People's Democratic Republic of Algeria
Ministry of Higher Education And Scientific Research

Hassiba Benbouali University Of
CHLEF

Faculty Of Exact Sciences And Computer Sciences

Department Of Computer Sciences

Graduation Project For Obtaining The bachelor's degree

Inventory Management System

Submitted by

WALID MEGUENNI

SALAHEDDINE BELAID

Supervised by

Dr.chakib nehnouh

Examining board

BOUSAHBA NASSIMA

GHOULAM AICHA

College Year: 2020/2021

Abstract

Storage today has become an essential pillar, regardless of its type, and with the economic development witnessed by the world and the growth in the volume of commercial exchanges, which led to the creation of a huge amount of data that must be stored, and of course, it cannot be stored in paper records due to the difficulty of storing them in them because it takes a long time to write and copy This does not please us at all, especially in the age of speed, and also because paper records are easy to damage (burning, eroding, getting wet) and because of the many writing errors.
Therefore, we need other solutions to store this important data, and there is no better than electronic management programs that preserve records of stores, exchanges, and warehousing work by storing records of sales, purchases, sales, products, financial and records of workers and customers also in digital databases, thus eliminating paper stock and human and recent errors Manual and speed up operations .

Contents

List of Figures

List of Tables

Chapter 1

Capture and analysis of needs

1.1 Introduction

An inventory management system is helpful for business operators, where shopkeepers keep records of purchases and sales. Mismanaged inventory means disappointed customers, too much cash tied up in slower sales and warehouses.[1]
This inventory is eliminated paperwork, human faults, manual delay and speed up the process. This inventory management system will have the ability to track sales and available inventory tells a shopkeeper when it's time to reorder and how much to purchase..
An inventory Management System is important to ensure quality control in businesses that handle transactions revolving around consumer goods.

1.2 Definition

1.2.1 inventory management program

An inventory management program is a combination of a software system and hardware designed to serve this system in order to keep track of inventory levels, sales, and deliveries. Companies use stock management systems to avoid overstocking, product interruptions,[1] and disposal of paper stock. This program converts unstructured data into structured data by converting it into tables that can be Read it easily and it should include:

- A system for identifying each stored item and its associated information.
- A program that provides a central database and a reference point for all inventory, as well as the ability to analyze data.

1.2.2 Goals

The objectives of the inventory management application are multiple:

- The application should also allow day-to-day management of sales and purchases:
    1. Order creation (sales and purchases) and their follow-up.
- Customer and supplier management.
- The application will also have to ensure a rapid response to the user's needs.

1.2.3 Inventory management system users

Users are divided into two categories:

- **Admin**: He is responsible for managing the site .
- **Participants**: they are business owners (company owner, store owner, shop owner...).

1.2.4 Advantages Inventory Management System

Without an inventory management system, the goods and products that flow through the organization will inevitably be in disarray. [1]So the benefit of this software is:

1. Cost savings.
2. improved cash flow.
3. Increased efficiency.
4. Warehouse organization.
5. Updated data.
6. Data security.
7. Improved supplier, vendor, and partner relationships.

1.2.5 Disadvantages inventory management software

The disadvantages of inventory management software are It's[1]:

1. Inventory management system is an expensive application.
2. We cannot say that the warehouse management system is an easy-to-use application, as it is somewhat complicated, as it forces business owners to train their workers on how to use it and extrapolate the data.

users	description
browser	Anyone can enter and browse the website .
admin	Anyone can enter and browse the website .
subscriber	The subscriber is anyone who has a account in the application .

Table 1.1: Table Definition of users

1.3 Capture and analysis of needs

1.3.1 Introduction

In Section Two, Chapter One, we will identify the actors in our software project in the needs analysis phase as well as functional needs where use cases, core functions, and user roles are defined.

1.3.2 Preliminary Study

1.3.2.1 Project description

This project aims to establish a unified central database to store through the development of Web applications via the Internet for operators of the business, whatever their field and to facilitate their work and effort, money, time, and reason to make the application of uniform as applications of social networking is a high price for the management system of stocks and this will help the owners of small business who cannot Pay for it, and the new juniors all have to subscribe annually or monthly for a small amount.

1.3.2.2 Identification of users

There are three types of users in this program, admin and subscriber ,browser, each with its own description and different roles:

1.3.3 Needs capture

1.3.3.1 Capture of functional requirements

To be able to identify the functional needs of our studies, we must divide our program into sections so that it is easy to identify them:
The division of the project into two main units:

1. Account management
2. User Management
3. Subscribers Management

4. Order Management
5. product management

1.3.3.1.1 Identification the functionality

unit	Functions	users
Account management	• Login • Sign up • Logout • Update Personnel Information • Upload Photo Profile • change password	• Admin • Subscriber
User Management	• add Person • update Person • remove Person	• Admin • Subscriber
Subscribers Management	• remove Subscribe	• Admin
Product management	• add Product • update Product • remove Product	• Admin • subscriber
Order Management	• add Order • update Order • remove Order	• Admin • subscriber
Material Management	• add Material • update Material • remove Material	• Admin • subscriber

Table 1.2: Table Definition of Functions

1.3.3.2 Capture of technical needs

Capturing technical needs identifies all technical limitations and options such as data access performance, system security, operability, application integration, system size, and method of use. The capture of technical needs is as follows:

- Capture of software specifications .
- Capture of specifications related to hardware configuration .

1.3.4 Application architecture

This project is engineered on MERN stack technology which can be 3 layers:

1. **Interface layer**:It is the front-end layer.
2. **Function layer (back-end)**:It is a hidden layer that handles user requests and is an intermediary between the interface and the database .
3. **Data layer**:All data is stored and preserved .

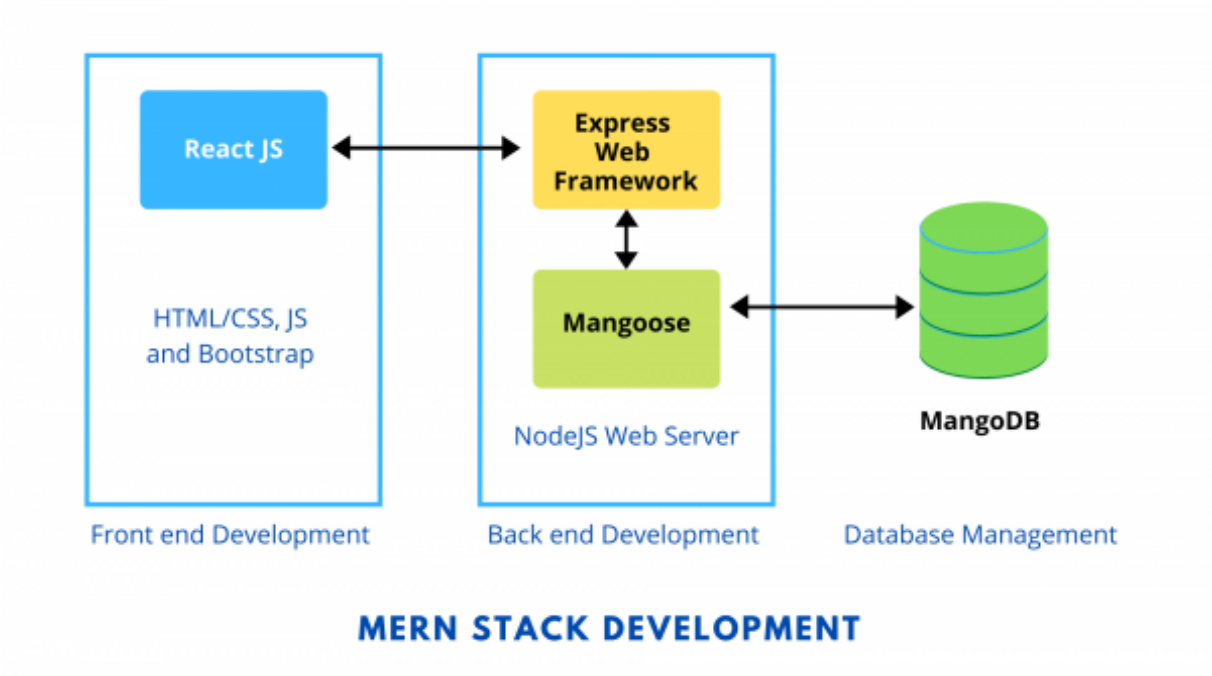

Figure 1.1: architecture of system

Chapter 2

Analysis and concept

2.1 abstract about UML

UML is an abbreviation of a unified modeling language that consists of a set of drawings and diagrams and aims to provide a standard way to visualize and define the design of the system. UML is an important part of the software development stages and one of the most important tools that a software engineer works with.
The creation of UML was originally motivated by the desire to standardize the disparate notational systems and approaches to software design. It was developed at Rational Software in 1994–1995, with further development led by them through 1996.In 1997, UML was adopted as a standard by the Object Management Group (OMG), and has been managed by this organization ever since. In 2005,[2] UML was also published by the International Organization for Standardization (ISO) as an approved ISO standard. Since then the standard has been periodically revised to cover the latest revision of UML.

