

**Conception And Development**

**Of Central Purchasing Application**

**SAE Software Architecture Engineering**

**Bridge Buyer**

**Realised by : -**

**Khater khrouf**

**Soulaima ben Romdhane**

**Houssem kouki**

**Farah Kammoun**

**Malek Soufi**

**Rawya Ferjeni**

**2022- 2023**

**Global Introduction**

Technological development and progress in digitization permeate all areas of life, of course not excluding companies and their activities and individual business processes.

Many countries have an opportunity to digitally transform public procurement systems to achieve enhanced efficiency, accountability, transparency, and participation of small and medium enterprises (SMEs). Digitally transforming public procurement would also accelerate national development objectives, such as enhancing public service delivery, developing human capital and the private sector, and gender empowerment.

When digital transformation should reach the maximum effect, digitalization should include business processes as a whole part. In the area of central purchasing which means the management of providers and all order types.

Artificial intelligence can support supply chain management. Analyzing the market situation, taking into account the reliability of providers and proposing optimal contractual conditions, better monitor the development of individual commodities. The system can even recommend what discounts to require.

Central purchasing applications are widely used all over the world by government and public organizations to streamline procurement processes and increase efficiency.

Its use varies by country and region, depending on the specific regulations, policies, and processes that apply. As a result, the use of central purchasing applications is likely to increase in the future as more organizations look to improve their procurement processes and reduce costs.

Contens

[Introduction 5](#_2et92p0)

[I. Presentation of the company 5](#_3dy6vkm)

[1. SSD Team: 5](#_4d34og8)

[1.1. Services 6](#_2s8eyo1)

[2. Hydatis: 6](#_3rdcrjn)

[II. Study of the existing: 7](#_lnxbz9)

[1. Tunisian central purchasing office: 7](#_35nkun2)

[1.1. Adis Tunisia 7](#_1ksv4uv)

[2.International central purchasing office: 8](#_5rigpvdomza6)

[2.1 Achat Centrale 8](#_3j2qqm3)

[2.2 Qantis: 9](#_1ci93xb)

[III. Criticism of the existing: 10](#_qsh70q)

[1. Problematic: 11](#_3as4poj)

[2. Proposed Solution: 11](#_1pxezwc)

[IV. Sustainable Development 11](#_6bthxkbdq9me)

[1. Decent work and Economic growth 11](#_oquab6hoaln6)

[2. Industry, innovation, and infrastructure 12](#_9fj9b75kj7zj)

[3. Responsible consumption and production 12](#_y7wp9ywy52k2)

[Conclusion: 13](#_147n2zr)

[Introduction 14](#_qwopijhhpeb9)

[I. Requirements and specifications 14](#_iuruq3p8yh42)

[1. Actors: 14](#_3o7alnk)

[2. Functional requirements: 15](#_ihv636)

[3. Global Use Case Diagram: 16](#_32hioqz)

[4. Detailed Use Case diagrams of each module 17](#_nuyhoaqmxgt5)

[5. Non-functional requirements: 21](#_41mghml)

[II- Methodology Adopted: 22](#_2grqrue)

[1. The Unified Process: 22](#_vx1227)

[2. Physical Architecture: 23](#_2u6wntf)

[3. Logical Architecture: 24](#_28h4qwu)

[4. Microservice Architecture 24](#_5339rbf2zc93)

[5. Class Diagram: 29](#_xmyzfwz54mwt)

[6. Object sequence diagram 30](#_g5p817r9ojbw)

[7. Mock-ups: 30](#_4k668n3)

[Conclusion 33](#_2dlolyb)

[Chapter III: Implementation 34](#_sqyw64)

[I. Working environment 34](#_4bvk7pj)

[II. Spring 38](#_ycuo60d0x34t)

[Introduction 38](#_2iq8gzs)

[1. Application test scenarios with postman Spring 38](#_qs7s99s8frjj)

[Conclusion 43](#_p1sesybj2w2o)

III. Data Mining 44

[1.1. Subject 44](#_domsslrnpw5z)

[1.2. Objectives 44](#_xvir7l)

[1.3. Suggested solution 44](#_3hv69ve)

[1.4. Data mining Objectives 44](#_rrcvlnlndvzb)

[2. Methodology of work 45](#_h6fnolow89fg)

[2.1. CRISP-DM 45](#_v23r1m5nnn6y)

[2.1.1. Definition 45](#_4h042r0)

[2.1.2. Steps of CRISP-DM plan 45](#_2w5ecyt)

[3. Data Analysis 46](#_1baon6m)

[3.1. Data Collection 46](#_uznf4ps0th36)

[3.1.1. Data Collection Methods 46](#_sxbd03asty8b)

[3.1.2. Data 47](#_b4oj8s2pqnl8)

[3.1.3. Data Understanding 47](#_3vac5uf)

[4. Data Preparation: 47](#_u4pm9wmj2mpn)

[4.1. Visualizations: 47](#_37qnid8vyzke)

Figure 38 : Payment Methods are most used 48

[4.2. NaN and duplicate values: 48](#_1dmfl9jtunvb)

[4.3. Feature Engineering: 49](#_ridskrc36fj4)

[4.4. Encoding: 49](#_d6ao1326o71h)

[4.5. Scaling: 50](#_avut8gbgc93d)

[5. Modeling: 50](#_bj2svsfxvucy)

[5.1. Hyperparameters Tuning: 51](#_ajaoq0y3ozso)

[5.2. Prediction models: 51](#_ulzl9ebc95qe)

[5.3 Classification models: 52](#_dijnm1fxk9k9)

[6. Evaluation: 53](#_dj1j1v4lts5o)

[Conclusion 55](#_kkqts9b0xhzz)

[IV. Front office web 56](#_vdn6tgcv5ab1)

[Introduction : 56](#_vnmc3eiosrex)

[1. Scenario Test 56](#_kuaynlyz1u0h)

[Conclusion 60](#_a0rvmoklkefq)

List of figures

[Figure 1:Logo SSD Team 7](#_Toc134656676)

[Figure 2:Services of SSD Team 8](#_Toc134656677)

[Figure 3:Logo Hydatis 8](#_Toc134656678)

[Figure 4:Logo Adis 9](#_Toc134656679)

[Figure 5 : web site Adis 10](#_Toc134656680)

[Figure 6: logo Achat centrale 10](file:///C:\Users\Admin\Downloads\softwareChasers-1%20(2)%20(1).docx#_Toc134656681)

[Figure 7:website Achat Centrale 11](#_Toc134656682)

[Figure 8: logo Qantis 11](#_Toc134656683)

[Figure 9: website Qantis 12](#_Toc134656684)

[Figure 10: Decent work and Economic growth 13](#_Toc134656685)

[Figure 11: Industry, innovation and infrastructure 14](#_Toc134656686)

[Figure 12: Responsible consumption and production 14](#_Toc134656687)

[Figure 13: Actors 16](#_Toc134656688)

[Figure 14: Use case Diagram global 18](#_Toc134656689)

[Figure 15: Detailed use case of module “User Management” 19](#_Toc134656690)

[Figure 16: Detailed use case of module “User Management” 20](#_Toc134656691)

[Figure 17: Detailed use case of module “User Management” 21](#_Toc134656692)

[Figure 18: Detailed use case of module “User Management” 21](#_Toc134656693)

[Figure 19: Detailed use case of module “Location and complaints Management” 22](#_Toc134656694)

[Figure 20: Physical Architecture 25](#_Toc134656695)

[Figure 21: Logical Architecture 26](#_Toc134656696)

[Figure 22 : Microservice Architecture 27](#_Toc134656697)

[Figure 23:Monolithic architecture vs microservice architecture 27](#_Toc134656698)

[Figure 24 : Eureka server 29](#_Toc134656699)

[Figure 25: Spring Cloud 31](#_Toc134656700)

[Figure 26 : Analysis class diagram global 31](#_Toc134656701)

[Figure 27: Object sequence diagram Add Offre And generate PDF 32](#_Toc134656702)

[Figure 28: Add user and authentication model 33](#_Toc134656703)

[Figure 29: Payment management model 34](#_Toc134656704)

[Figure 30: Tender Management model 34](#_Toc134656705)

[Figure 31: Spring Framework 36](#_Toc134656706)

[Figure 32: Angular Framework 36](#_Toc134656707)

[Figure 33: Python 37](#_Toc134656708)

[Figure 34: MySql 37](#_Toc134656709)

[Figure 35: IntelIj 37](#_Toc134656710)

[Figure 36: WebStorm software 38](#_Toc134656711)

[Figure 37: XAMPP 38](#_Toc134656712)

[Figure 38 : Anaconda 38](#_Toc134656713)

[Figure 39: Postman 39](#_Toc134656714)

[Figure 40: GitHub 39](#_Toc134656715)

[Figure 41: StarUML 39](#_Toc134656716)

[Figure 42: Adding a New User with Role and Privilege in Postman 40](#_Toc134656717)

[Figure 43: Verify email 41](#_Toc134656718)

[Figure 44: Get all Comment 41](#_Toc134656719)

[Figure 45: Adding purchase request 42](#_Toc134656720)

[Figure 46: Payement and mailing of payment 43](#_Toc134656721)

[Figure 47: Add Invoice 44](#_Toc134656722)

[Figure 48: CRISP-DM methodology 46](#_Toc134656723)

[Figure 49: Study of the distribution of selling price values 48](#_Toc134656724)

[Figure 50: Features correlation 49](#_Toc134656725)

[Figure 51: Payment Methods are most used 49](#_Toc134656726)

[Figure 52 : Exploring non null values 50](#_Toc134656727)

[Figure 53: Extracting month and year from date 50](#_Toc134656728)

[Figure 54: Payment Methods 51](#_Toc134656729)

[Figure 55 : Encoding 51](#_Toc134656730)

[Figure 56: Distribution of Selling Price 51](#_Toc134656731)

[Figure 57: Best parameter 52](#_Toc134656732)

[Figure 58: Run and fit the model 52](#_Toc134656733)

[Figure 59: Predict values for the test split 52](#_Toc134656734)

[Figure 60: Create model ARIMA 53](#_Toc134656735)

[Figure 61: Test data with prediction 53](#_Toc134656736)

[Figure 62: Create model Knn 54](#_Toc134656737)

[Figure 63: Test Model Knn 54](#_Toc134656738)

[Figure 64: Create Model SVM 54](#_Toc134656739)

[Figure 65: correlation circle 55](#_Toc134656740)

[Figure 66: Access the model using R-squared score 55](file:///C:\Users\Admin\Downloads\softwareChasers-1%20(2)%20(1).docx#_Toc134656741)

[Figure 67: Test Error 55](#_Toc134656742)

[Figure 68: List of purchase request 57](#_Toc134656743)

[Figure 69: purchase request addition 57](#_Toc134656744)

[Figure 70: user addition interface 58](#_Toc134656745)

[Figure 71: Interface login user 59](#_Toc134656746)

[Figure 72: order management and online payment interface 60](#_Toc134656747)

[Figure 73: invoice addition interface 61](#_Toc134656748)

**Chapter I: Project Background**

# Introduction

In this chapter we will give more context to the development of our project, firstly by presenting the company, we will present and analyze similar applications, the problematic of the latter for which we create the solution, The planning phase is essential to ensure the proper conduct of the subsequent stages of the development of an IT solution. Indeed, it is by understanding the company's expectations that we can best determine the needs to be addressed. We will identify the actors of our application and list the needs to be taken into account.

# Presentation of the company



Figure :Logo SSD Team

## SSD Team:

SSD (Sustainable Social Development) Team: are an RDI team at Esprit-Tech made up of research professors at ESPRIT. Their goal is to improve the well-being of everyone in society and provide proper use of technology to solve several social problems.

* Founded in 2022
* Number of employees: 2-10
* Specializations: Artificial Intelligence, Software Development, Serious Game, EHealth et RDI
* Website: <https://ssdteam-rdi.github.io/>
* Email:[ssd.team@esprit.tn](mailto:ssd.team@esprit.tn)
* Location: Z.I Chotrana II - B.P. 160 Technological Pole 2083 Cite El Ghazala Raoued, Ariana, Tunis

### Services

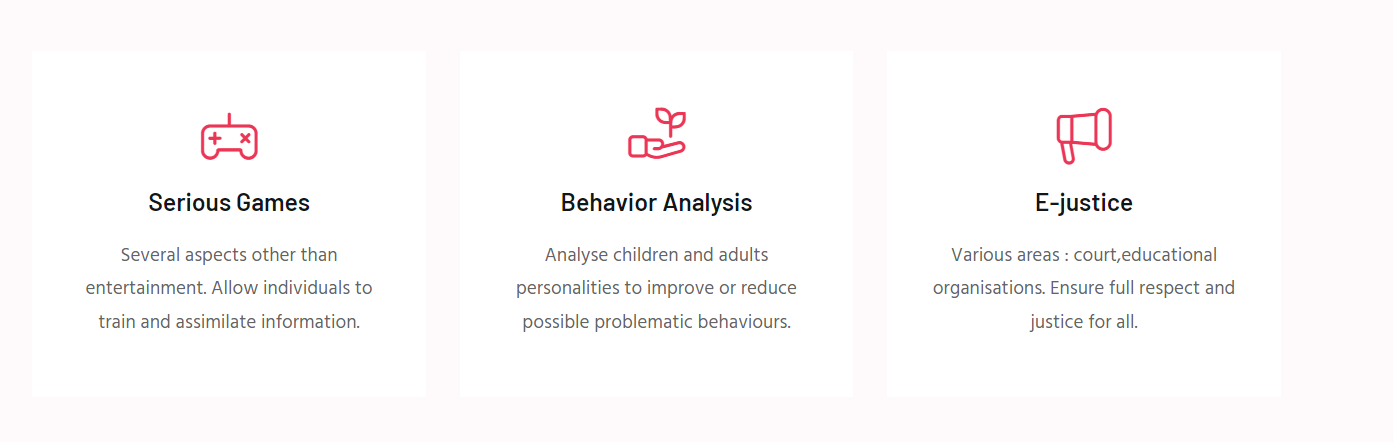


Figure :Services of SSD Team

## Hydatis:



Figure :Logo Hydatis

**Hydatis** is a Tunisian leading provider of digital outsourcing, services center and technology consulting for world wide companies' undergoing digital transformation. We help businesses solve the problem of local talent shortage. Build a cross-functional team with custom hired software engineers and benefit from our value-added services. We source, vet & deploy the top technology professionals Tunisia has to offer. Global companies trust Hydatis to make their hiring faster, easier & cost-effective.

* Founded in 2020
* Private society
* Number of employees : 11-50
* Different domains : outsourcing, offshore, nearshore, Digital, It and Data intelligence
* WebSite : https://www.hydatis.com/

# Study of the existing:

Clients can already insure themselves using a variety of applications. As a result, prior to commencing on the fulfillment of our project, a thorough examination of existing solutions is crucial. With that in mind, we've put together this study of existing solutions and their flaws.

## Tunisian central purchasing office:

There are many purchasing centres, we find Tunisian ones as well as national ones

### Adis Tunisia

Is a central purchasing office dedicated to professionals in the restaurant and hotel sector. As a food wholesaler.



Figure :Logo Adis

* **Advantages:**

-The platform is simple and easy to use.

