôles des acteurs

3.2 Presentation of the User stories related to authentication

3.2.1 Theme Authentication management

3.2.1.1 Login

#ID: 1	Type :
Fonctionnelle	
Description As a department store, administrator and supplier I wish to log in In order to access my user area [store inventory]	
Management rules If the login credentials are incorrect Then return an error message such as incorrect phone number or password	

Acceptance criteria Authentication is only possible by entering the following fields: telephone number or e-mail and password Authentication must involve a redirection to the user area Two-factor authentication must be possible The first and last name must be visible from the user area	
Acceptance tests « Nominal authentication scenario » As I am a user with login details When I enter my phone number or email and my password And click on the "login" button Then I am redirected to my user area And my username is displayed in the top right-hand corner of the menu bar « Incorrect authentication scenario » As I am a user with a login, characterised by a telephone number and a password When I click on the "login" button Then an error message appears notifying that the login credentials are wrong	
Point d'effort __ 8 __	Valeur _100_

3.2.1.2 Disconnect

#ID: 2	Type :
Fonctionnelle	
Description As a department store, administrator and supplier I wish to log out To stop using my user space	
Management rules N/A	
Acceptance criteria Disconnection is only possible when the user is logged in The disconnection implies a redirection to the home page of the site	

Acceptance tests	
« Disconnection scenario » Since I am a user with login credentials When I click on the "logout" button Then I am redirected to the home page of the site	
Point d'effort __5__	Valeur _100_

3.2.1.3 Change the password

#ID: 3	Type :
Fonctionnelle	
Description As an inventory manager, administrator and supplier I want to change my password In order to use my new password to log in	
Management rules Instant logout and invitation to reconnect to the solution with the new password	
Acceptance criteria Password modification is only possible when the user is logged in Changing the password implies a redirection to the login form	
Acceptance tests “ Password change scenario” Since I am a user with a login and password When I click on the "change my password" button Then I am redirected to the modification page then connection	
Point d'effort __5__	Valeur _100_

3.2.1.4 Reset password

#ID: 4	Type :
Fonctionnelle	
Description	

As a retailer, administrator and supplier I want to reset my password In order to use the solution again
Management rules Mail sent to the user with a link to reset his password
Acceptance criteria The password reset is only possible if the email sent to receive the password reset link corresponds to a user of the solution
Acceptance tests “ Password reset scenario” Since I am a user who has a login but forgot his password When I click on the "reset my password" button Then I receive a reset link by email that redirects me to the reset page
Point d’effort __5__ Valeur _100_

3.2.2 Inventory Management Theme

3.2.2.1 See the stocks

#ID: 5	Type :
Fonctionnelle	
Description As a department store, administrator and supplier I wish to see the stocks In order to have the visibility on the present stocks	
Management rules Stock pages with product list and quantity	
Acceptance criteria Inventory visibility is only possible when the user is logged in Stock visibility involves a redirection to the stock page	
Acceptance tests “ Inventory visibility scenario” Since I am a logged-in user When I click on the stock button	

Then I am redirected to the stock page with all the products present and the quantity	
Point d'effort __5__	Valeur _100_

3.2.2.2 Add a product

#ID: 6	Type :
Fonctionnelle	
Description As a supplier I want to add a product In order to update the stock I have	
Management rules Stock pages with the possibility to add a product	
Acceptance criteria Adding a product is only possible when the user is logged in Adding a product implies a redirection to a pop-up giving the possibility to add a product	
Acceptance tests “ Scenario for adding a product” Since I am a logged-in user When I click on the "add a product" button Then a pop-up appears allowing me to add a product (form)	
Point d'effort __5__	Valeur _100_

3.2.2.3 Delete a product

#ID: 7	Type :
Fonctionnelle	
Description As an inventory manager, administrator and supplier I want to delete a product	