2.2 use case

A user case state diagram belongs to "behavior diagrams" one of the types of UML diagrams. use case model describes a system's functional requirements in terms of use cases. It is a model of the system's intended functionality (use cases) and its environment (actors). Use cases enable you to relate what you need from a system to how the system delivers on those needs.[3]
Think of a use-case model as a menu, much like the menu you'd find in a restaurant. By looking at the menu, you know what's available to you, the individual dishes as well as their prices. You also know what kind of cuisine the restaurant serves: Italian, Mexican, Chinese, and so on. By looking at the menu, you get an overall impression of the dining experience that awaits you in that restaurant.[4]
Think of a use-case model as a menu, much like the menu you'd find in a restaurant. By looking at the menu, you know what's available to you, the individual dishes as well as their prices. You also know what kind of cuisine the restaurant serves: Italian, Mexican, Chinese, and so on. By looking at the menu, you get an overall impression of the dining experience that awaits you in that restaurant.

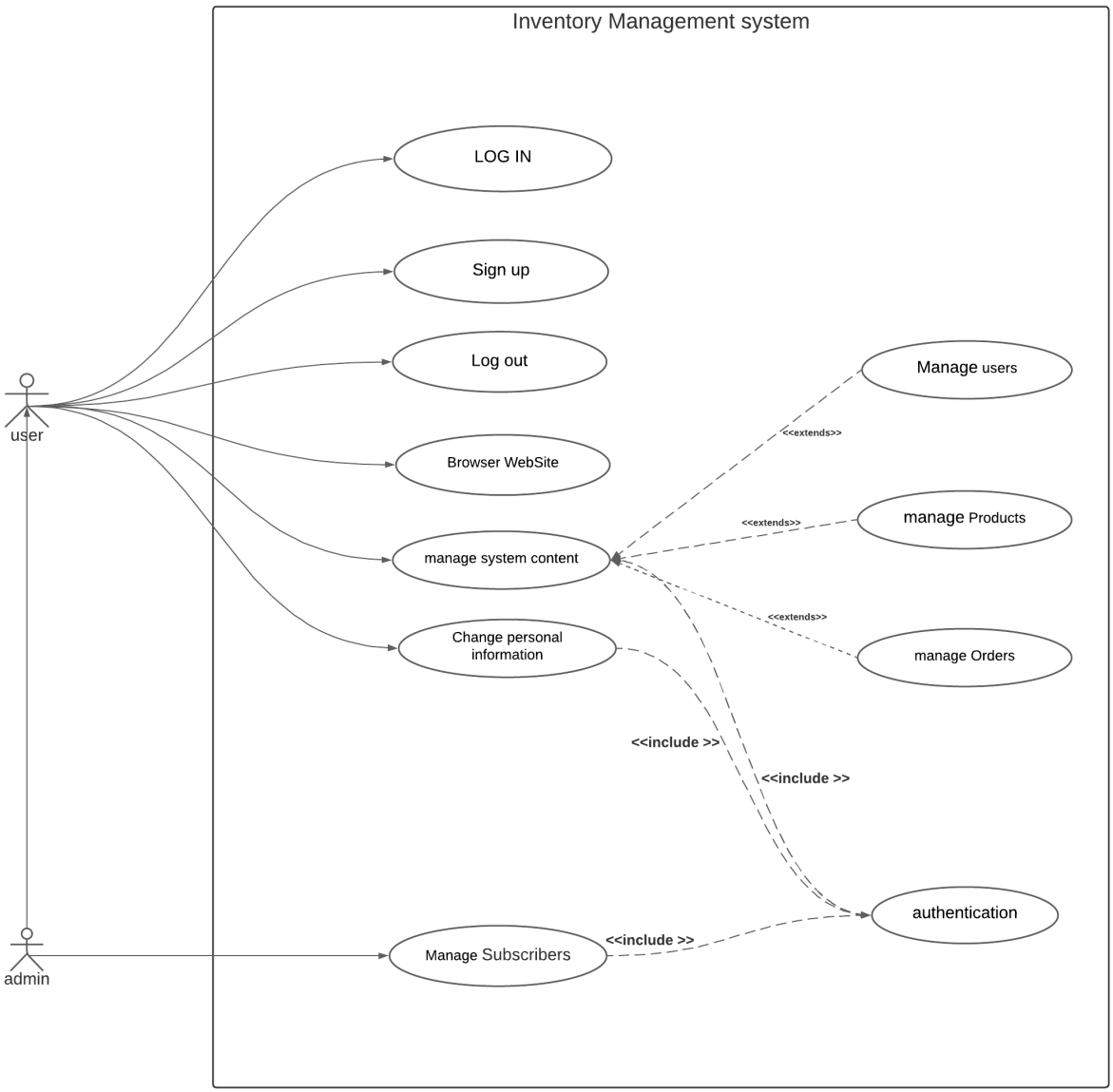

Figure 2.1: The diagram of the use case

Use Case Number	UC-01
Use Case Name	Sign up
Overview	he create new account into the spacestor
Actor(s)	1. User
Pre-condition(s)	user browsing in website, and have an internet connection,
Scenario Flow	1. user open the website 2. system display page have login and register 3. user click Sign up Button 4. system show the formula of sin up 5. user fill the form 6. system displays notification confirm email 7. user click link confirm email 8. system displays the login page
Alternate Flows	1. if the email don't exist on the database or not valid show notification error 2. if the password is wrong show notification error
Post Condition	display sigup page
Use Case Number	UC-2
Overview	user login to his account into the spacestore
Actor(s)	1. User 2. admin
Pre-condition(s)	user browsing in website, and have an internet connection, user have account
Scenario Flow	1. user open the website 2. system display page have login and register 3. user click login Button 4. system show the formula of login 5. user fill the form 6. system displays the main page

Alternate Flows	1. if the email don't exist on the database or not valid show notification error 2. if the password is wrong show notification error
Post Condition	display login page
Use Case Number	UC-03
Use Case Name	Log out
Overview	user want to logout the spacestor
Actor(s)	1. User 2. admin
Pre-condition(s)	user browsing in website, and have an internet connection,user have already an account
Scenario Flow	1. click on button Logout 2. system display page logout 3. user click logout Button 4. system display page login
Alternate Flows	1. if the command dosn't have autrozation show notification error
Post Condition	display Logout page
Use Case Number	UC-04
Use Case Name	manage Product
Overview	he want to beggin work on the spacestor
Actor(s)	1. User 2. admin
Pre-condition(s)	login authorization

Scenario Flow	1. Click button add 2. Fill The data of the form 3. Click button create 4. Click button edit 5. fill the data of the form 6. Click button Update 7. Click on image 8. Click button upload 9. choise image 10. Click button save 11. click botton delet 12. click botton confirme delete
Alternate Flows	1. if no authrization show notification error
Post Condition	display Product page

Table 2.1: Table of Diagram use case

2.3 class diagram

In the Unified Modeling Language, a component diagram depicts how components are wired together to form larger components or software systems. It is used for general conceptual modeling of the structure of the application, and for detailed modeling, translating the models into programming code. [5] and the class diagram is the main building block of object-oriented modeling.