-The existing products are detailed

* **Disadvantages:**

-It does not cover all types of products, it limited to food only

-The platform isn't dynamic

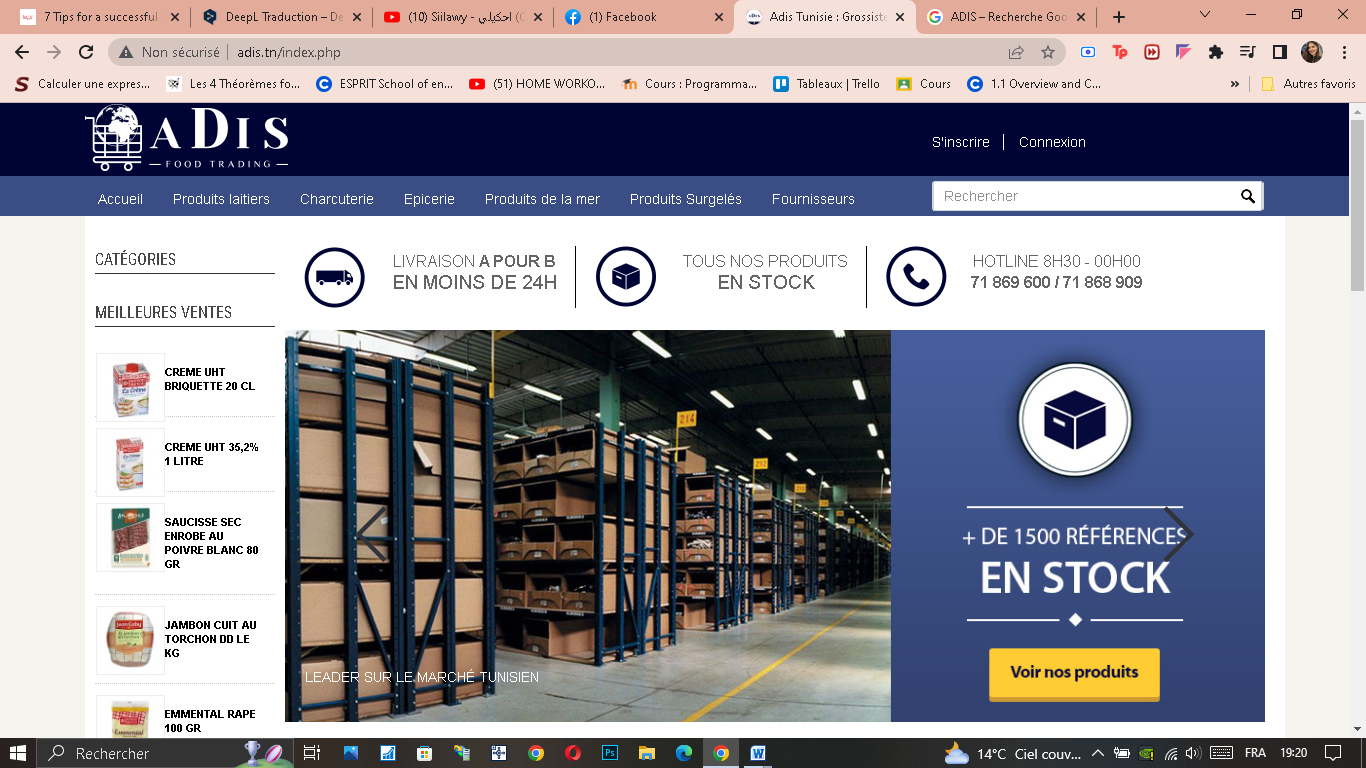


Figure : web site Adis

## 2.International central purchasing office:

### 2.1 Achat CentraleLa centrale d'achats des entreprises - Achat Centrale

Figure : logo Achat centrale

**Figure :Logo Achat Centrale**

Since 2011, Achat Centrale has helped French companies refocus on their core business. Without sufficient purchasing volume, SMEs pay more than large groups. By grouping thousands of companies, Achat Centrale provides to its members the same prices and services as those reserved for large companies.

Achat Centrale's main purpose is to offer companies advantageous commercial conditions thanks to the purchasing volume of thousands of member companies

* **Advantages:**

-Platform is well organized

-All categories of products are available

- The platform is dedicated to all types of companies

* **Disadvantages:**

- It is a monolingual stage, they just deal with French as the principal language.



Figure :website Achat Centrale

### 2.2 Qantis:



Figure : logo Qantis

Created in 2001 on the initiative of independent SME and ETI managers, under the name of PME Centrale, our purchasing center enables you to benefit from advantageous purchasing conditions for your products or services, by pooling your needs and negotiating framework agreements with providers.

* **Advantages:**
* Multi-varied and dynamic platform

- Diversification of product types and providers

- The platform is dedicated to large, medium and mid-sized companies

* **Disadvantages:**
* Integration: Integrating Qantis with existing systems and processes can be challenging, and may require additional time and effort.
* Cost: Qantis can be expensive, and the costs associated with the software, including licensing, implementation, and support, can be significant.
* 

Figure : website Qantis

# Criticism of the existing:

The process of criticizing the existing situation is very important for the creation of the project, which ensures its credibility and persistence, and whose positive and negative points are criticized.

* **Positive angles :**

- 100 percent Free Candidates Offer.

- They give straightforward connection points of their substance and alluring by their plans.

- They offer organizations administrations to advance their enrollment and the administration of inside abilities.

* **Negative angles:**

There were holes in the capacities presented by the current application and a few issues:

- It is a monolingual stage; they just deal with French as the principal language.

- Length of enrollment processes are excessively long.

- Long and drawn-out demand handling time.

- Absence of appeal of the client for the applicants.

- Intricate and ugly enlistment apparatuses.

- Absence of perceivability on all current, dropped and finished enrollments

## Problematic:

Today there are various central purchasing applications all throughout the world.

Despite this, they only cover a small portion of the overall need, the process is hard to reach and even they face many problems related to integration challenges with existing systems which limit their offer.

Central purchasing applications, while widespread, still have several limitations such as lack of customization, high cost, resistance to change, data security concerns, complexity, and limited vendor options. These challenges can limit the effectiveness and accessibility of central purchasing processes, making it difficult for organizations to fully realize the benefits of these systems.

## Proposed Solution:

We propose our solution which incorporates new and advanced features that address the issues with existing solutions by improving user experience, enhancing more security tools, automating the overflow and specially providing a streamlined integration process which reduces times and efforts.

# Sustainable Development

Sustainable development is a concept that aims to meet the needs of the present without compromising the ability of future generations to meet their own needs.

There are several Sustainable Development Goals (SDGs) that our central purchasing app can help to achieve. Here are a few examples:

## Decent work and Economic growth

**Icon

Description automatically generated**

Figure : Decent work and Economic growth

Our central purchasing app can help businesses to streamline their procurement operations, saving time and resources. This can lead to increased productivity and economic growth, creating more job opportunities and promoting decent work.

## 2. Industry, innovation, and infrastructure

****

Figure : Industry, innovation and infrastructure

The app's ability to integrate with other business applications and leverage the latest in web development technology promotes innovation and improves infrastructure in the procurement industry.

## 3. Responsible consumption and production

****

Figure : Responsible consumption and production

Our central purchasing app can encourage responsible consumption and production by providing businesses with access to multiple suppliers in one place, enabling them to compare prices and make more informed purchasing decisions. This can help to reduce waste and promote sustainable production practices.

Overall, our central purchasing app can contribute to several SDGs by promoting economic growth, innovation, responsible consumption and production, and climate action.

# Conclusion:

In this chapter, we have presented the important steps in the realization of this application. We have also started some sustainable development that we can integrate into our project. The next chapter will be devoted to the design of the application.

**Chapter II: Need Analysis**

# Introduction

In this chapter we will present the functional requirements as well as the non-functional needs. We will also better demonstrate the functionalities of our solution by showing the use case diagram and the class diagram.

We'll move on to this chapter once our application has been well-detailed. The first section of this chapter will be devoted to a discussion of the programming tools that we employed during the application's development. After that, we'll go over the servers and database management systems that are used to run our tools. We will also demonstrate the tests of our application using screens caps from postman.

# Requirements and specifications

This part contains the essential requirements of our project and includes different sub-parts.

## Actors:

****

Figure : Actors

* **Administrator**:

The administrator who ensures the proper functioning of the platform as well as the management of users and the parameters of the application.

* **Buyer**:

The buyer can initiate their own purchase request and wait for possible offers to order and pay for the requested items.

* **Operator**:

The operator acts as an intermediary between the supplier and buyer, working closely with suppliers to ensure product availability, timely delivery and choose the suitable quotation for the purchase requisition while also helping buyers find desired products and resolving any issues related to their purchase.

* **Supplier**:

The supplier responsible for responding to tenders by providing quotes and managing the delivery and invoicing process.

## Functional requirements:

* **User Management:**

The administrator can manage users as well as these roles, permissions also authentication and forget password.

* **Location Management:**

The administrator can manage user locations, shipping address and department address.

* **Management of purchase requests:**

The buyer can create a purchase request and specify these requested items as well as their natures and units.

* **Tender Management:**

The operator can create a tender and manage the quotes provided by the suppliers to launch an offer.

* **Order Management:**

The buyer can pay an order invoiced by the suppliers with these taxes and these financial charges.

* **Delivery Management:**

The supplier can manage the return slips and invoices associated with a delivery received.

## Global Use Case Diagram:



Figure : Use case Diagram global

## 4. Detailed Use Case diagrams of each module

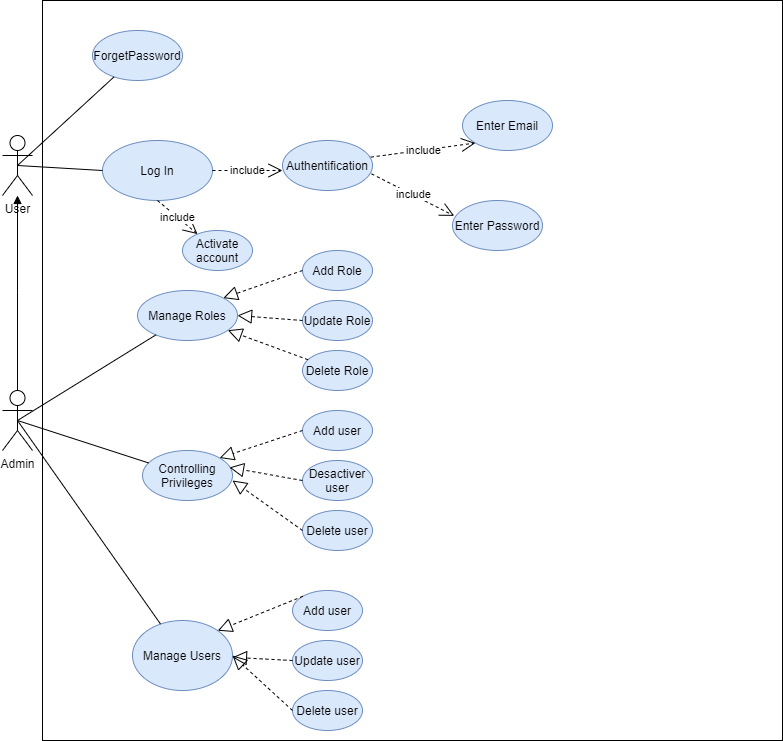
****

Figure : Detailed use case of module “User Management”

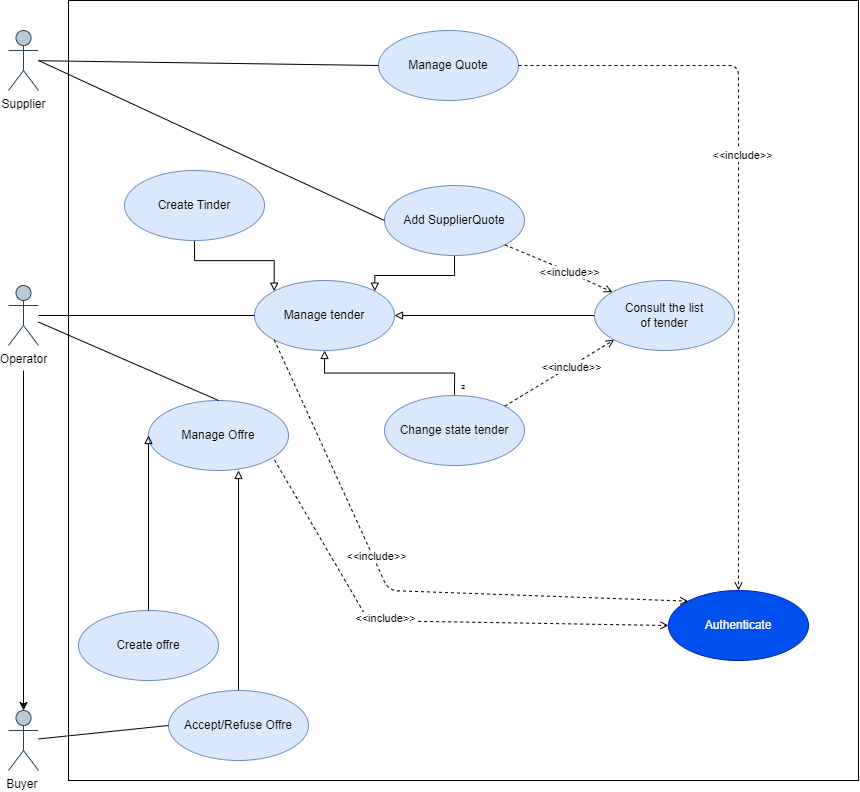


Figure : Detailed use case of module “User Management”

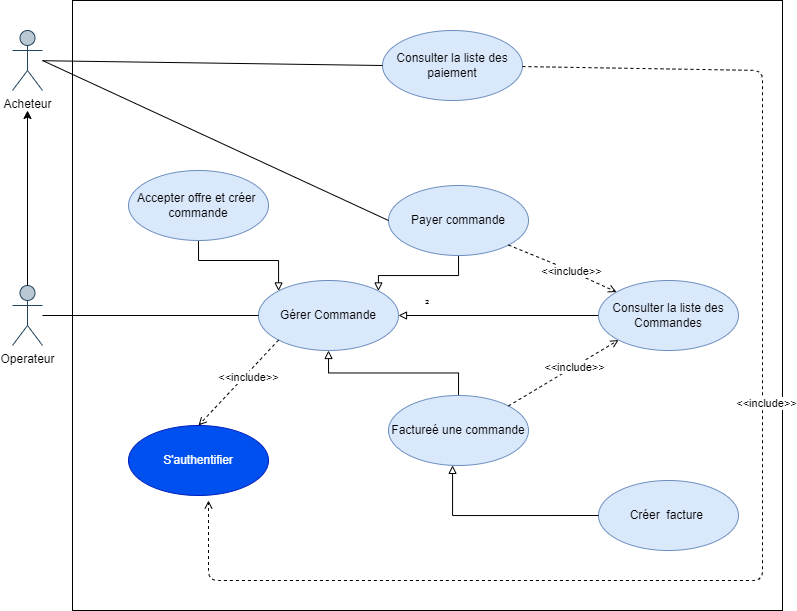


Figure : Detailed use case of module “User Management”