In order to update the stock I have
Management rules Stock pages with the possibility to delete a product
Acceptance criteria Deleting a product is only possible when the user is logged in The deletion of a product implies a redirection to a pop-up giving the possibility to delete a product
Acceptance tests “ Scenario for deleting a product” Since I am a user with credentials and I am logged in When I click on the delete button on a product Then a pop-up appears asking me if I am sure I want to delete the product and if yes then the product is deleted and the stock is updated
Point d’effort __5__ Valeur _100_

3.2.2.4 Update a product

#ID: 8 Fonctionnelle	Type :
Description As a retailer and supplier I want to modify a product In order to update the stock I have	
Management rules Stock pages with the possibility to modify a product	
Acceptance criteria The modification of a product is only possible when the user is logged in The modification of a product implies a redirection to a pop-up giving the possibility to modify a product	
Acceptance tests “ Product modification scenario” Since I am a logged-in user When I click on the edit button on a product Then a pop-up appears giving me the possibility to modify a product and update the stock	

Point d'effort __5__	Valeur _100_

3.2.3 Messaging theme

3.2.3.1 See the messages

#ID: 9	Type :
Fonctionnelle	
Description As a department store, administrator and supplier I wish to see my messages In order to read my messages	
Management rules Messaging pages	
Acceptance criteria Messaging is only accessible when the user is logged in Messaging involves a redirection to a messaging page	
Acceptance tests “ Message reading scenario” Since I am a user with credentials and I am logged in When I click on the mail button Then a page opens and I see the message history and the new messages received	
Point d'effort __5__	Valeur _100_

3.2.3.2 Send a message

#ID: 10	Type :
Fonctionnelle	
Description As a department store, administrator and supplier I want to see my messages In order to send a message	

Management rules Messaging pages with the ability to send a message	
Acceptance criteria Messaging is only accessible when the user is logged in Messaging involves a redirection to a messaging page with the possibility to send a message	
Acceptance tests “ Scenario for sending a message” Since I am a user with credentials and I am logged in When I click on the mail button and then on send a message Then a pop-up opens, I can then fill in the recipient, write my message and send it	
Point d’effort __5__	Valeur _100_

1. Architectural choice

1.1 General architecture

The first choice that was made concerns the general architecture of the application. The choice that was made is based on the following axes:

The durability of the application: indeed, the objective being that the application is long-lasting, it would be wise that the chosen architecture be flexible and scalable

Scalability: the application should be reliable in case of heavy load

Maintainability: this important element is strongly taken into account because the durability of the application depends on the ability to maintain it as easily as possible

Thus, the first element to remember is that the application will be a so-called Fullstack application in the sense that it will feature a front-end that will be dedicated to the client side and a back-end that will be dedicated to the server side. The backend will be characterized by a REST API that will expose several web services with which the front-end will interact. The data being persisted in base there will be of course a database in the center of the interactions.

This 1st choice is made first of all because it allows to satisfy the main criteria which were mentioned, moreover it is pledge of modernity and separation of the concerns, the context in which we are situated being more and more heterogeneous, the choice to set up a REST API makes sense more especially as that would allow to give a simpler vision of the application and the exchanges which we will find.

In addition, the presence of a cross platform frontend that will only deal with the GUI interactions aspect will only reinforce the cohesion of the application and will better highlight this separation of concerns. To summarize, the figure below illustrates better my comments on this 1st choice