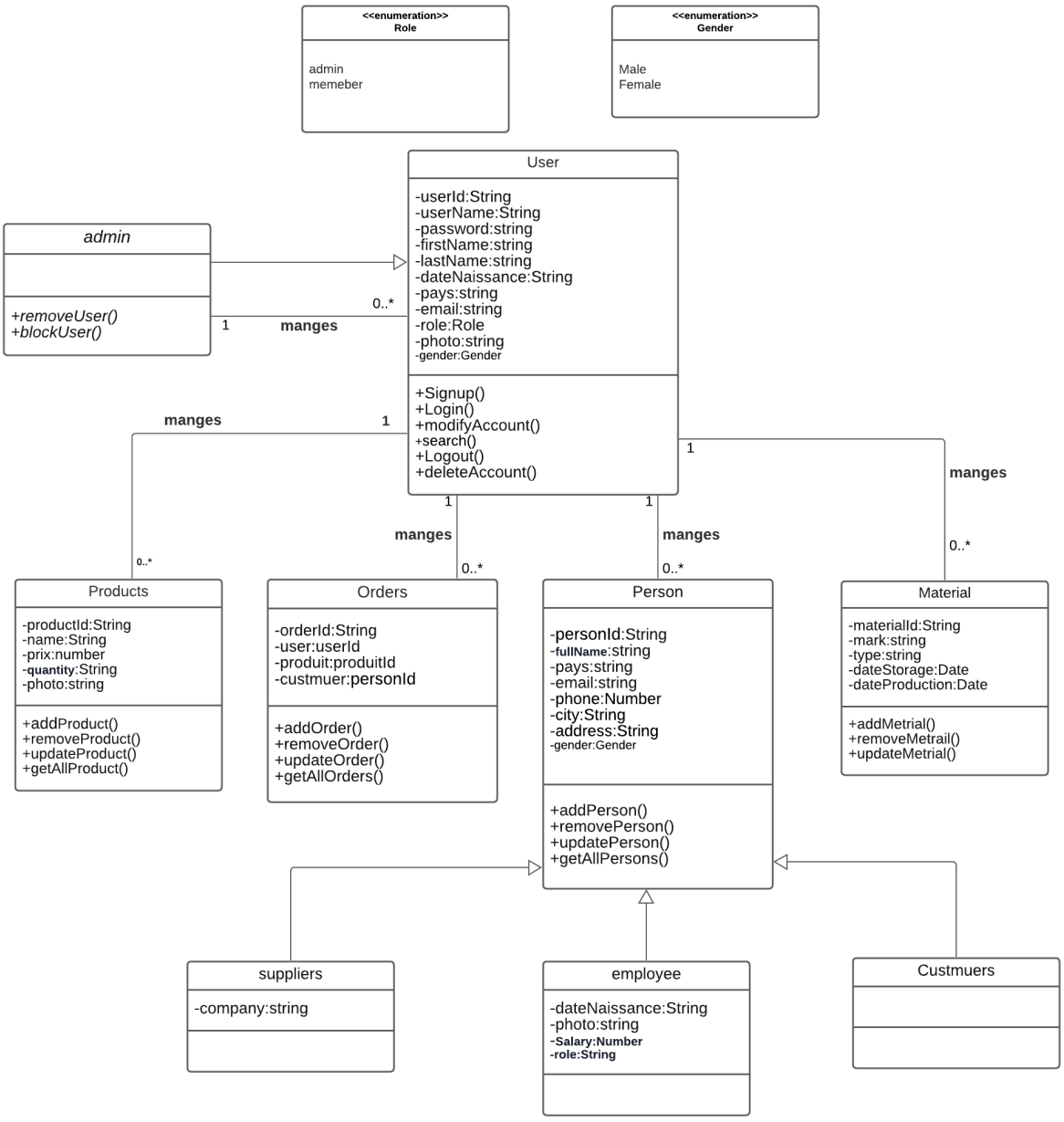

Figure 2.2: The diagram of the class diagram

2.4 Sequence Diagram

The Sequence Diagram models the collaboration of objects based on a time sequence[6]. It shows how the objects interact with others in a particular scenario of a use case.

2.4.1 Sequence Diagram of create account (sign up)

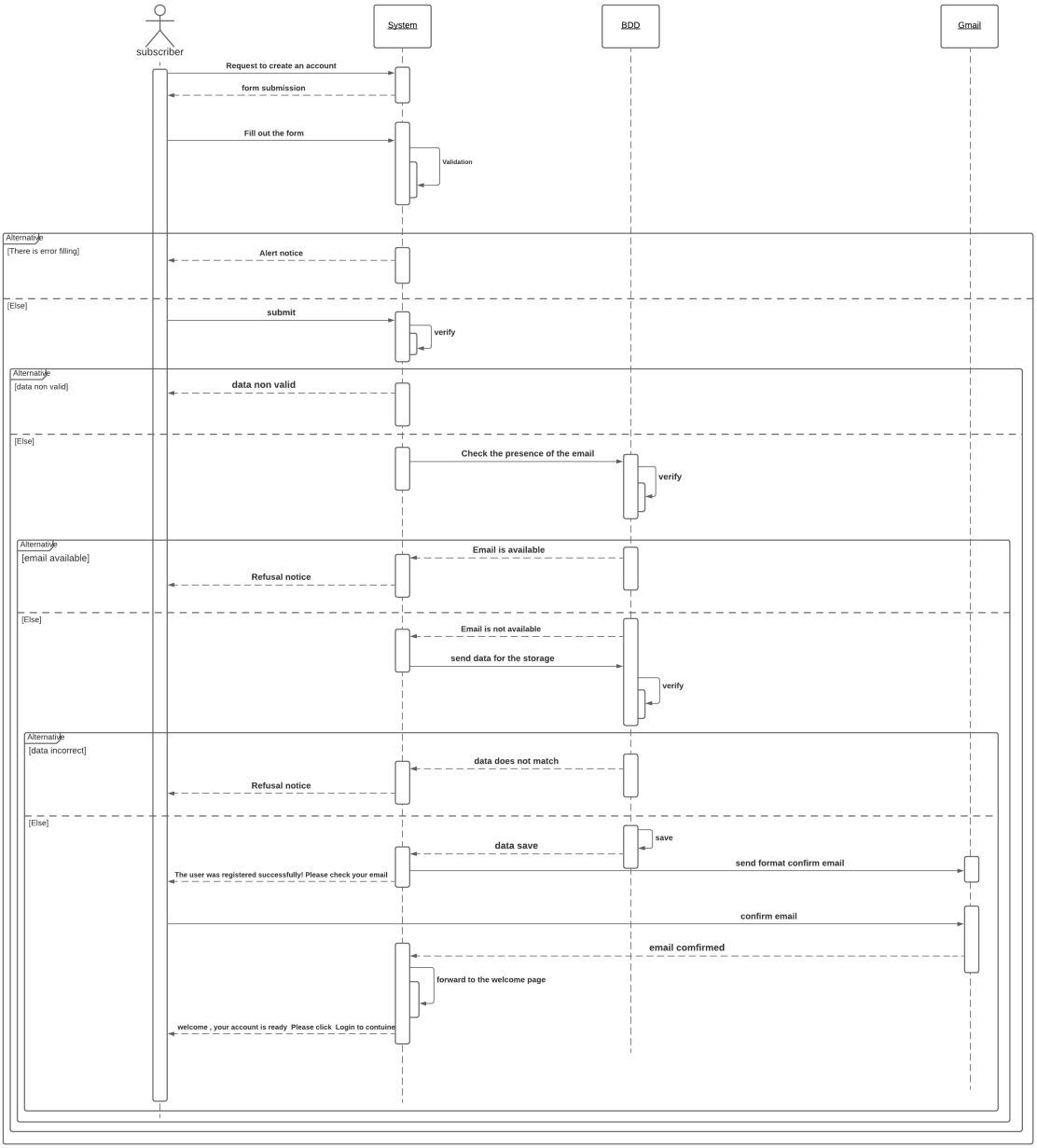

Figure 2.3: The diagram of the create account (sign up)