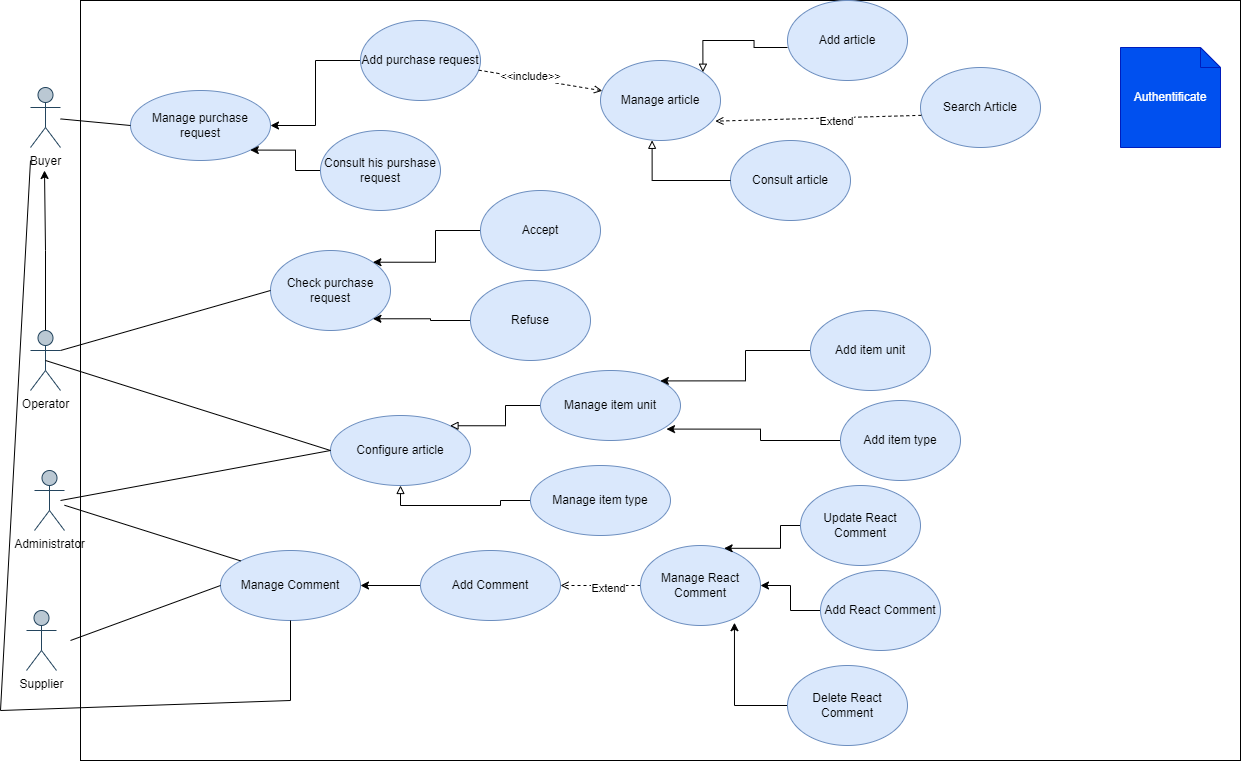


Figure : Detailed use case of module “User Management”

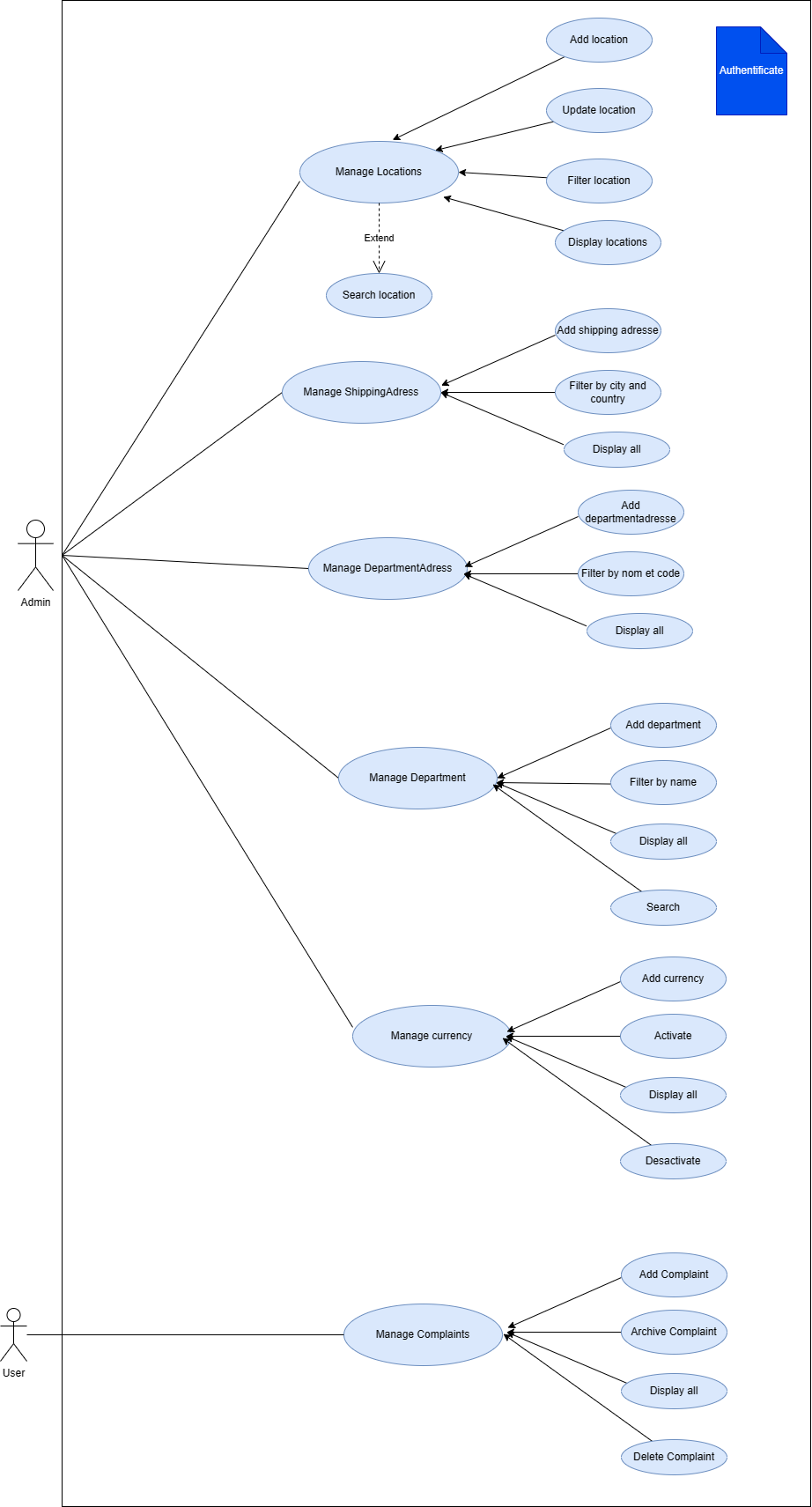


Figure : Detailed use case of module “Location and complaints Management”

## 5. Non-functional requirements:

* **Security (OAuth2 et Spring Security, BCrypt ):**

User authentication configuration: Spring Boot provides simple integration with authentication technologies such as OAuth2 and Spring Security.

You can configure user authentication using an external identity provider or by storing the identification information in your own database.

* **Role and authorization management:**

Spring Security allows for defining user roles and authorizations for different parts of the application.

You can also use annotations to specify the required authorization to access certain parts of the application.

* **Encryption of sensitive data:**

Spring Boot supports encryption of sensitive data such as passwords and credit card information. You can use libraries such as BCrypt to securely encrypt this data.

* **Performance (Spring Boot Actuator et Micrometer):**

We can also measure the performance of other components of our application using appropriate annotations and expose them using the Actuator endpoint using the Spring Boot Actuator and Micrometer dependencies.

* **Quality (Sonar):**

SONAR is an open-source code quality tool used to analyze and measure the quality of the source code of an application.

It allows developers to find and fix code quality issues, such as bugs, security vulnerabilities, coding convention violations, etc., during development.

* **Maintainability (open/close):**

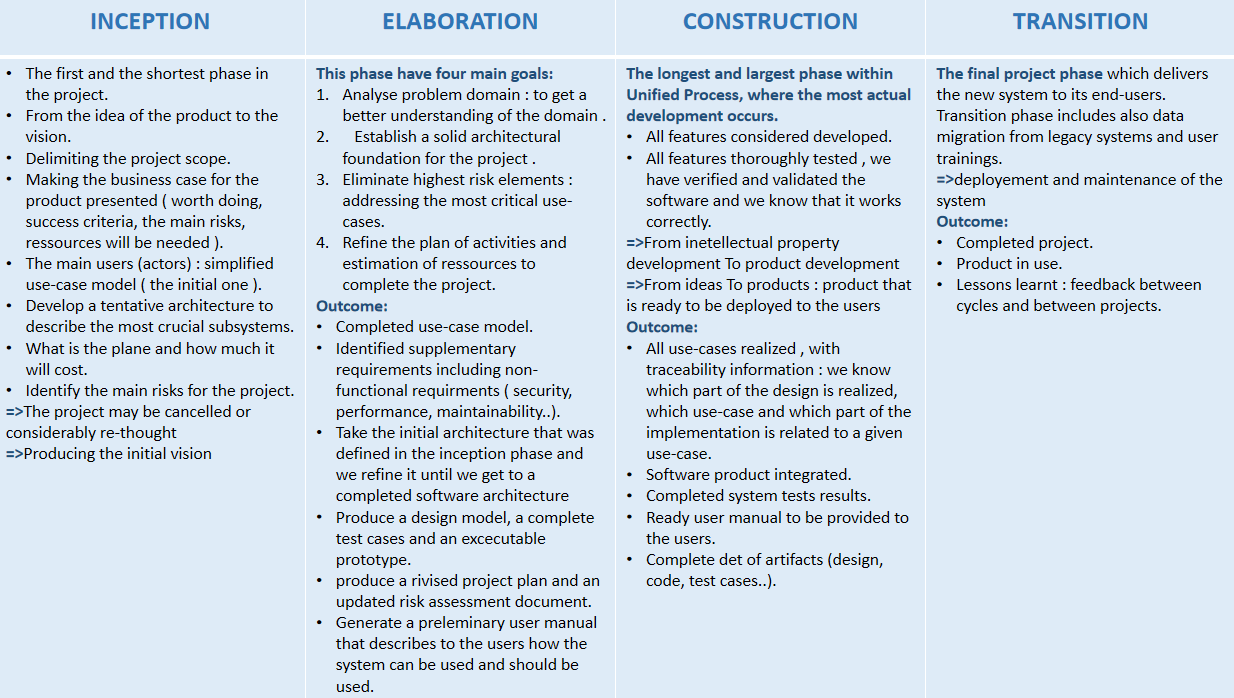
The Open/Close Principle is one of the SOLID principles, which are considered best practices for developing robust and maintainable software.

This principle states that classes should be designed in such a way that they can be extended to add new features without altering their current functioning.

# II- Methodology Adopted:

## The Unified Process:

Is an architecture centric, use case driven, iterative and incremental development process from which a customized process can be defined. It recognizes the importance of customer communication and streamlined methods for describing the customer’s view of a system. This process divides the development process into four phases:



**Figure :Methodology unified process**

We chose the Unified Process as our adopted methodology because it: Allows us to deal with

Changing requirements of the project thanks to the four phases. Forces integration to occur throughout development, specifically within the construction phase where all other coding and development is taking place.

## Physical Architecture:

The physical architecture of our web application is depicted in the diagram below. It shows the interaction between layers; in fact, the Framework SpringBoot connects to our database via TCP, and all views are displayed using Postman first, then a navigator.

# Diagram Description automatically generated

Figure : Physical Architecture

The client sends a request via the URL to the angular server (node) and which will send an HTTP request to the spring boot server (apache) which will query the database by sending a request and it will receive the required data and that it will return them to the node server in the form of JSON and the node server will return a web page to the browser to display it to the client.

## Logical Architecture:

The figure below shows the Logical architecture of our web application.

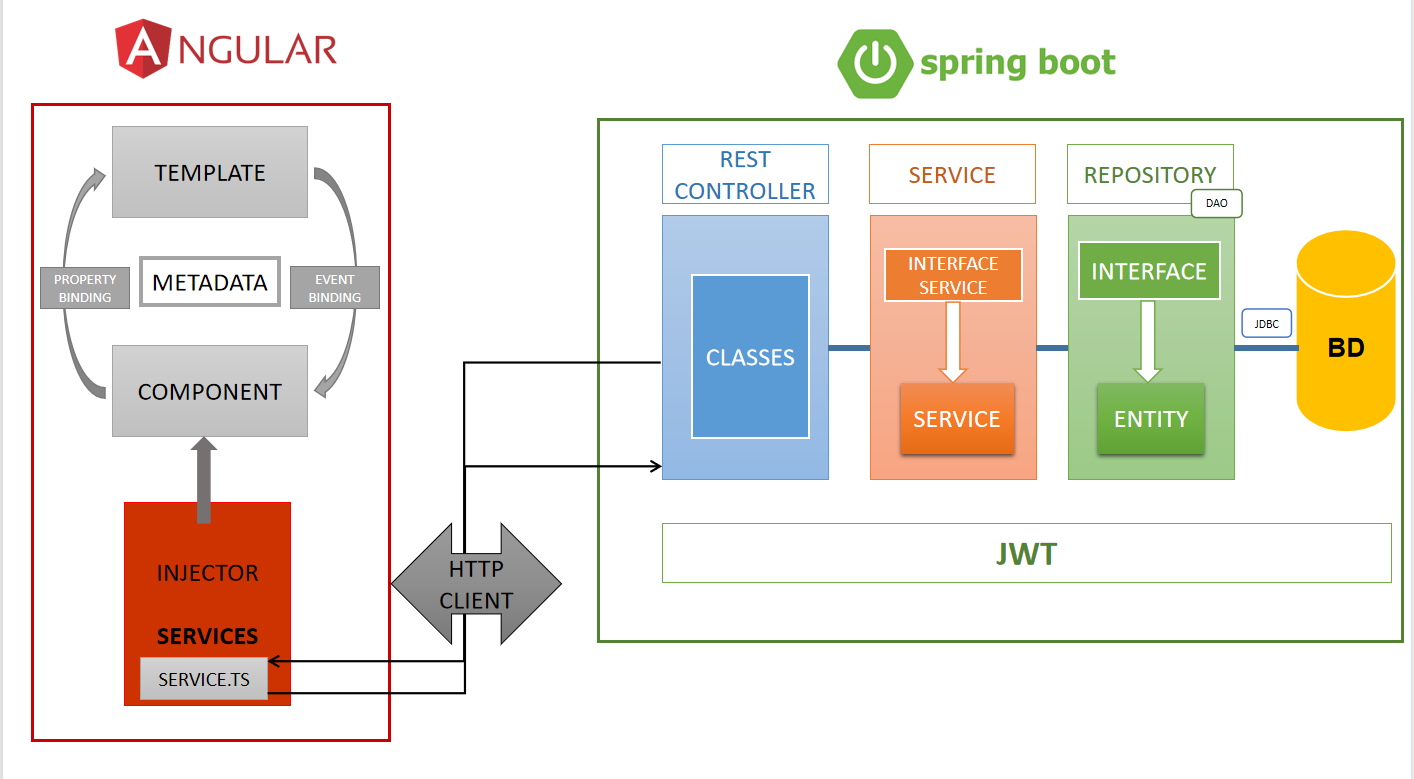


Figure : Logical Architecture

The view interacts with the component which uses the service methods and which in turn interacts with the Spring Boot controller to consume the api. Spring boot rest controller interacts with the service which interacts with the repository (Spring JPA) that is related to the mysql database.

## Microservice Architecture

Microservice architecture is an approach to software development where a large application is broken down into smaller, independent services that work together to deliver the required functionality. Each microservice is designed to perform a specific business function, has its own database and can be deployed and scaled independently.