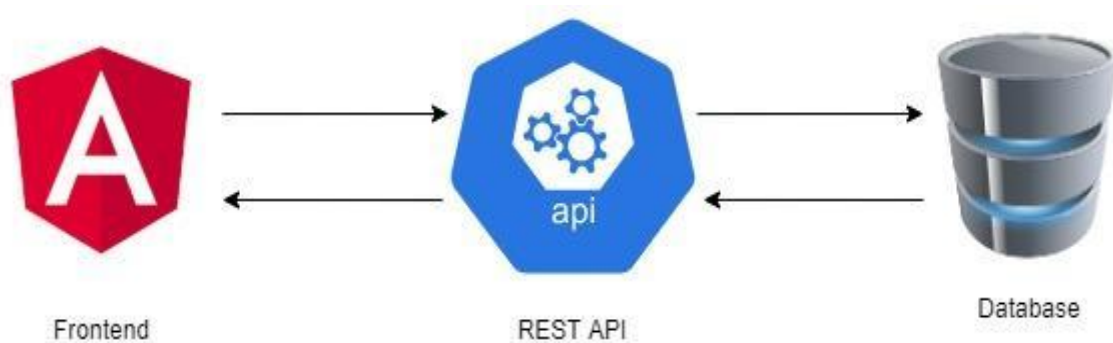


Figure 1: Architecture générale

1.2 Detailed architecture

The functional analysis allowed us to highlight different functional areas on the :

- User
- Role
- Large area
- Message

Based on these domains, the chosen architecture is based on a separation by functional domain. This means that instead of having a classical layer architecture as in the figure below :

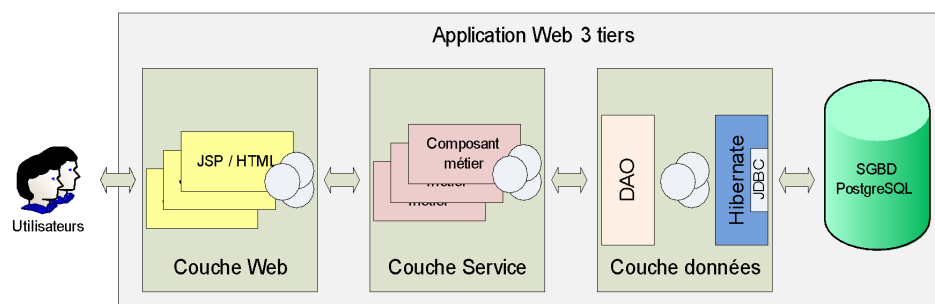


Figure : Architecture en couche

We will use a realization approach driven by functional area, and in each functional area we will find the different layers associated with these areas, this architecture is vertical and it is chosen because it offers the following advantages :

- The separation of concerns makes it easier to identify each feature of the application
- In the perspective of a migration to a micro service architecture, this architecture will allow to make changes more easily

Thus the architecture will have the following framework :