2.4.2 Sequence Diagram of Login (authentication)

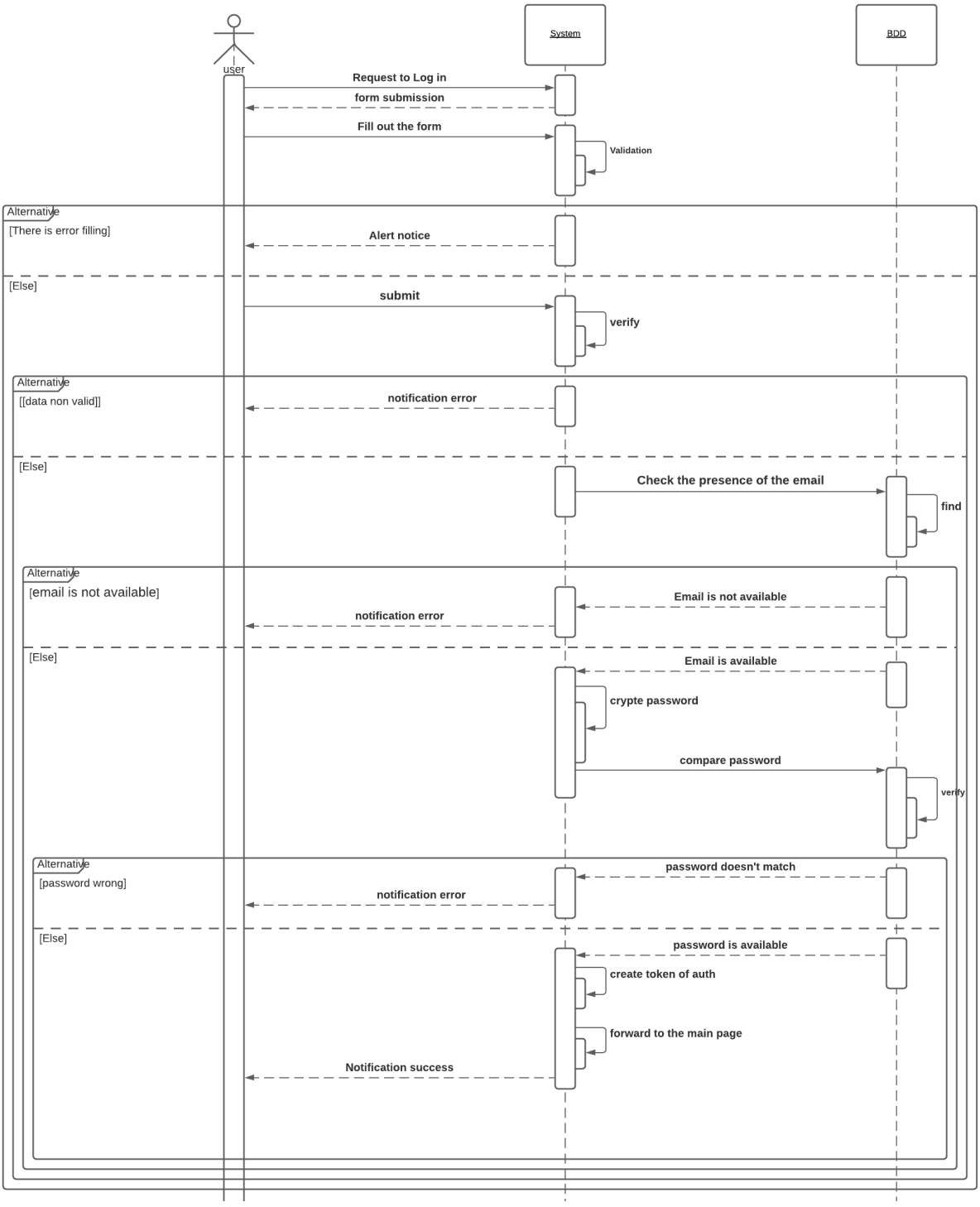

Figure 2.4: The diagram of the Login (authentication)