In microservice architecture, each microservice communicates with other microservices over a network using lightweight protocols such as REST or messaging. Each microservice can be developed using a different technology stack and programming language, as long as they can communicate with each other.

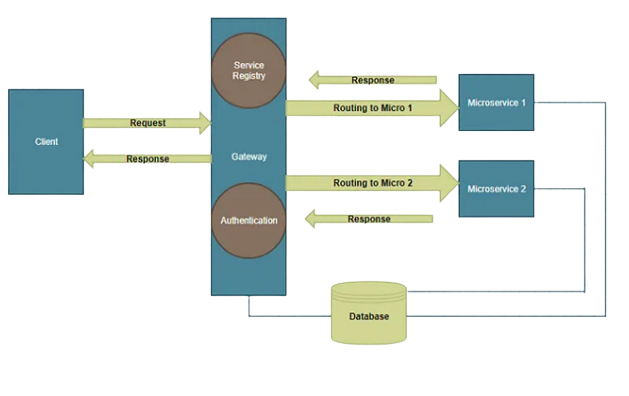
[](#_2et92p0)

Figure : Microservice Architecture

* **Monolithic architecture vs microservice architecture**

The monolithic architecture pattern has been the architectural style used in the past, pre-Kubernetes and cloud services days.

In a monolithic architecture, the software is a single application distributed on a CD-ROM, released once a year with the newest updates. Examples are Photoshop CS6 or Microsoft 2008.

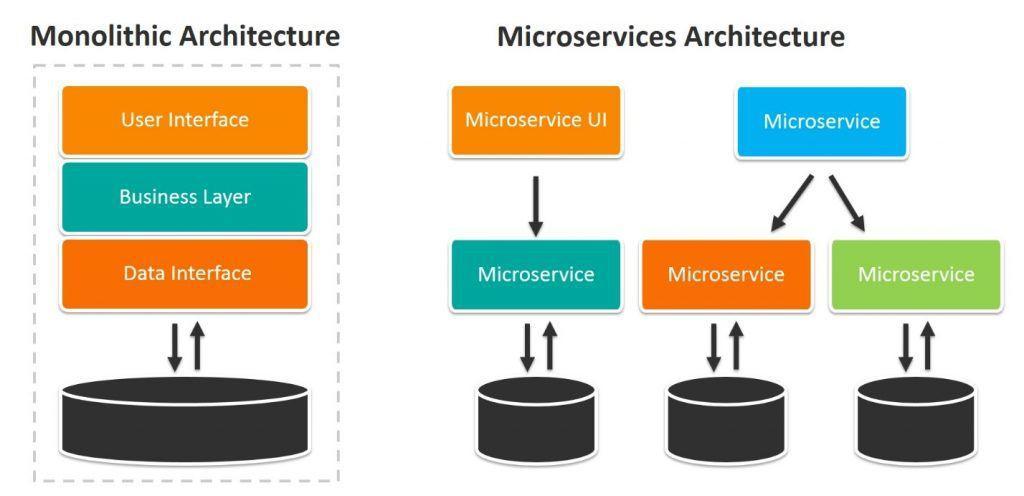


Figure :Monolithic architecture vs microservice architecture

Figure 23

That style was the standard way of building software. But as tech has evolved, so too the architectural style must advance. In an age of Kubernetes, and CI/CD workflows, the monolithic architecture encounters many limitations—companies need to push to microservices. Characteristics of a monolithic architecture:

* Changes are slow
* Changes are costly
* Hard to adapt to a specific, or changing, product line

Monolithic structures make changes to the application extremely slow. Modifying just a small section of code can require a completely rebuilt and deployed version of software. If we wish to scale certain functions of an application, we must scale the entire application, further complicating changes and updates. Microservices help to solve these challenges.

* **Eureka Server :**

Eureka Server is a component of Netflix OSS (Open Source Software) that is used for implementing service registration and discovery. Eureka Server is a service registry that allows microservices to register themselves and discover other microservices that are registered with it.

When a microservice is launched, it registers itself with the Eureka Server by sending a heartbeat signal. This signal includes information about the microservice such as its hostname, port, and other metadata. The Eureka Server stores this information in its registry and makes it available to other microservices.

Microservices can then use the Eureka Server to discover other microservices by querying its registry. This allows microservices to communicate with each other without having to know the exact location of the other microservice.

In addition to service registration and discovery, Eureka Server also provides features such as load balancing and failover. With load balancing, requests can be distributed across multiple instances of a microservice to improve performance. Failover ensures that if one instance of a microservice goes down, requests are automatically rerouted to another instance.

Overall, Eureka Server plays a crucial role in Spring microservice architecture by providing a central point for service registration and discovery, load balancing, and failover.

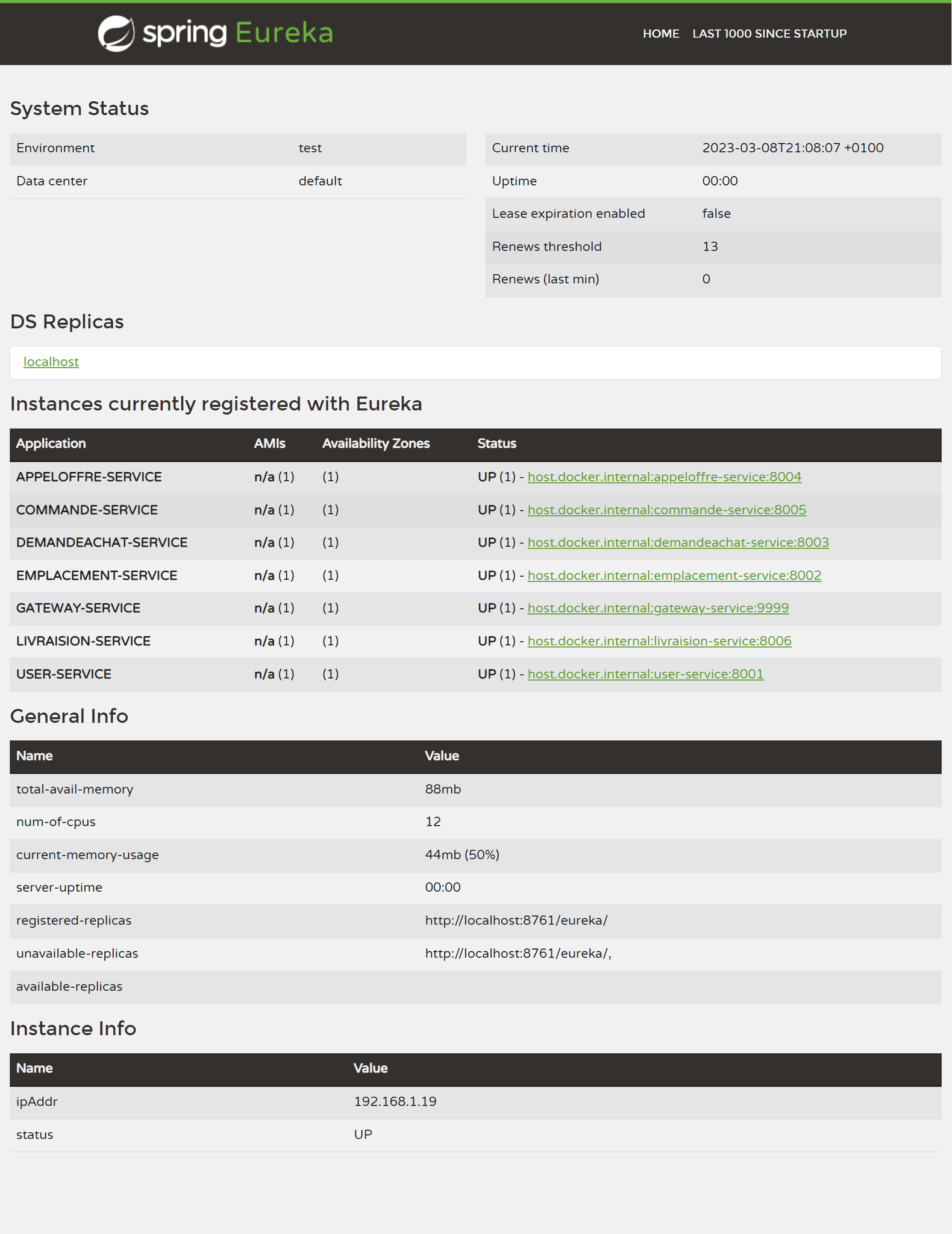


Figure : Eureka server

* **Gateway Server :**

a gateway server is a central entry point that serves as an interface between the clients and the microservices. The gateway server acts as a reverse proxy that routes requests from the clients to the appropriate microservices based on the request URL or other information in the request.

The gateway server performs several important functions in a microservice architecture. These include:

1. Service discovery: The gateway server maintains a registry of all available microservices and their endpoints, allowing it to route requests to the correct service.
2. Load balancing: The gateway server can distribute incoming requests to multiple instances of the same microservice, providing scalability and fault tolerance.
3. Authentication and authorization: The gateway server can authenticate users and authorize access to specific microservices based on user roles or other criteria.
4. Routing and transformation: The gateway server can route requests to different microservices based on the request URL or other information, and it can also transform or filter the request and response messages as needed.

By providing a single entry point for clients to access multiple microservices, the gateway server simplifies the architecture and makes it easier to manage and scale microservices.

* **Advantages to Microservices**

Applications built as a set of independent, modular components are easier to test, maintain, and understand. They enable organizations to:

* Increase agility
* Improve workflows
* Decrease the amount of time it takes to improve production

While each independent component increases complexity, the component can also have added monitoring capabilities to combat it.

* **Spring Cloud**

Spring Cloud is a framework for building robust cloud applications. The framework facilitates the development of applications by providing solutions to many of the common problems faced when moving to a distributed environment.

Applications that run with microservices architecture aim to simplify development, deployment, and maintenance. The decomposed nature of the application allows developers to focus on one problem at a time. Improvements can be introduced without impacting other parts of a system.

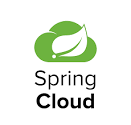
****

Figure : Spring Cloud

## 5. Class Diagram:

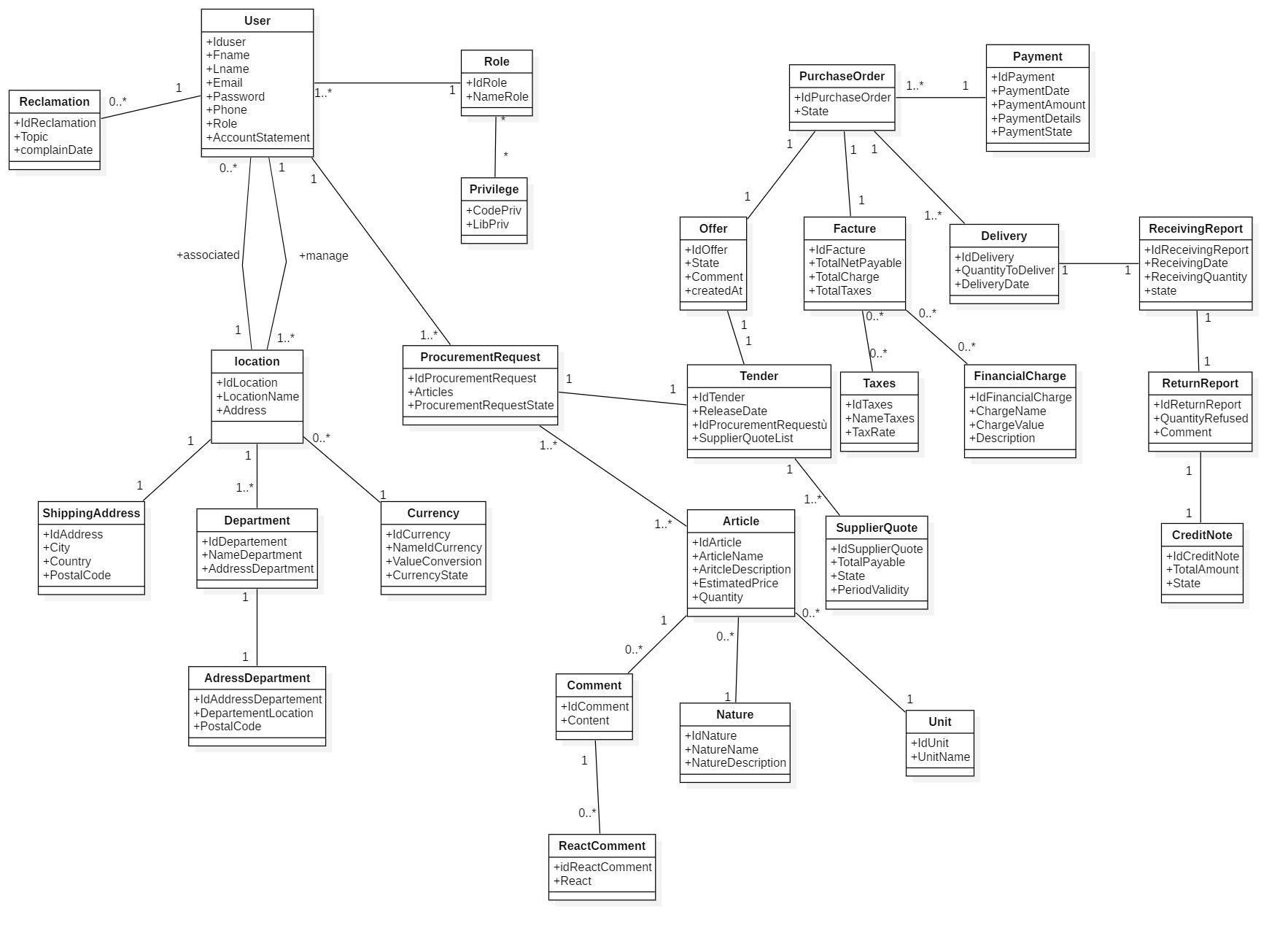
****

Figure : Analysis class diagram global

## 6. Object sequence diagram

Once the user has entered all the required information, the system creates a new "Offer" object containing the provided details. The system then sends a request to a "Document Generator" object to create a PDF document.

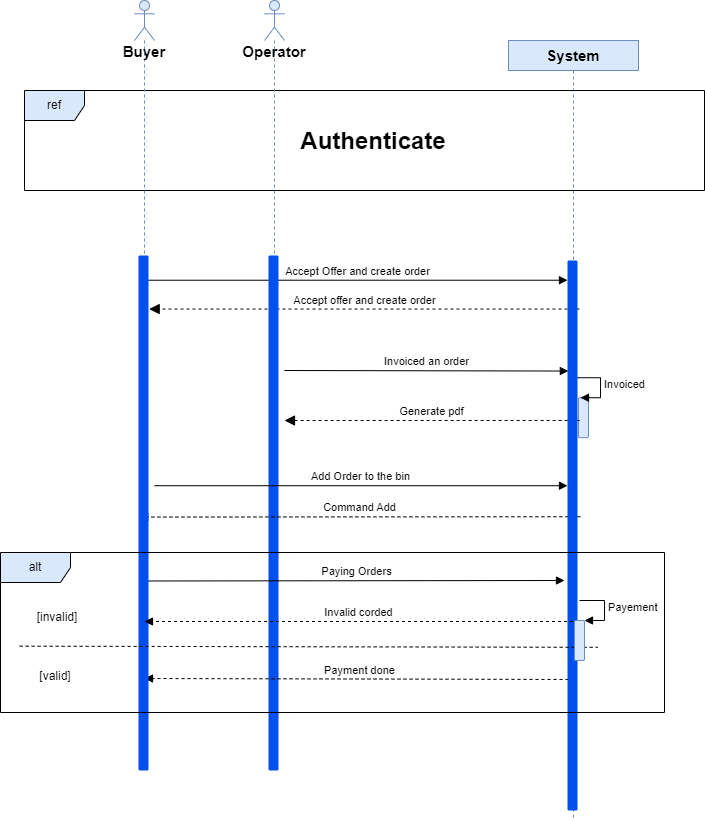
****

Figure : Object sequence diagram Add Offre And generate PDF

## 7. Mock-ups:

In this part we have arranged our work plan. We caught the useful requirements necessities of our application as well as the non-practical requirements, we have indicated the client jobs.