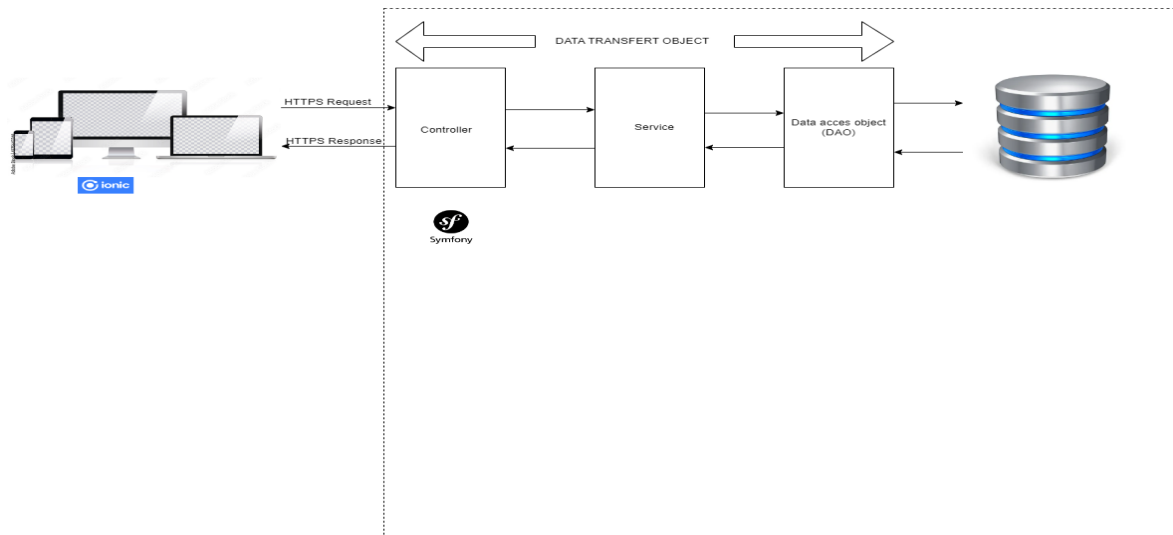


Figure 3: Architecture détaillée

present layer / functional area	Module	Comment
Client / Presentation layer	Front-end	The client represents here the browser which allows to display the web pages, this element will render the presentation layer which is really the front-end of the project
Web Layer	Controller	A controller allows to encapsulate the logic associated to the actions performed by a user, in our case it exposes endpoints linked to the actions performed by a user
Service layer	Service	This layer contains the business logic of the application
Data access layer	Data Access Object (DAO)	Design pattern proposing to group access to persistent data in specific classes. The basic operations are create, read, update, delete.
Data Transfer Object (DTO)	Data Transfer Object (DTO)	DTOs will be used to separate direct interactions between data and higher layers

Choice of technologies :

Angular for the Front-End will be implemented in order not to lose time in relation to the skills of each developer. In the same logic, the database will be in MySQL and the back in Symfony or Java.

List of tasks :

Front end :

- Creation of the project
- Creation of the different folders (shared, interfaces...)
- Creation of the different pages
 - Auth
 - Stocks
 - Products
 - Configuration
- Front-end integration
- Unit testing

Back end :

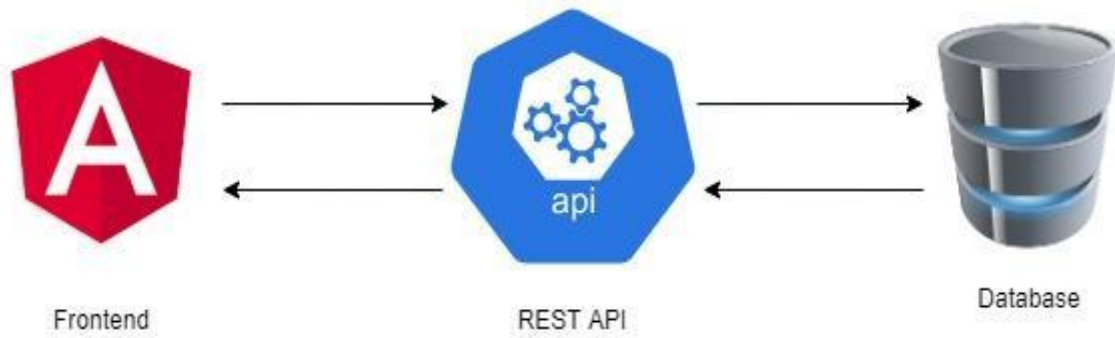
- Creation Bdd
- Set up project
- Creation of the different possible routes
- Auth
- Unitary test

Risks :

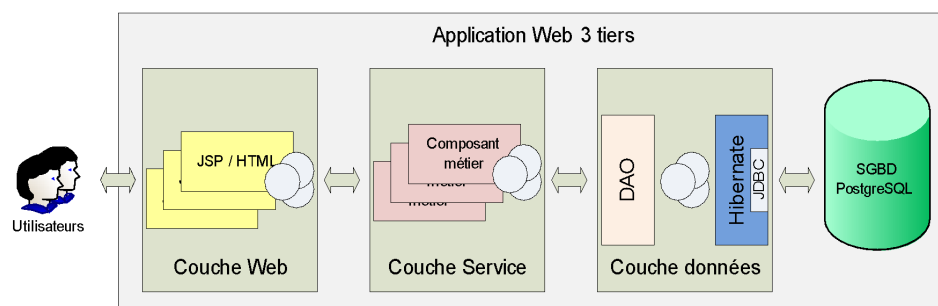
- Database hacking -> Loss of data -> 90% probability of realization
-> Protect yourself by looking at all the known vulnerabilities.
- Lack of space on the server -> Server overloaded -> 50% probability -> Follow the evolution of daily users and act accordingly
- Sickness -> Absence of dev for an indefinite period -> 50% probability -> Take this into account in the planning of tasks
- Lack of time -> several possible causes -> 50% -> Follow the implemented scrum to the letter