2.4.3 Sequence Diagram of add new Product

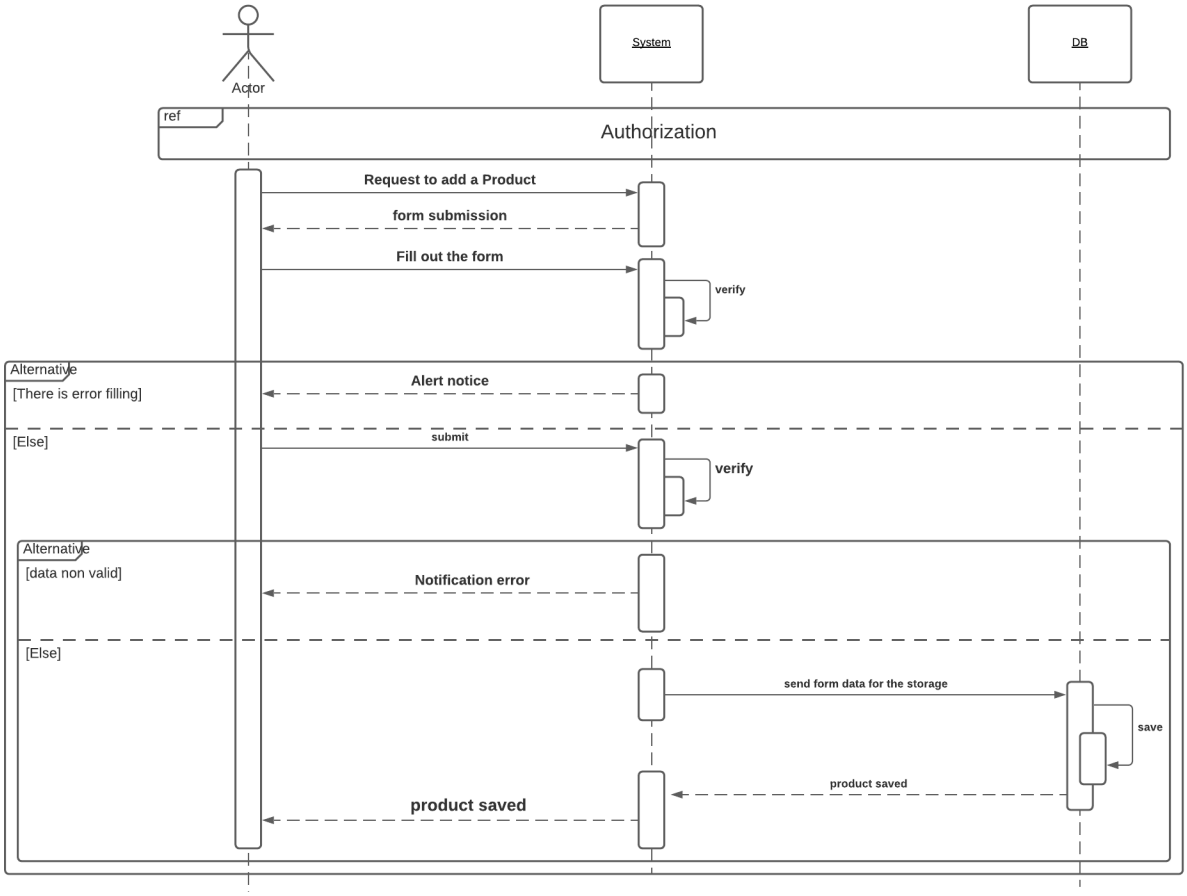

Figure 2.5: The diagram of the add new Product

2.4.4 Sequence Diagram of Update Product

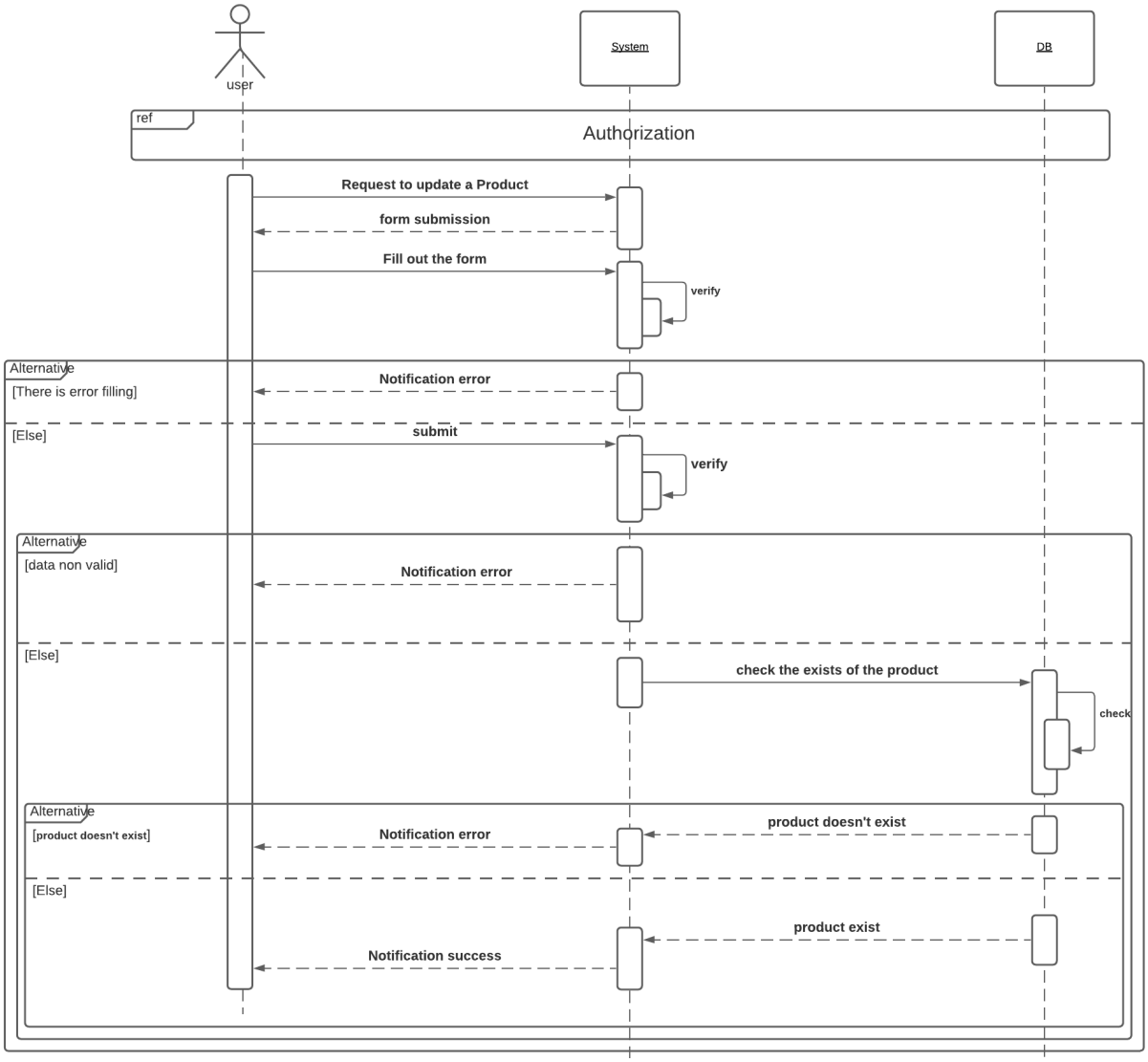

Figure 2.6: The diagram of the Update Product

2.4.5 Sequence Diagram of Delete Product

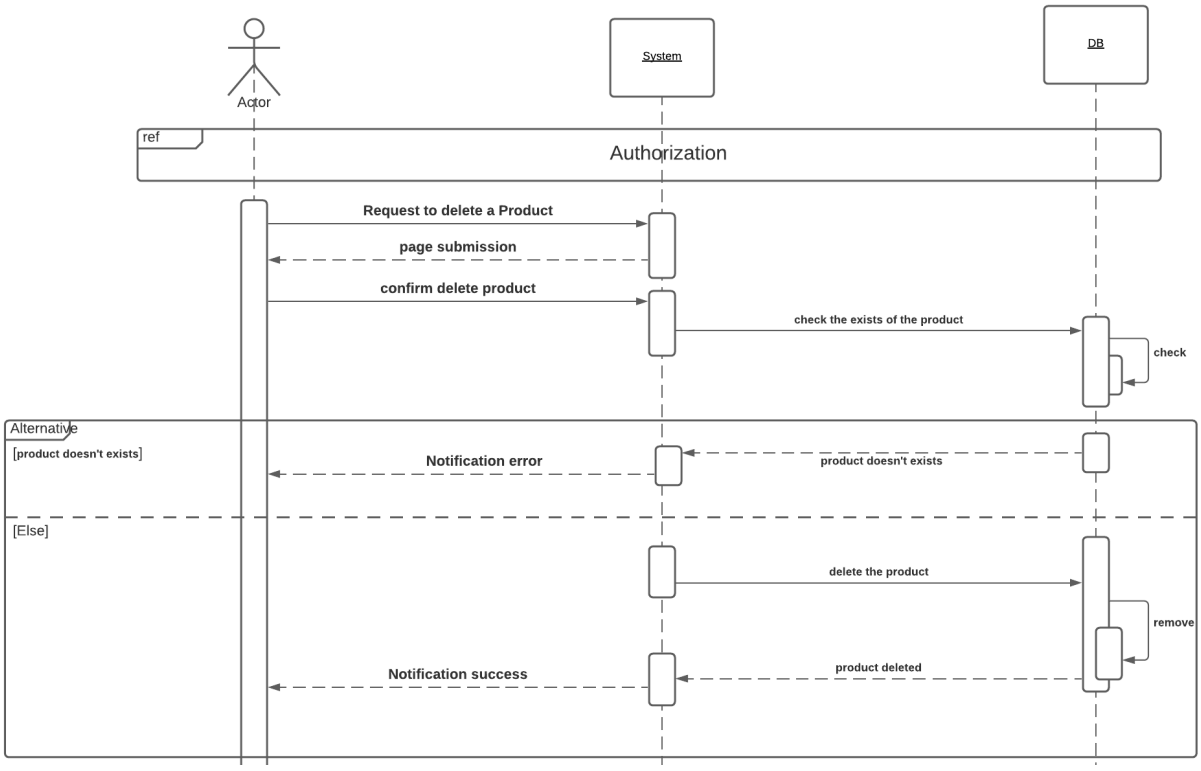

Figure 2.7: The diagram of the Delete Product

Chapter 3

Implementation

3.1 Description of the development environment

3.1.1 Express Js

Express.js is a back-end web application framework for Node.js, released as free and open-source software under the MIT License[10]. It is designed for building web applications and APIs, written in JavaScript and hosted within the Node.js runtime environment, Express js is a minimal and flexible framework that provides a robust set of features for web and mobile applications.
The original author, TJ Holowaychuk, described it as a Sinatra-inspired server, meaning that it is relatively minimal with many features available as plugins.[11] Express is the back-end component of popular development stacks like the MEAN, MERN, or MEVN stack.