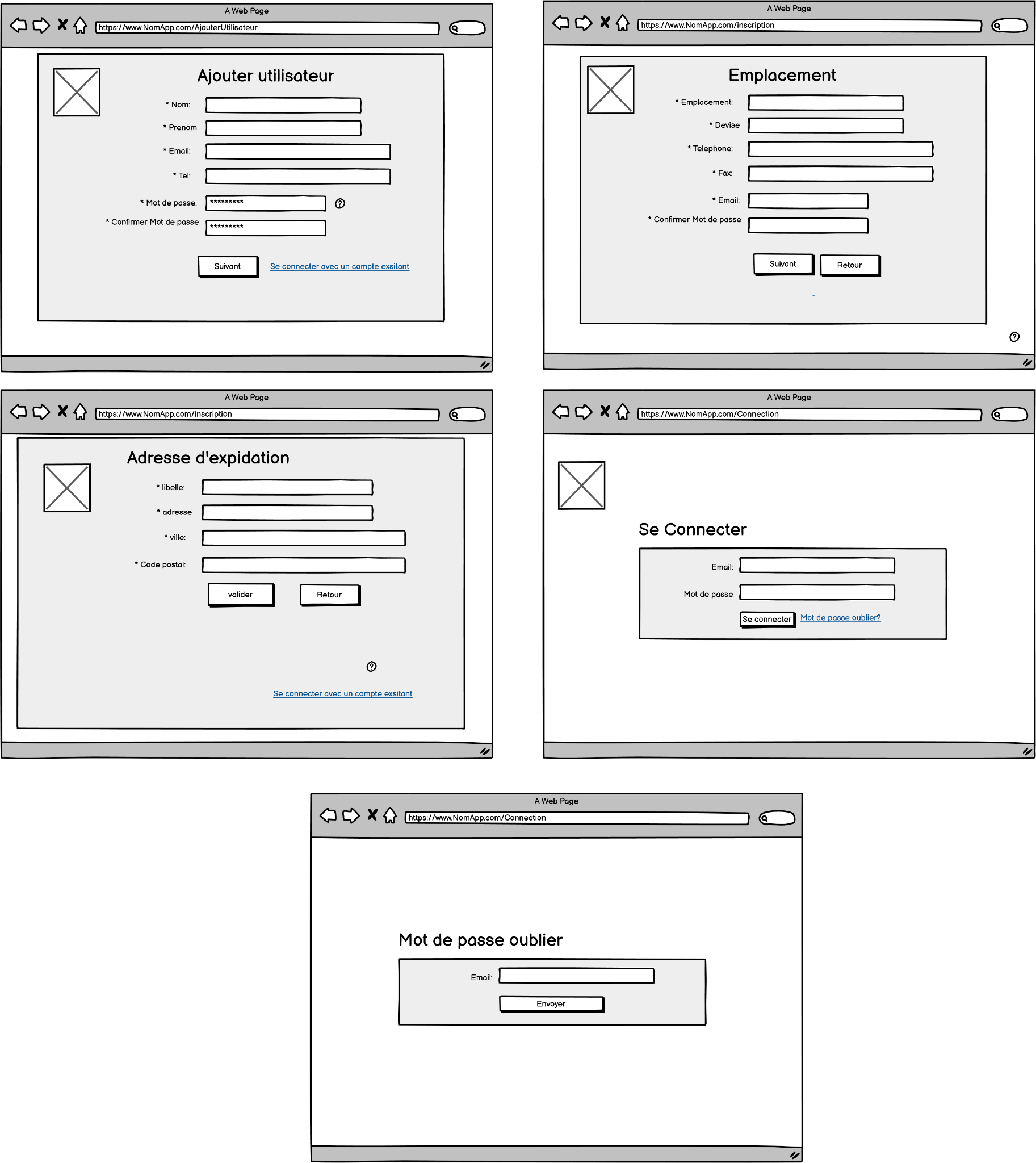


Figure : Add user and authentication model

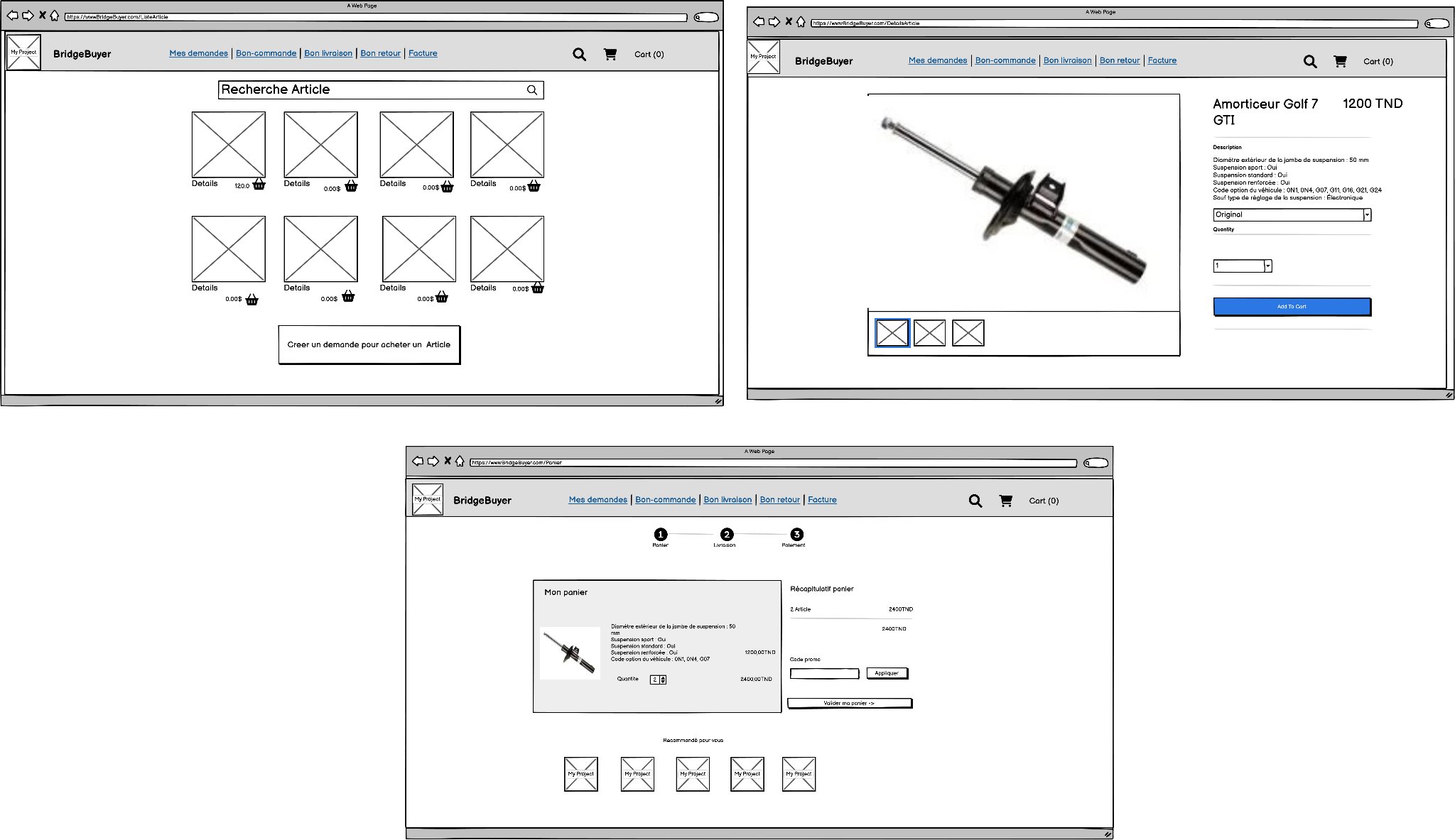


Figure : Payment management model

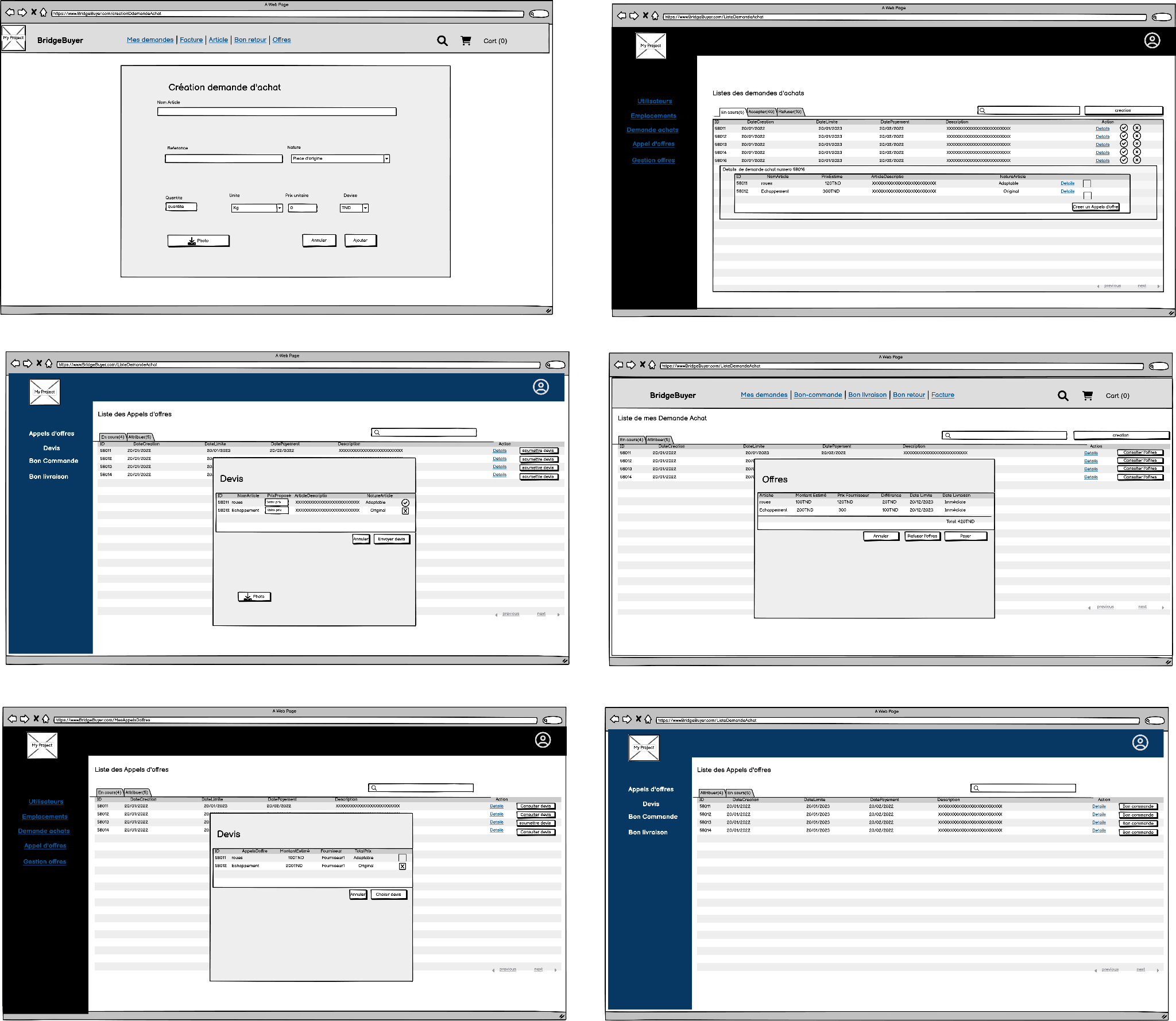


Figure : Tender Management model

# Conclusion

This chapter covers the essential aspects of the project's realization, including identifying project needs, creating design using sequence and class diagrams, modeling application interfaces, and planning development. Additionally, the chapter discusses the project's physical and logical architecture .The next chapter will be devoted to the implementation of our application.

# 

# Chapter III: Implementation

# Working environment

The selection of a software environment is an important aspect of the accomplishment process. We'll go over the software utilized in this section briefly.

●      **Spring**



Figure : Spring Framework

Spring is an open source Java-based framework used to create a micro Service. It is developed by Pivotal Team and is used to build stand-alone and production ready spring applications.

●      **Angular**



Figure : Angular Framework

Angular is a framework for creating client applications on a single page "Single Page Application" using HTML and TypeScript .

 ●      **Python**

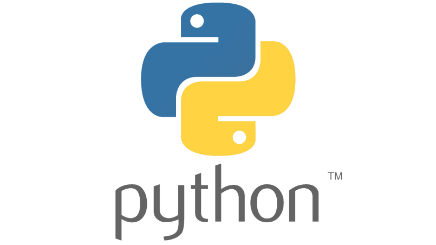


Figure : Python

Python is a high-level, interpreted programming language widely used for a variety of tasks such as web development, scientific computing, data analysis, artificial intelligence, and more.

●    **MySQL**



Figure : MySql

MySQL is an open-source, fast, reliable, and flexible relational database management system, typically used with PHP. This chapter is an introductory chapter about MySQL, what is MySQL, and the main features of MySQL are described here.

●      **Intellij**



Figure : IntelIj

IntelliJ IDEA is an Integrated Development Environment (IDE) for JVM languages designed to maximize developer productivity.

**●      WebStorm**



Figure : WebStorm software

[WebStorm](https://www.jetbrains.com/webstorm/) is an integrated development environment for coding in [JavaScript](https://www.jetbrains.com/help/webstorm/javascript-specific-guidelines.html) and its related technologies, including [TypeScript](https://www.jetbrains.com/help/webstorm/typescript-support.html), [React](https://www.jetbrains.com/help/webstorm/react.html), [Vue](https://www.jetbrains.com/help/webstorm/vue-js.html), [Angular](https://www.jetbrains.com/help/webstorm/angular.html), [Node.js](https://www.jetbrains.com/help/webstorm/developing-node-js-applications.html), [HTML](https://www.jetbrains.com/help/webstorm/editing-html-files.html), and [style sheets](https://www.jetbrains.com/help/webstorm/style-sheets.html). Just like IntelliJ IDEA and other JetBrains IDEs, WebStorm makes your development experience more enjoyable, automating routine work and helping you handle complex tasks with ease

●      **Xampp**



Figure : XAMPP

Using XAMPP is for testing your clients' sites or your own website before uploading it to the remote web server. This XAMPP server software gives you the proper environment to test MYSQL, PHP, Apache and Perl Project on the local computer.

●      **Anaconda**



Figure : Anaconda

Anaconda is a popular open-source distribution of the Python and R programming languages for scientific computing, data science, and machine learning.

* **Postman**

We will use Postman to test the APIs and services used in the application.

Postman simplifies each step of the API lifecycle and streamlines collaboration so we can create better APIs faster.