DAT :

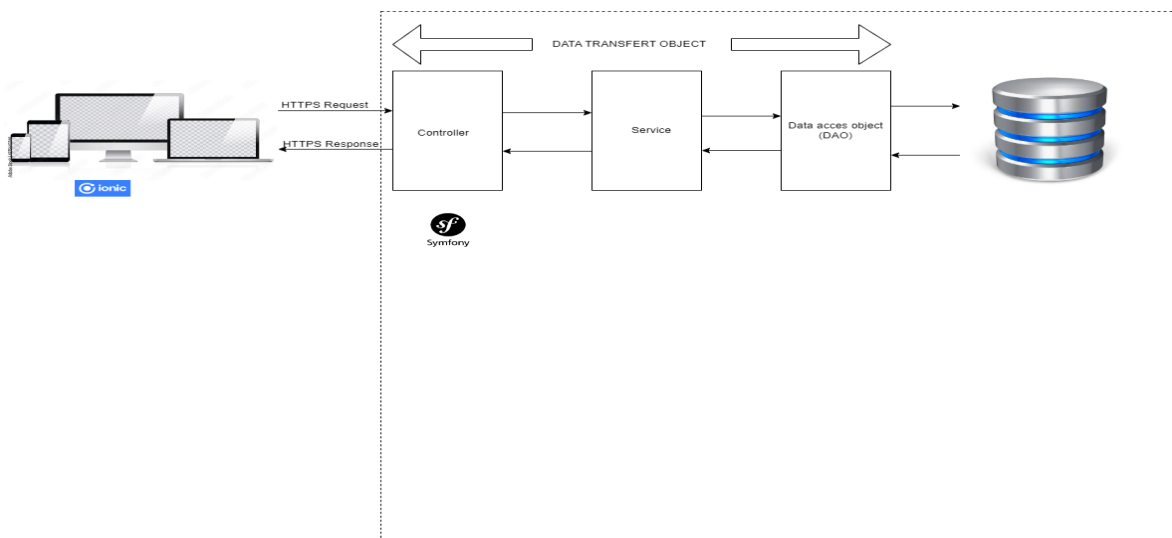
- operational representation



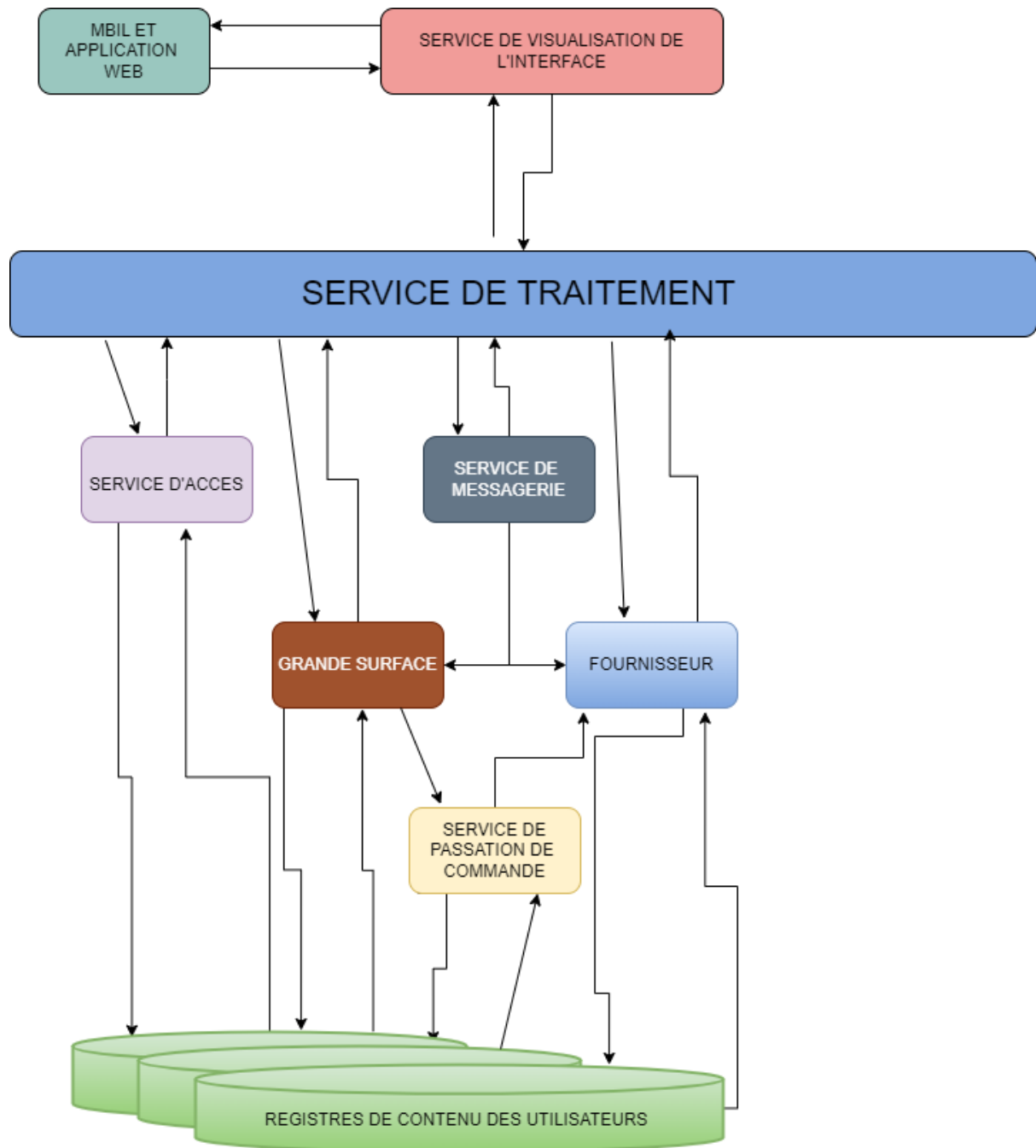
- functional representation



- application representation



- technical representation



- Costs

- Dev : 12 915€ / month
- Server : OVH -> 44€ / month
- Apple store license -> 90 € / year
- Play store license -> 10 € for life
- Trainings 1000 €

- Total cost : 14 059 €.

Plan de déploiement

Available resources: 2 servers

A server will be used to host the database. It will be necessary to install MySql in order to create the database.

The second server will be allocated to the Front-end and Back-End part. For this, we need to set up a server with nodeJS in 15.0.6 minimum in order to run Angular 15 and all the dependencies. For the back-end, we need to install php 8.0.2 in order to run Symfony without any problem. We will also have to add composer in order to install the libraries.

The project will be a web app so there is no need to install anything on the customer's side, just give him a user account and a password.

The project will be deployed on our infrastructure. Our database server will be set up with a sql script that will create the necessary tables. For the second server, NGNIX will allow the connection to the project. A Linux will also be installed on the project.

For the deployment, it is first to create the database, then the back-end and then the front because the back is dependent on the database and the front is dependent on the back-end.

For the maj, it will be necessary to put in maintenance the solution on the front in order to avoid the losses of data then to go in the same order as before, i.e. first the database then the back end and finally the front.

Link github front :

https://github.com/thomas-gfr/architecture_logiciel_front

Link github Back :

https://github.com/jeanyvesellokoea/back_stock_grandesurface