3.1.2 Node Js

Node.js is an open-source, cross-platform, back-end JavaScript runtime environment that runs on the V8 engine and executes JavaScript code outside a web browser,[7] allows you to run JavaScript on the server[9].it was developed by Ryan Dahl in 2009 and its latest version is v0.10.36. and two of many features of Node.js

- Asynchronous and Event Driven All APIs of the Node.js library are asynchronous, that is, non-blocking. It essentially means a Node.js based server never waits for an API to return data.
- Very Fast Being built on Google Chrome's V8 JavaScript Engine, the Node.js library is very fast in code execution.

there are Applications and companies which are using Node.js. like eBay, General Electric, GoDaddy, Microsoft, PayPal, Uber, Wikipins, Yahoo!, and Yammer to name a few.[8]

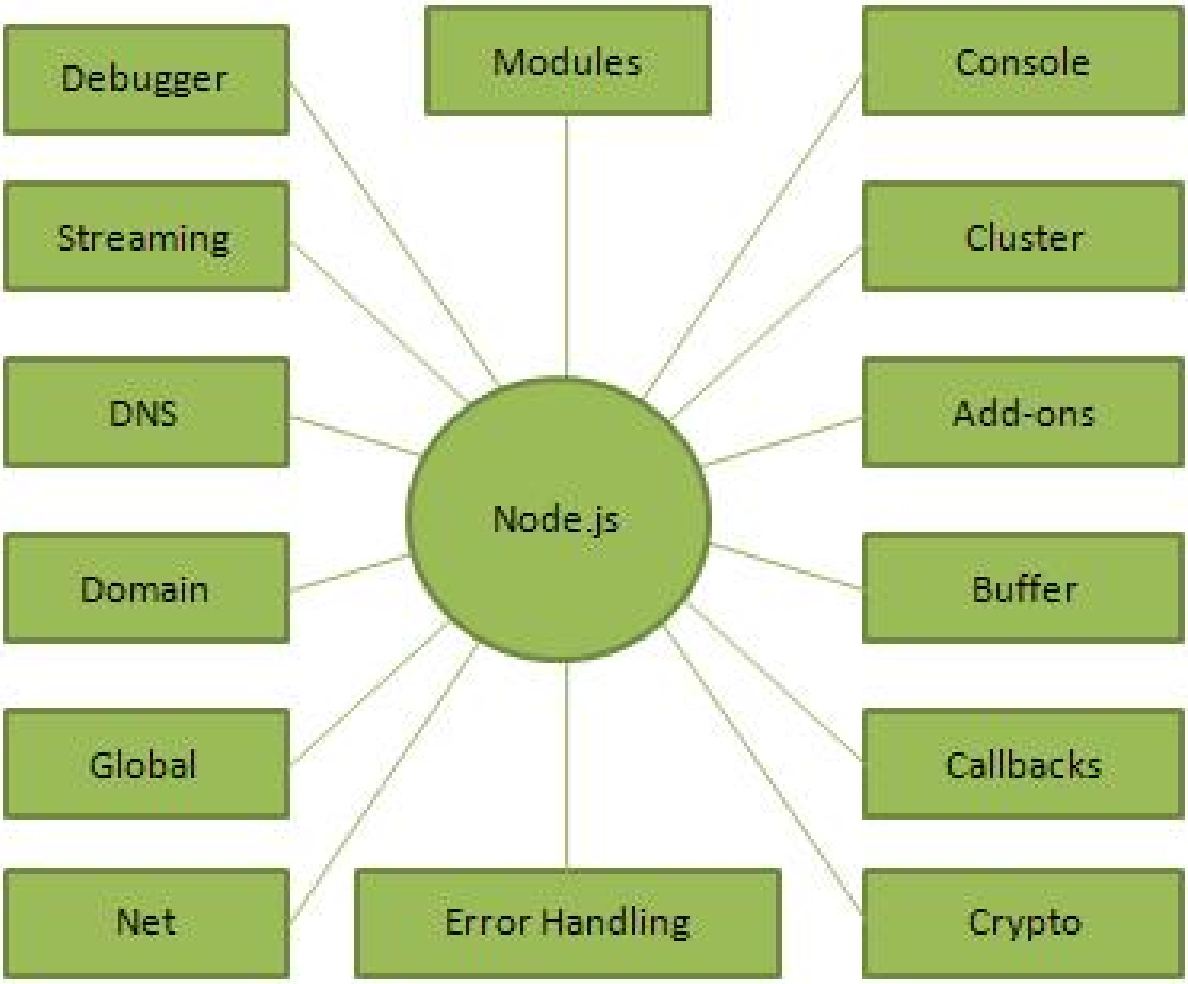

Figure 3.1: concept of node js

3.1.3 Mongo db

MongoDB is an open-source document database cross-platform document-oriented database program and leading NoSQL database[12], uses JSON-like documents with optional schemas "name of this feature is mongoose ". MongoDB is written in C++ and developed by MongoDB Inc. and licensed under the Server Side Public License (SSPL).

3.1.4 RESTful API Web Services

REST stands for REpresentational State Transfer. REST is a web standards-based architecture and uses HTTP Protocol. REST was first introduced by Roy Fielding in 2000.[13] it's built to work best on the Web. REST is an architectural style that specifies constraints, such as the uniform interface, In the REST architectural style, data and functionality are considered resources and are accessed using Uniform Resource Identifiers (URIs), typically links on the Web. The resources are acted upon by using a set of simple, well-defined operations[14] .
Some HTTP methods.

- **POST** : Used to create a new resource.
- **DELETE** : Used to remove a resource.
- **PATCH** : Used to update a existing resource.
- **GET** : Provides a read only access to a resource.

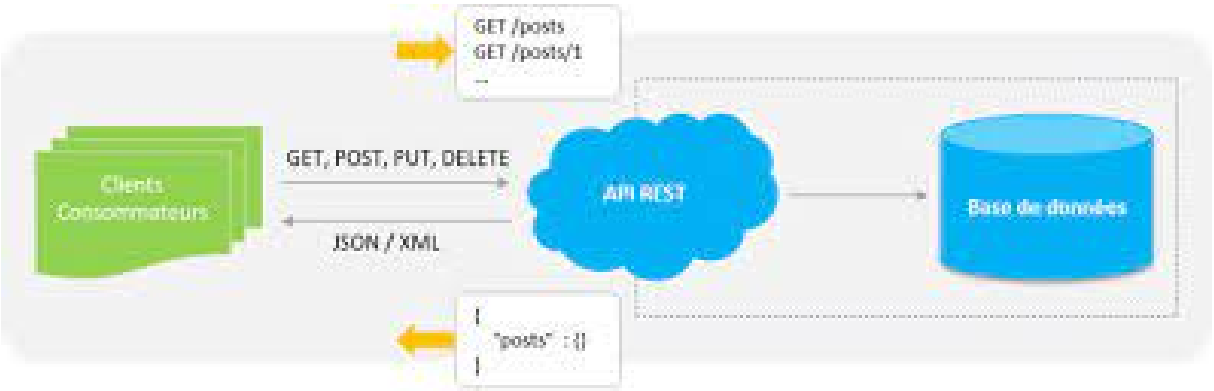

Figure 3.2: architecture of RESTful Web Services

3.1.5 Html 5

HTML is a Hypertext Markup Language, the standard language for describing the contents and appearance of Web pages.HTML5 was developed to solve compatibility problems that affect the current standard, HTML4. One of the biggest differences between HTML5 and previous versions of the standard is that older versions of HTML require proprietary plugins and APIs.[15]
(This is why a Web page that was built and tested in one browser may not load correctly in another browser.) HTML5 provides one common interface to make loading elements easier.

3.1.6 Css3

Cascading Style Sheets (CSS) is a style sheet language used for describing the look and formatting of a document written in a markup language. CSS3 is a latest standard of css earlier versions(CSS2)[17]. CSS3 makes changes to how some visual elements are implemented and rendered by a browser. However, it is not a single hugely unwieldy specification, unlike CSS2. CSS3 is separated into separate modules to facilitate development. This means that the specification comes out in chunks, with more stable modules than others.[16]

3.1.7 JavaScript

JavaScript is a programming language used on web pages. The language has was created in 1995 by Brendan Eich on behalf of Netscape Communications Corporation, Execution code is performed by your internet browsers such as Firefox or Internet Explore.[18] The peculiarity of JavaScript is to create small scripts on a page HTML for the purpose of adding a small animation or a particular effect on the page. The advantage of JavaScript is to execute code without having to reload again the page. But now JavaScript is no longer just a simple interaction language, but rather it has been developed to work as the basic language for server, mobile, and desktop applications by upgrading it by adding frameworks and an operating environment.