Figure : Postman

* **GitHub**

GitHub is a web-based version control and collaboration platform for software developers. Microsoft, the biggest single contributor to GitHub, acquired the platform for $7.5 billion in 2018. GitHub, which is delivered through a software as a service ([SaaS](https://www.techtarget.com/searchcloudcomputing/definition/Software-as-a-Service)) business model, was started in 2008. It was founded on Git, an open source [code](https://www.techtarget.com/whatis/definition/code) management system created by Linus Torvalds to make software builds faster. Git is used to [store the source code for a project](https://www.techtarget.com/searchvmware/tip/How-to-use-GitHub-vRA-integration-to-keep-track-of-blueprints) and track the complete history of all changes to that code. It lets developers collaborate on a project more effectively by providing tools for managing possibly conflicting changes from multiple developers.

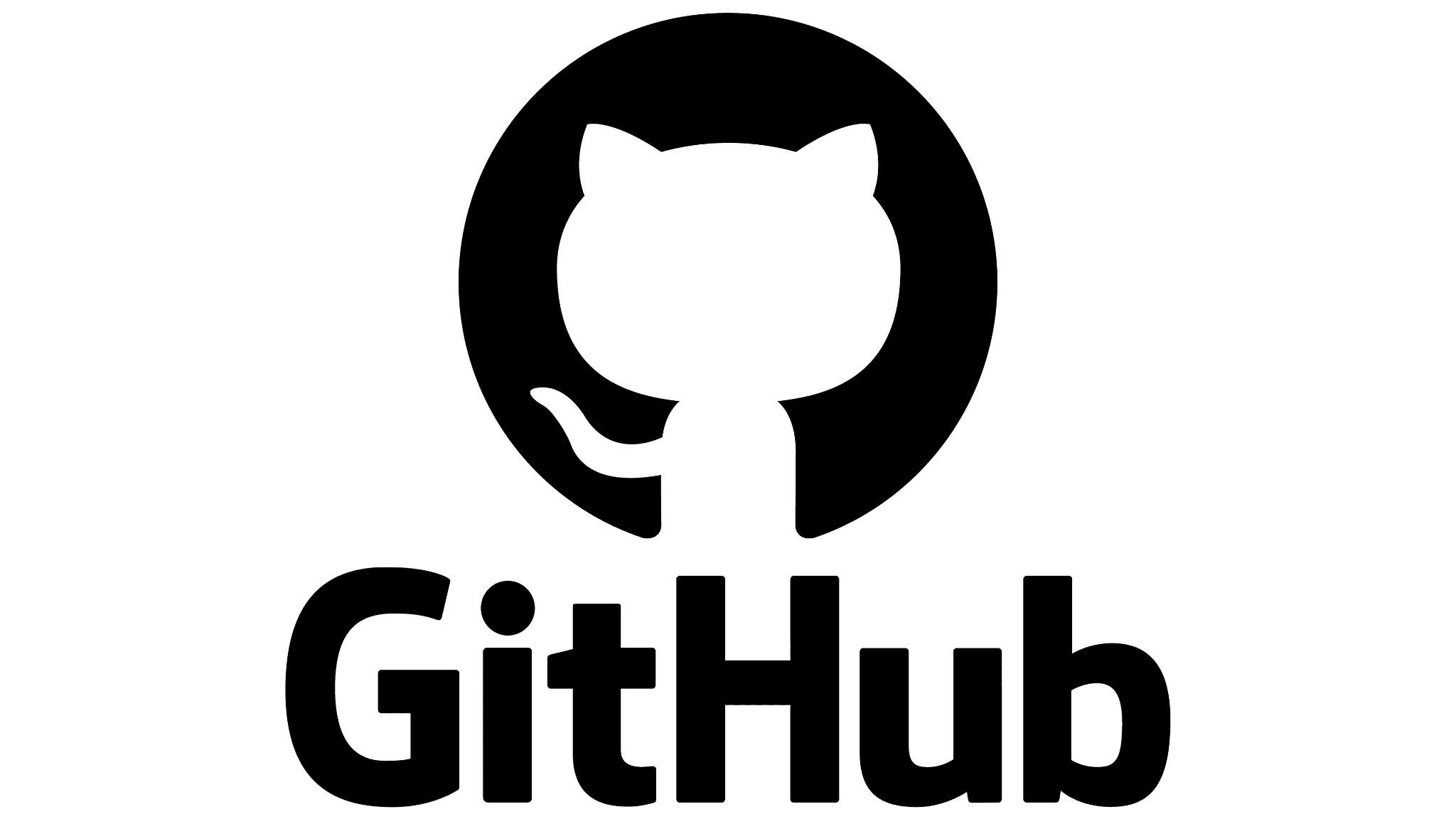


Figure : GitHub

* **StarUML**

To model the different UML diagrams, we will be using StarUML. This software will provide us with:

An easy and fast way to realize the diagrams thanks to a user-friendly interface.



Figure : StarUML

# Spring

## Introduction

Building a central purchasing order system involves two crucial stages: coding and testing. This report provides an overview of these phases, discussing the programming languages and testing techniques used. The report concludes with a summary of findings and lessons learned, offering insights for software developers and project stakeholders.

## Application test scenarios with postman Spring

After register, user should activate his account by verifying his email. he can also reset

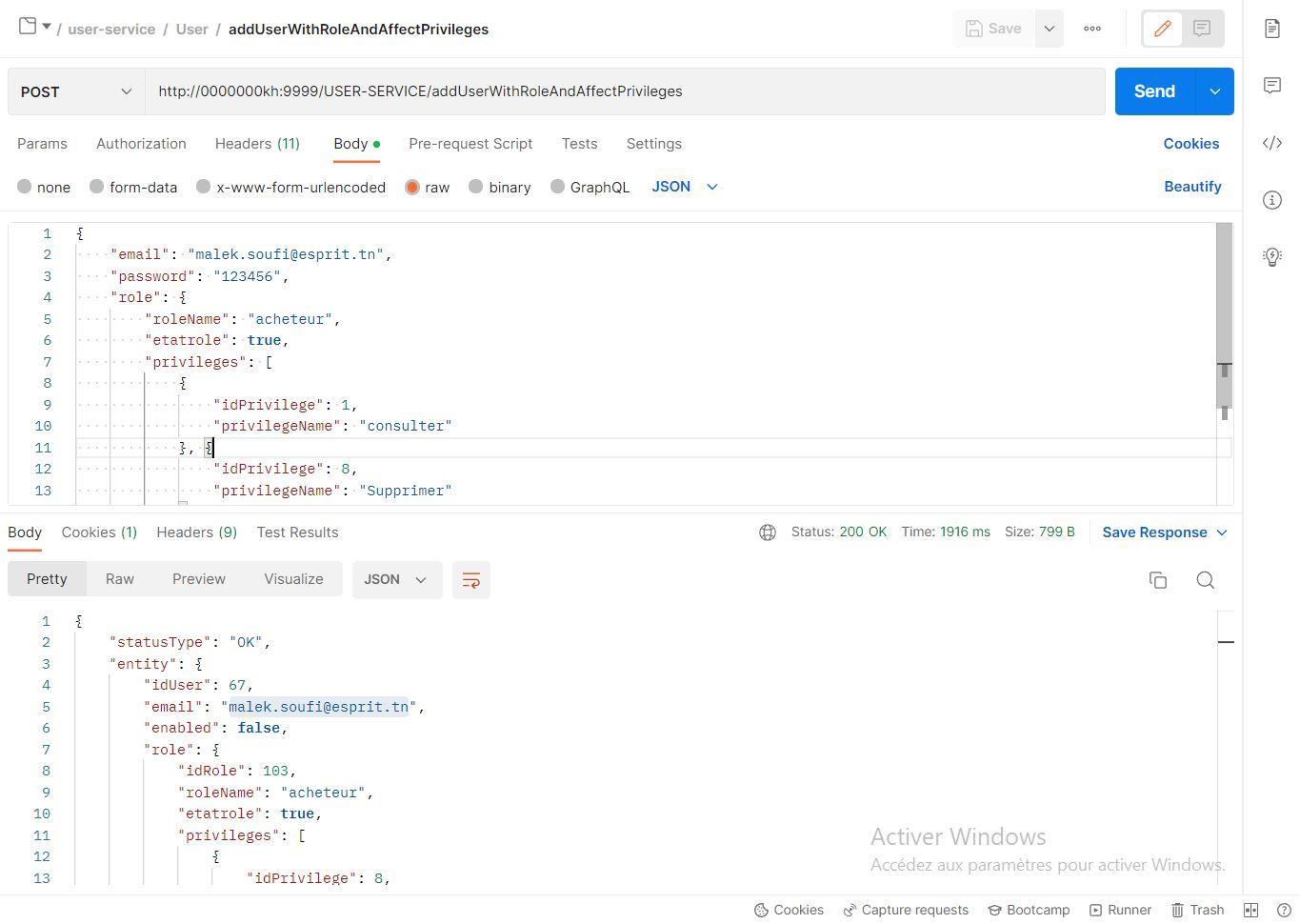
his password if he forgot it.****

Figure : Adding a New User with Role and Privilege in Postman

****

Figure : Verify email

The user can retrieve all the comments linked to an article and check if the comment contains bad words and replace it with \*\*\*\*

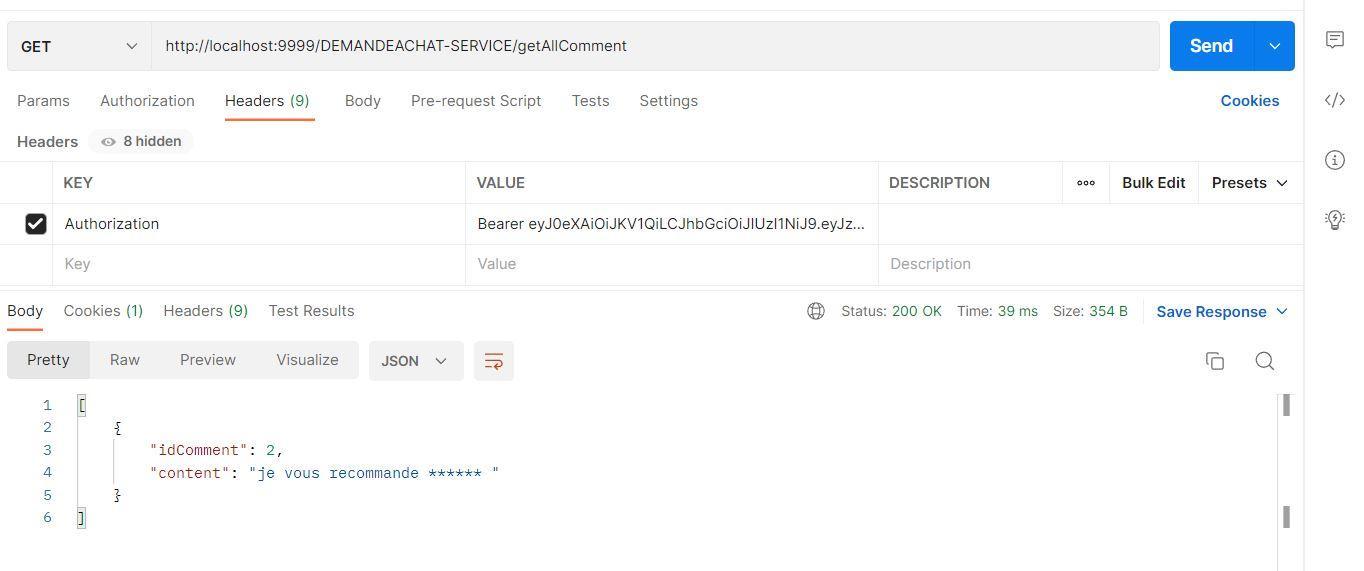
****

Figure : Get all Comment

the buyer can add a purchase request with two methods:

1- it recovers the list of articles as well as their unit of article and nature of article and assigns it to this purchase request

2- the user will create a new article with its article unit and its nature of article and assign it to this purchase request

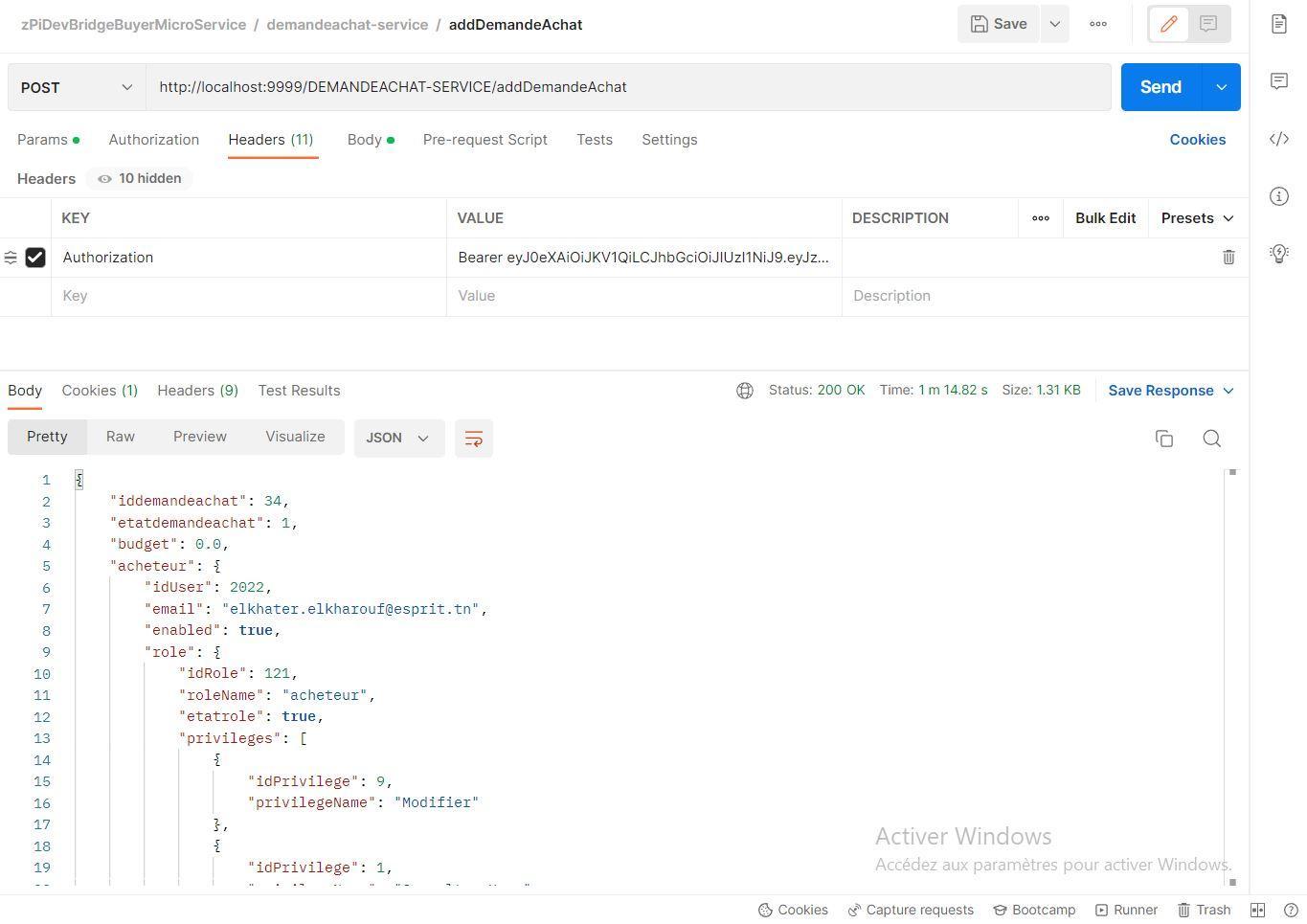
****

Figure : Adding purchase request

user can do dynamic search on article list with any item attribute

the buyer can pay for one or more orders after invoicing

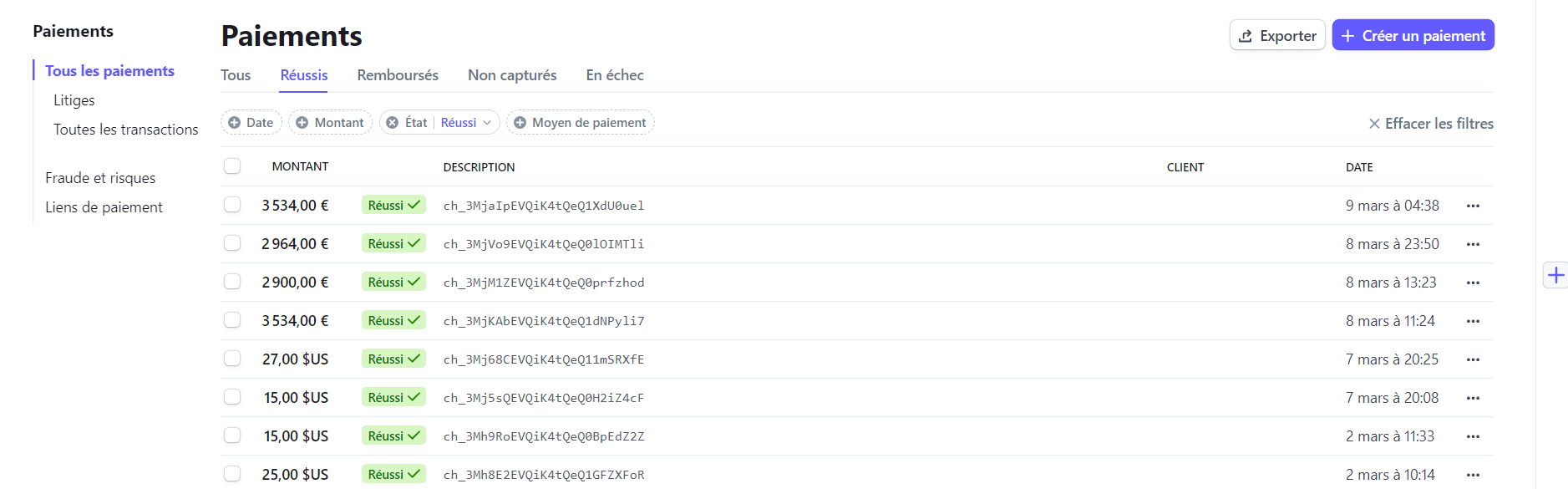
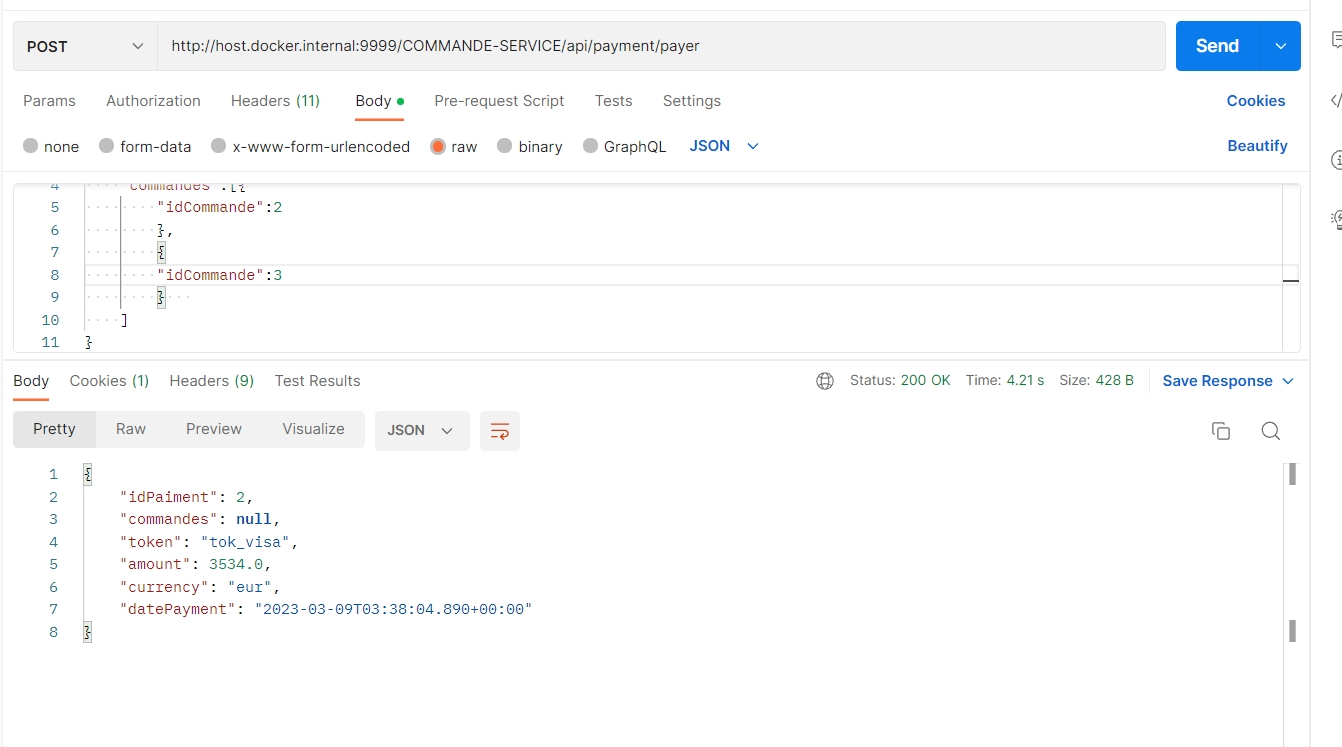


Figure : Payement and mailing of payment

supplier can add an invoice and assign it to an order

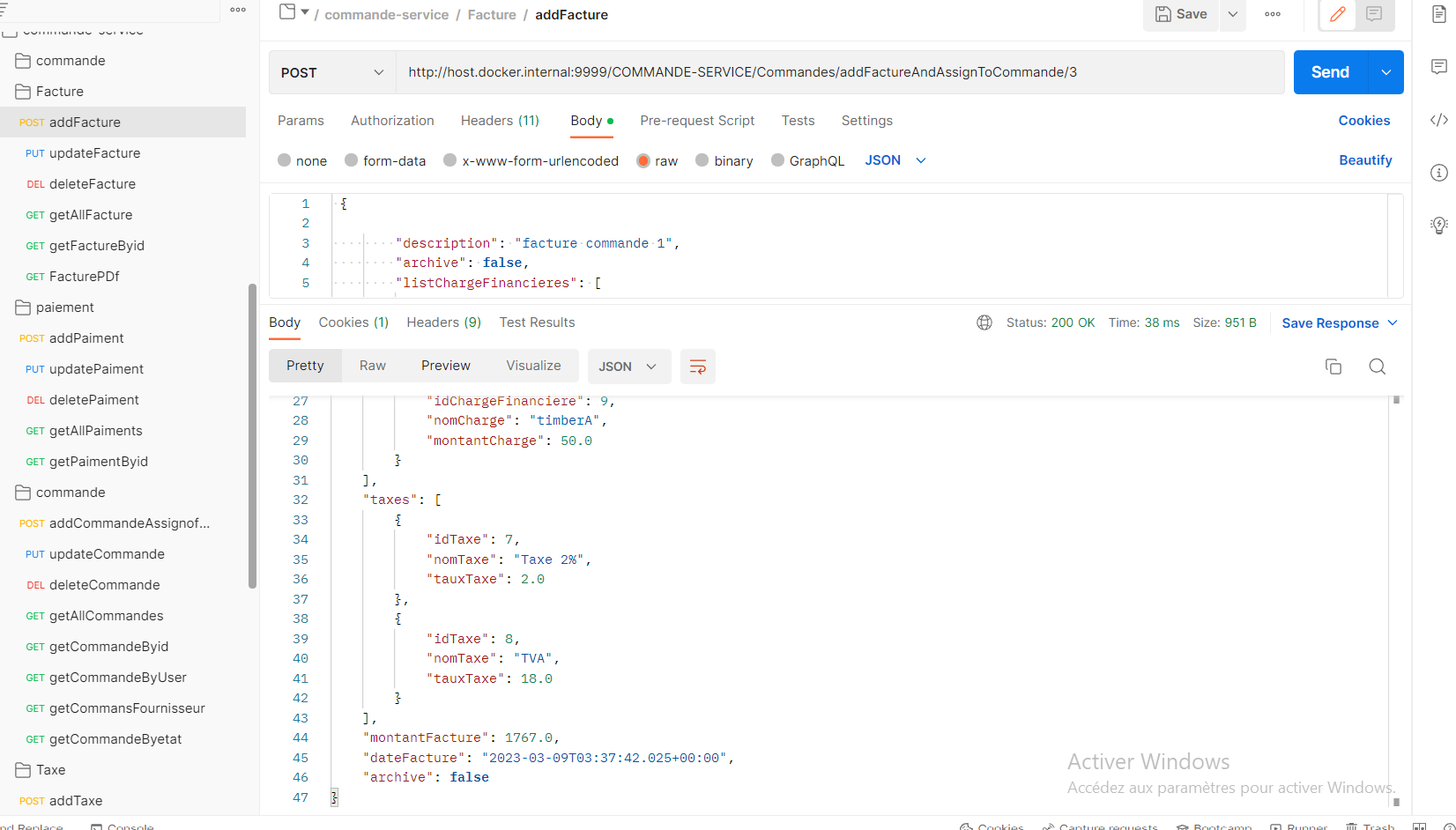


Figure : Add Invoice

## Conclusion

In summary, the Spring phase of our central purchasing application was a crucial step in the development of our project. Thanks to the use of advanced functionalities, we were able to achieve the objectives and the request of our client.

# III. Data Mining

### Subject

In recent years, with the emergence of new techniques and different inventions, many solutions have been developed to ease the lifestyle of human beings. Nowadays generations have access to buy their needs from home and that with different ecommerce websites developed in the last years. E-commerce websites often feature a variety of features such as online catalogs, shopping carts, payment gateways, and customer service tools, making it easy for consumers to find and purchase the products they need. As a result, e-commerce has become a crucial aspect of modern business, and many companies are investing in e-commerce projects to improve their online presence and increase their sales revenue.

-

### Objectives

With e-commerce, consumers can shop from the comfort of their own homes, at any time of the day or night, while businesses can reach a wider audience and increase their sales potential. Our platform offers a variety of possibilities where customers can ask for certain products and check for their availability while making sure to specify the price and details for their demand.

### Suggested solution

Artificial Intelligence has the potential to revolutionize E-Commerce by implementing some models that will help us generate better results and predict the behavior of some elements.

### Data mining Objectives

Out of the business objectives, we would extract 3 main data mining objectives as follows:

* Selling Price prediction: When a customer is submitting an offer for a certain product, a price range will be suggested in function of the different attributes’ product.
* Regression Model to predict the product price.
* Forecasting sales peak: Suppliers will be able to visualize customers’ behaviors through the time range specified and find predictions for upcoming periods.
* Prediction Model to predict the sales through time.
* Recognize fraud cases: When a customer is paying for a certain product, some frauds can happen from time to time.
* Classification Model to classify transactions into frauds and non-frauds.

## 2. Methodology of work

### 2.1. CRISP-DM

#### 2.1.1. Definition

CRISP-DM stands for Cross-Industry Standard Process for Data Mining. It is a widely-used data mining process model that outlines a set of steps to guide data analysts and data scientists through the stages of a data mining project.

#### 2.1.2. Steps of CRISP-DM plan

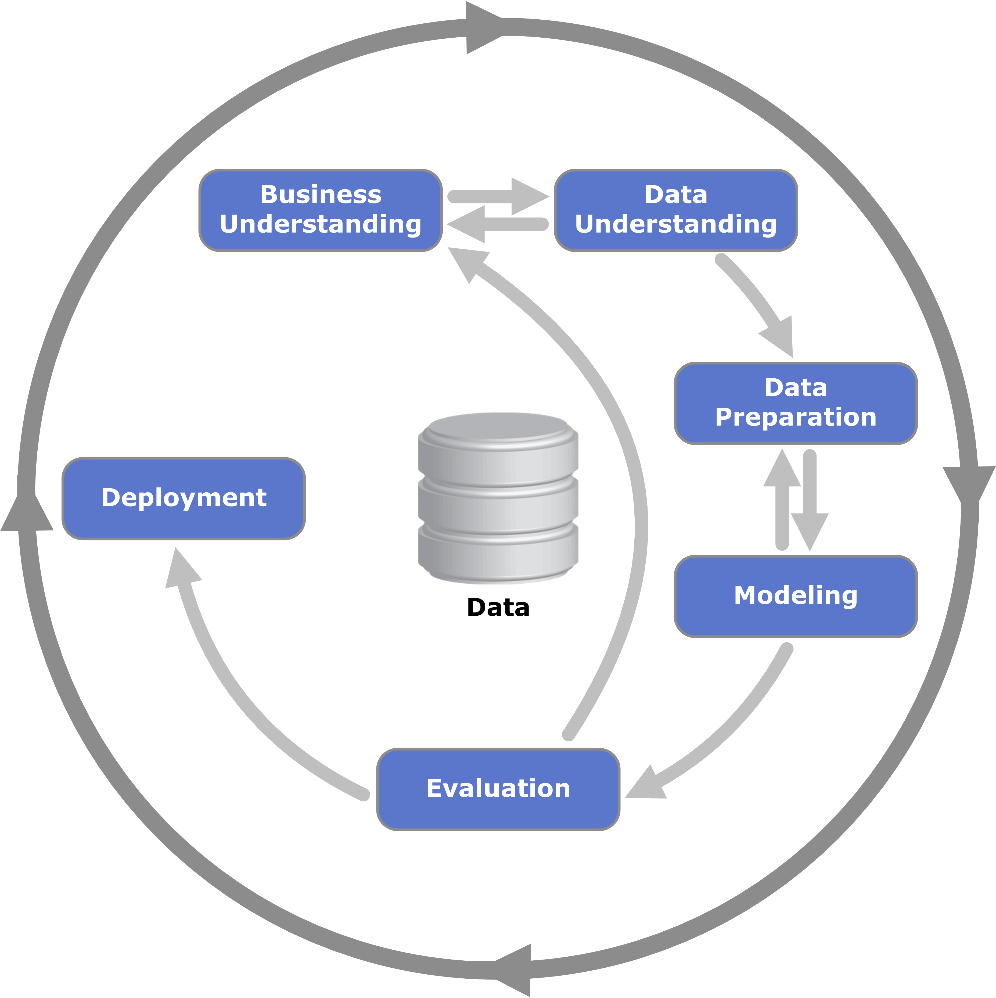


Figure : CRISP-DM methodology

The CRISP-DM model is comprised of six main phases:

* **Business Understanding:**

In this phase, we will define the problem, the solution, and extract data mining objectives from the business objectives.

* **Data Understanding:**

In this phase, the data requirements for the project are identified, and the available data is assessed for suitability and quality.

* **Data Preparation:**

In this phase, the data is cleaned, transformed, and formatted to prepare it for analysis.