3.1.8 React Js

React Js is a free and open-source front-end JavaScript library for building user interfaces, it's used to build single-page applications,[19] and allows us to create reusable UI components, and One of the most important features and the reason who made it appear is the ability to re-update without rotating the browser, React can be used as a base in the development of mobile applications also.it developed by Facebook and It is currently one of the most popular JavaScript libraries and has a strong foundation and large community behind it, there are many apps and companies that use react as the main tool for building front-end like Facebook, Twitter ...[20]

3.1.9 Material Ui

Material-UI is an open-source project that features React components that implement Google's Material Design[21].
Material-UI is one of the top user interface libraries for React out there, uses more grid-based layouts, responsive animations, and transitions, padding, and depth effects such as lighting and shadows, Its success didn't come without challenges though. Designed with LESS, Material-UI v0.x was prone to common CSS pitfalls, such as global scope, which lead the project on the CSS-in-JS trajectory[**materialuiwiki**].

3.2 How the program works and how to organize it

The method of writing the code was regulated using one of the most famous design patterns that is MVC (model-view-controller)

3.2.1 MVC or (model-view-controller)

is an architectural pattern that separates an application into three main logical components: the **model**, the **view**, and the **controller**[22]. Each of these components are built to handle specific development aspects of an application.

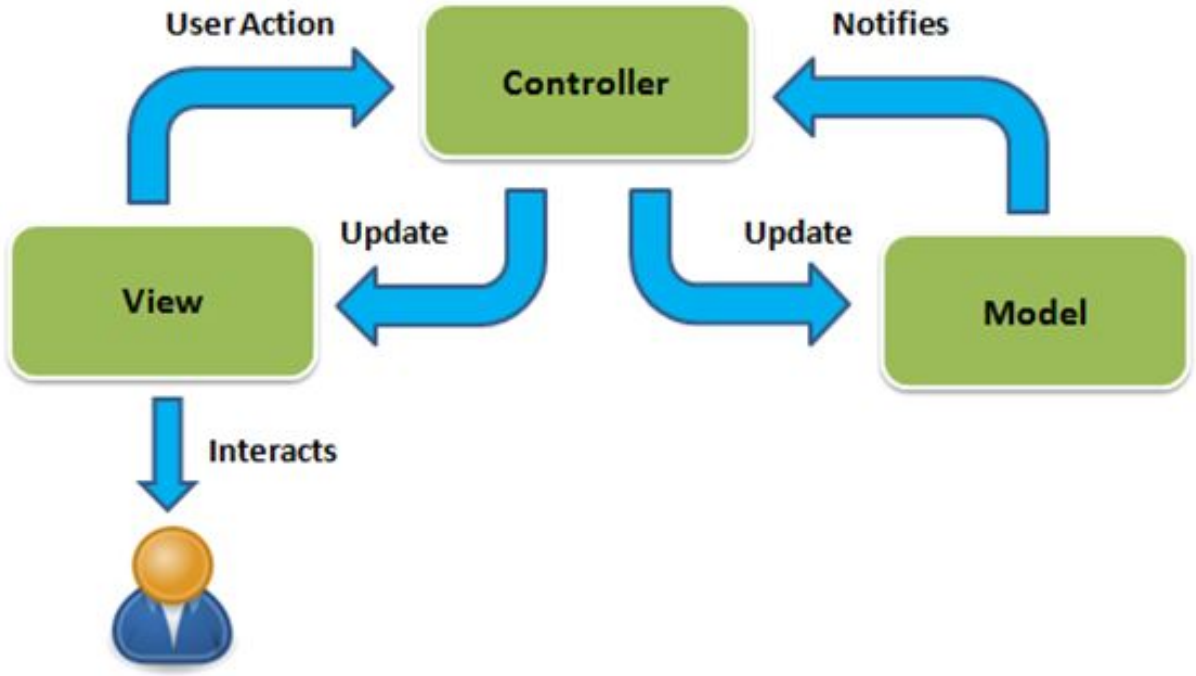

Figure 3.3: architecture of MVC

3.2.2 MVC Components

Customize the components according to the program that has been taken :

3.2.2.1 model

The model component corresponds to every schema related to the data that the user is working with and here MongoDB 'mongoose' schemas correspond with the Model of component MVC .

3.2.2.2 View

The View component is used for all the UI logic of the application which represent React Js .

3.2.2.3 Controller

Controllers act as an interface between Model and View components to process all the business logic and incoming requests, manipulate data using the Model component and interact with the Views to render the final output. Represent RestFull api .

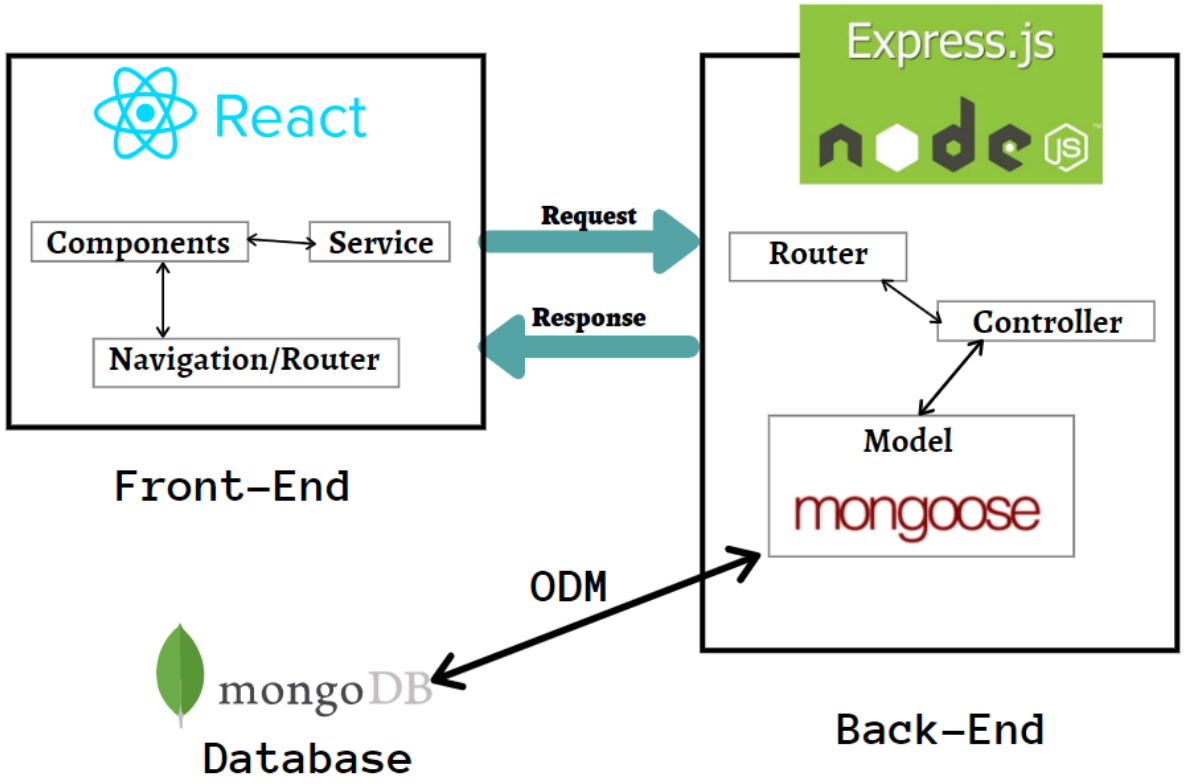

Figure 3.4: architecture of MVC with mern stack

3.3 IDE vs code

Vs code is an extensible code editor developed by Microsoft for Windows, Linux, and macOS2, its features include support for debugging, syntax highlighting, smart code completion, snippets, code refactoring, and built-in Git. Users can change the theme, keyboard shortcuts, preferences, and install extensions that add additional functionality.
visual Studio Code was ranked as the most popular development environment tool, with 50.7% of 87,317 respondents reporting using[23].