* **Modeling**:

In this phase, statistical and machine learning models are developed and tested using the prepared data.

* **Evaluation**:

In this phase, the model's performance is evaluated based on predefined criteria, and modifications are made as necessary.

* **Deployment**:

In this phase, the final model is deployed into production, and the project's results are communicated to stakeholders.

The CRISP-DM model is an agile process, and the results of each phase inform the other phases. The model is intended to be flexible and iterative, allowing for adjustments and refinements as the project progresses.

## 3. Data Analysis

### 3.1. Data Collection

Data collection is the process of gathering data from various sources, such as databases, files, surveys, and external sources. It involves designing and executing data collection methods and storing the collected data for analysis.

#### 3.1.1. Data Collection Methods

There are mainly three methods to collect data:

* Data gathering through forms and questionnaires.
* Web scraping.
* Direct download.

#### 3.1.2. Data

* E-commerce Products dataset: 2451 lines in function of 8 features (Products and their categories and subcategories with the date and price).
* Transactions dataset: 623 lines in function of 12 features (Customers, payments methods and transactions amounts).
* Sales dataset: 1048575 lines in function of 26 features (item, status, amount, method and price).

#### 3.1.3. Data Understanding

Data understanding is the process of exploring and familiarizing oneself with the collected data. It involves examining the data to identify patterns, trends, and relationships and gain a better understanding of the quality and completeness of the data.

## 4. Data Preparation:

The data preparation phase involved cleaning, transforming, and organizing the raw data to ensure it was accurate, complete, and ready for analysis. As we use two different types of data, both images and problems, the preparation phase differs from one type to another.

# 

### 4.1. Visualizations:

Data visualization is an important aspect of data mining because it allows us to communicate complex information in a visual and intuitive way. A well-designed visualization can help to identify patterns, trends, and outliers in data that may not be apparent from raw data or summary statistics.

# Chart Description automatically generated with medium confidence

Figure : Study of the distribution of selling price values

# Chart, treemap chart Description automatically generated

Figure : Features correlation

# Chart, bar chart Description automatically generated

Figure : Payment Methods are most used

### 4.2. NaN and duplicate values:

NaN (Not a Number) values and duplicate values are common issues that data scientists encounter when working with datasets. Those values may be caused because of the missing data or undefined one.

# 

Figure : Exploring non null values

### 4.3. Feature Engineering:

Feature engineering is the process of transforming raw data into features that can be used to train machine learning models. The goal of feature engineering is to extract relevant information from the raw data and create informative, discriminating, and non-redundant features that capture the underlying patterns and relationships in the data.

# Text Description automatically generated

Figure : Extracting month and year from date

### 4.4. Encoding:

Encoding is the process of converting categorical data into a numerical representation that can be used as input for machine learning algorithms. Categorical data refers to data that consists of non-numerical values, such as color, gender, or country of origin.

Some methods of encoding include One Hot Encoding (creates binary column for each category), Binary Encoding (creates a binary representation for each category) and Frequency Encoding (replaces each category with its numerical values).

# Text Description automatically generated

Figure : Payment Methods

# 

# Text Description automatically generated with low confidence

Figure : Encoding

### 4.5. Scaling:

Scaling is a crucial step in the data preprocessing phase, and it is particularly important when dealing with numerical features that have different scales or units. The goal of scaling is to transform the features so that they have similar scales and ranges, which can improve the performance of machine learning algorithms.

Some examples of scaling are the minmax scaler, log transformation and standardization.

# Chart, histogram Description automatically generated

Figure : Distribution of Selling Price

## 5. Modeling:

The modeling phase in data mining involves selecting and training a machine learning model on the preprocessed data. The goal of this phase is to develop a model that can make accurate predictions on new, unseen data.

### 5.1. Hyperparameters Tuning:

Hyperparameters are parameters that are not learned by the model during training but are set by the user before training. Grid Search can be applied to find the best hyperparameters.

# Text, letter Description automatically generated

Figure : Best parameter

The result for the tuning can be retrieved as follows:

### 5.2. Prediction models:

* **Random Forest:**

In a random forest model, many decision trees are created, each based on a random subset of the training data and a random subset of the input features. When making a prediction, each decision tree in the random forest independently produces its own prediction, and the final prediction is based on the majority vote of all the individual trees.

# Graphical user interface, text Description automatically generated

Figure : Run and fit the model

# 

Figure : Predict values for the test split

* **ARIMA:**

ARIMA (Autoregressive Integrated Moving Average) is a time series forecasting model that is primarily used for making predictions about future values of a time series. It is not designed for classification tasks.

# Text Description automatically generated

Figure : Create model ARIMA

# Graphical user interface, chart, line chart Description automatically generated

Figure : Test data with prediction

### 5.3 Classification models:

* **KNN:**

KNN (K-Nearest Neighbors) is a supervised machine learning algorithm that is used for classification and regression tasks. It is a simple and easy-to-understand algorithm that is based on the idea of finding the K nearest neighbors of a given data point and using their labels or values to make a prediction.

# Graphical user interface, text, application Description automatically generated

Figure : Create model Knn

# Chart, line chart Description automatically generated

Figure : Test Model Knn

* **SVM**:

SVM works by finding the hyperplane that maximizes the margin between the two classes, which is the distance between the hyperplane and the nearest data points of each class.

# Text Description automatically generated with medium confidence

Figure : Create Model SVM

## 6. Evaluation:

Model evaluation is the process of assessing the performance of a machine learning model on a test dataset to determine how well it can generalize to new, unseen data. The goal of model evaluation is to estimate how accurately the model will predict the target variable on new data and to identify any potential problems with the model.

# Square Description automatically generated

Figure : correlation circle

Text, application

Description automatically generated

# 

Figure : Access the model using R-squared score

# Graphical user interface, text, application Description automatically generated

Figure : Test Error

## 

## Conclusion

To sum up, the data mining phase of our central purchasing app has been a crucial step in the development of the system. Through the use of advanced data analysis techniques, we were able to extract valuable insights from large amounts of data, which has helped us to make more informed decisions and optimize the procurement process. We have successfully implemented data mining algorithms to identify trends, patterns, and anomalies in the data, which has allowed us to detect fraud, reduce costs, and improve the overall efficiency of the system.

The data mining phase has been an essential part of our efforts to develop a comprehensive and effective central purchasing app, and we look forward to further refining and expanding this aspect of the system in the future

# IV. Front office web

## Introduction :

The Angular frontend app has revolutionized the way we manage procurement in our central purchasing system. This report chapter will provide an in-depth analysis of the benefits and features of this powerful technology, exploring how it has transformed the procurement process for both buyers and suppliers. From its user-friendly interface to its dynamic and responsive design, the Angular frontend app has brought a new level of efficiency and productivity to our central purchasing system. Join us as we dive into the details of this innovative technology and discover how it can enhance your procurement experience.

## 1. Scenario Test

The purchase request addition interface for our web application enables buyers to add purchase requests and launch lists of items by entering the necessary information, providing an organized and efficient way to manage procurement

# 

Figure : List of purchase request

# 

Figure : purchase request addition

The user addition interface for our web application enables operators to add users with specific roles and privileges by entering their necessary information

# Graphical user interface, application Description automatically generated

Figure : user addition interface

Login interface for our web application allows users to access their account by entering their email and password, providing a secure and convenient way for users to authenticate and access our platform.

# Graphical user interface, website Description automatically generated

Figure : Interface login user

The order management and online payment interface for our web application enables buyers to manage their orders and make payments online by entering the necessary information

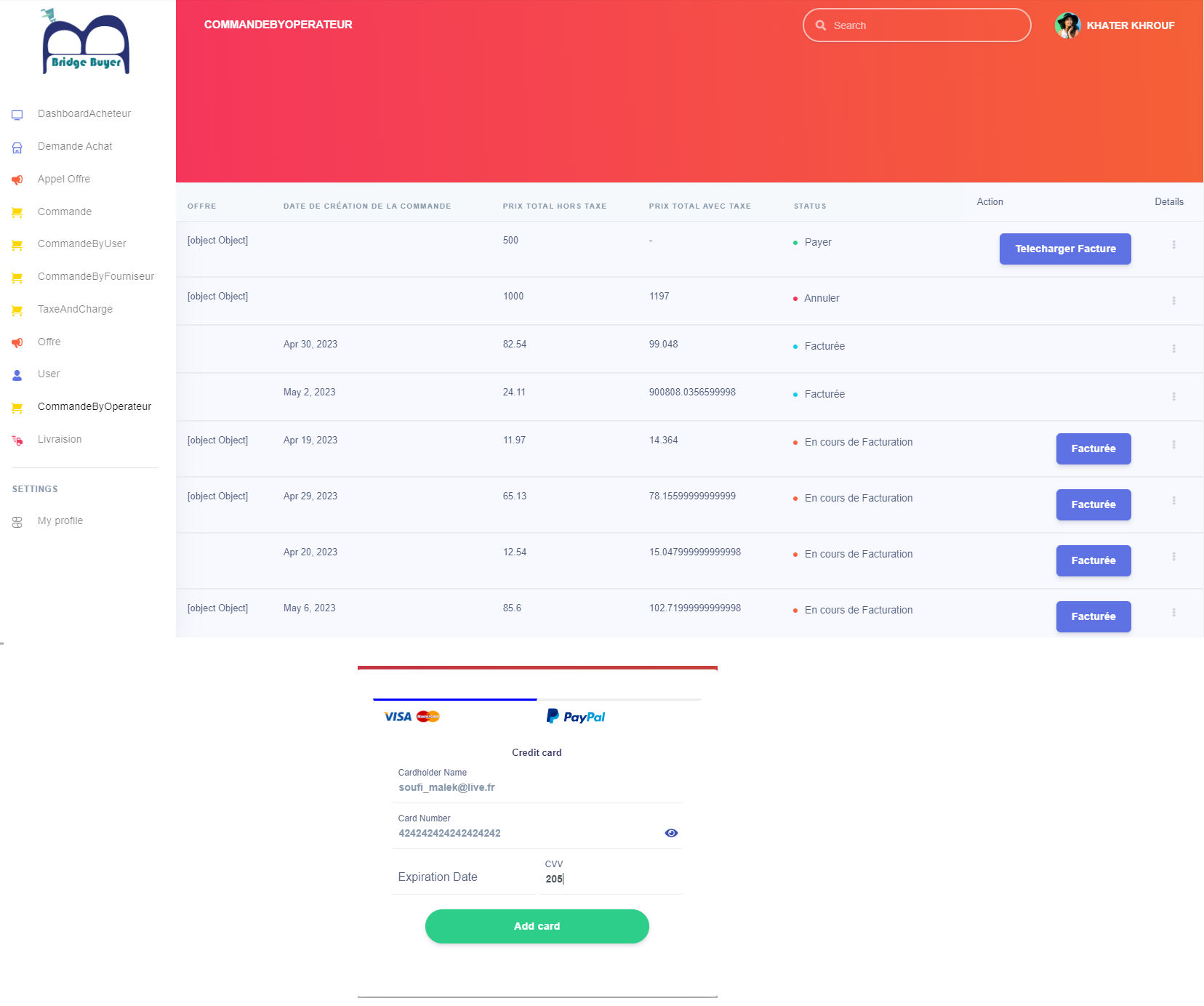


Figure : order management and online payment interface

The online invoice addition interface for our web application enables suppliers to add invoices by entering the necessary information, including financial charges and taxes for a particular order

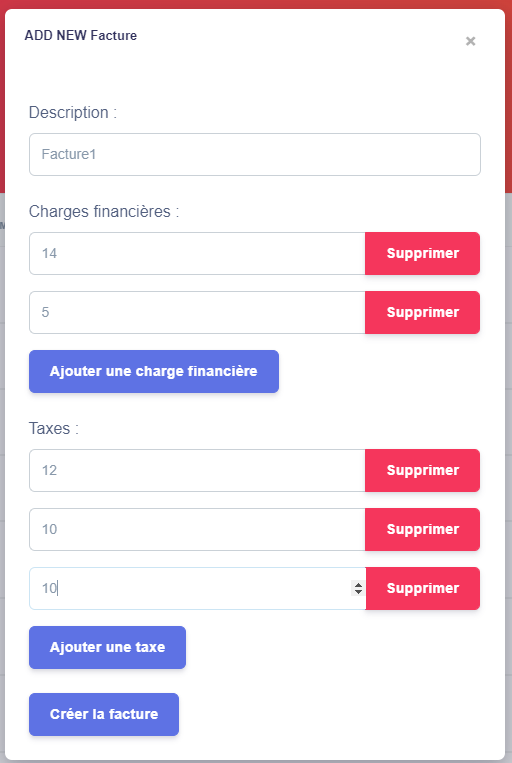


Figure : invoice addition interface

## Conclusion

In conclusion, the development of a central purchasing app using Angular is a significant achievement that promises to streamline procurement operations for businesses. The app offers a sleek and intuitive user interface, enabling users to easily search for and purchase products from multiple suppliers in one place. The use of Angular ensures that the app is highly responsive and delivers a seamless user experience across different devices.

**General conclusion**

In today's fast-paced business world, the importance of efficient procurement processes cannot be overstated. Procurement is a crucial function that directly impacts a company's bottom line, and businesses are always looking for ways to optimize this process. The development of a central purchasing app using Angular, Spring Boot, and data mining is a prime example of how technology can be harnessed to improve procurement operations. This app promises to streamline purchasing, save time and money, and provide valuable insights that can inform purchasing decisions. It represents a significant step forward in procurement and highlights the power of combining cutting-edge technologies to drive innovation.

From a broader perspective, the central purchasing app is not just a remarkable achievement for individual businesses but also has the potential to drive industry-wide change. By improving procurement processes, businesses can become more agile and responsive to changing market conditions. This app can also foster greater transparency and collaboration between buyers and suppliers, leading to more sustainable and ethical business practices.

Furthermore, the development of this app sets a precedent for future innovation in procurement. As technology continues to evolve, we can expect to see more advanced tools and applications that take advantage of emerging technologies like machine learning, blockchain, and the Internet of Things. The central purchasing app with Angular, Spring Boot, and data mining is just the beginning, and we can expect to see more exciting developments in the field of procurement in the years to come.

**Bibliography**

<https://b24-lwnxv4.bitrix24.com/company/personal/user/6/tasks/?F_STATE=sVg0>

<https://sabricole.developpez.com/uml/tutoriel/unifiedProcess>

[http://fdigallo.online.fr/cours/uml.pdf](https://sabricole.developpez.com/uml/tutoriel/unifiedProcess)

[https://cdn.uclouvain.be/public/Exports%20reddot/iag/documents/WP\_109\_Kolp.pdf](https://sabricole.developpez.com/uml/tutoriel/unifiedProcess)

[https://www.slideshare.net/createursite/gp-unified-process](https://sabricole.developpez.com/uml/tutoriel/unifiedProcess)

[https://slideplayer.fr/slide/482289](https://sabricole.developpez.com/uml/tutoriel/unifiedProcess)

[http://www.univ-orleans.fr/lifo/Members/todinca/Cours/GL/cours\_umlProcessusUnifie.ppt](https://sabricole.developpez.com/uml/tutoriel/unifiedProcess)

[http://www-sop.inria.fr/members/Serena.Villata/Resources/COO-s6-C02.pdf](https://sabricole.developpez.com/uml/tutoriel/unifiedProcess)

[http://files.gl3.webnode.fr/200000067-4b6db4c670/Chapitre1-ACOO2.pdf](https://sabricole.developpez.com/uml/tutoriel/unifiedProcess)