3.4 Graphics Editor Figma

Figma is a vector graphics editor and prototyping tool. It is mostly web-based, with additional offline features enabled by desktop apps for macOS and Windows, and Android and iOS allow you to view Figma prototypes on mobile devices. Figma's feature set is focused on use in UI and user experience design, with a focus on real-time collaboration.Dylan Field and Evan Wallace began working on Figma in 2012[24].

3.5 Description of the application

3.5.1 Home Page

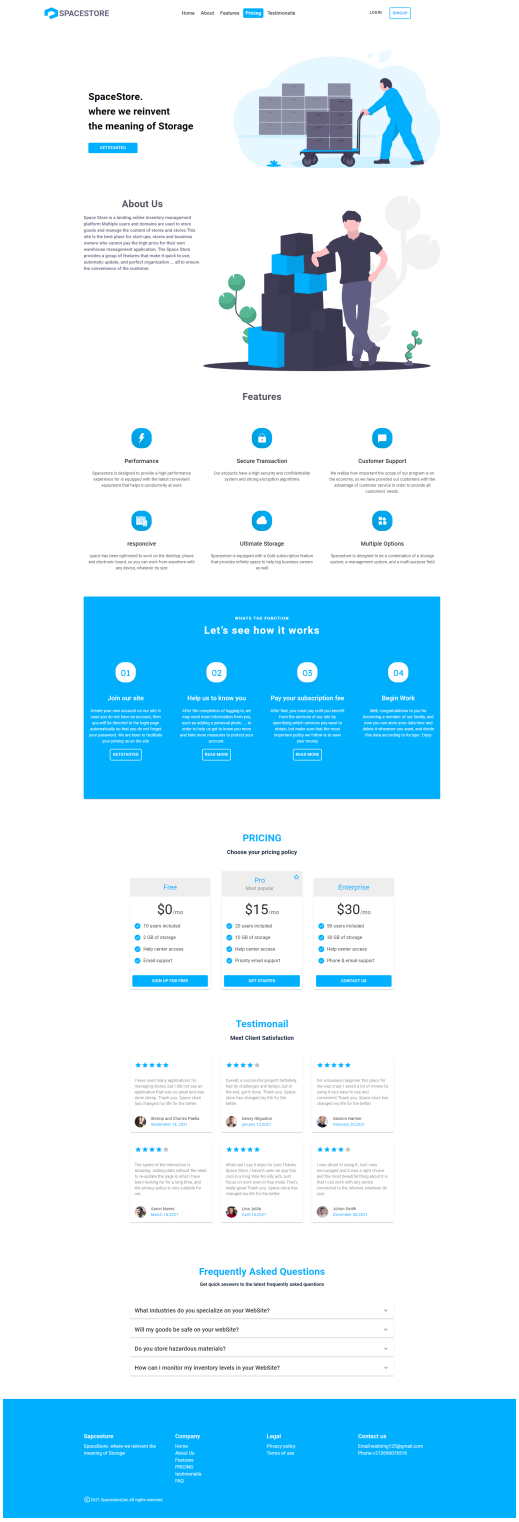

Figure 3.5: Home Page

3.5.2 Signup Page

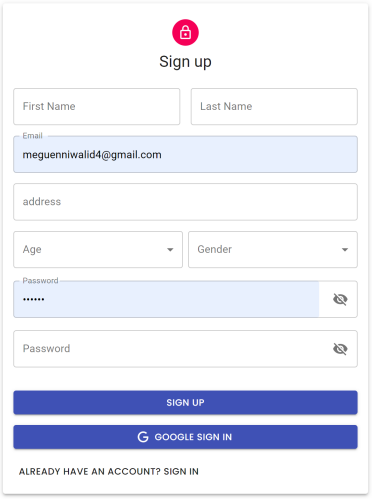

Figure 3.6: Sigup Page

3.5.3 Login Page

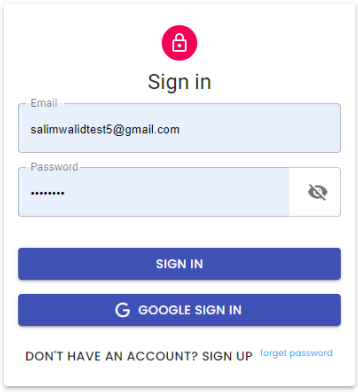

Figure 3.7: Login Page

3.5.4 Welcome

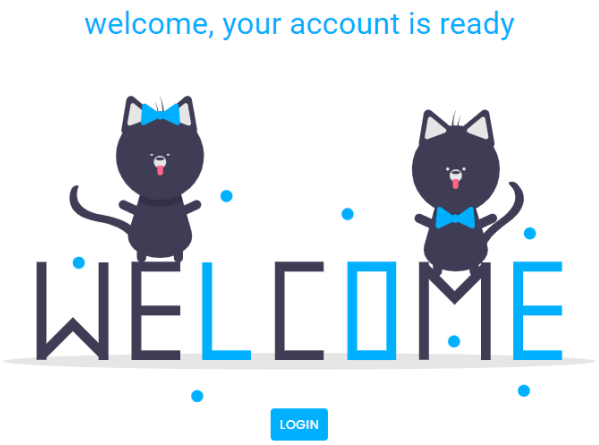

Figure 3.8: Welcome Page

3.5.5 Check Email

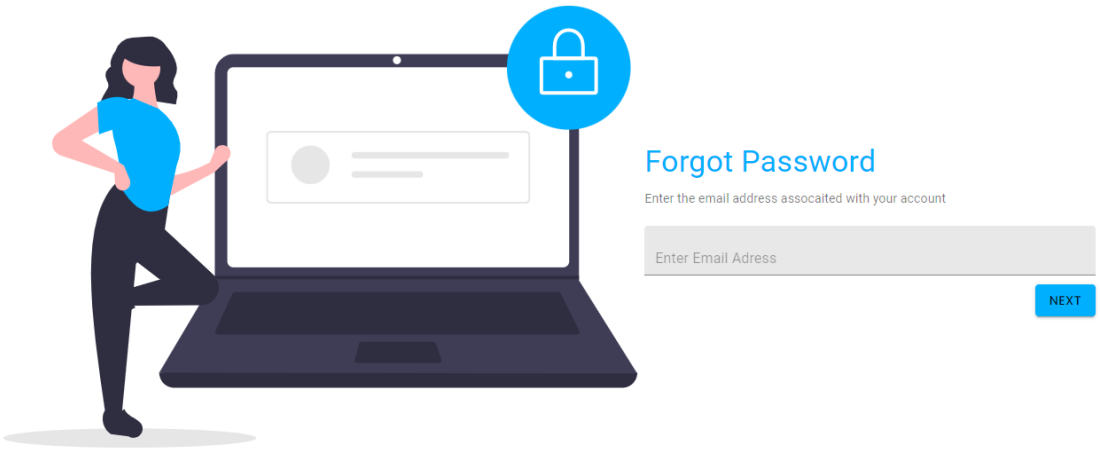

Figure 3.9: Check Email

3.5.6 Change Password

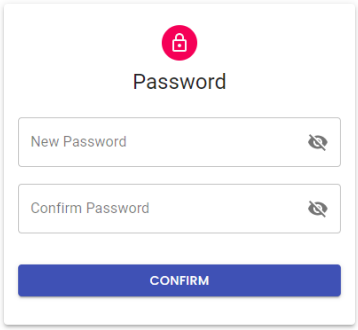

Figure 3.10: Change Password

3.5.7 Dashboard Page

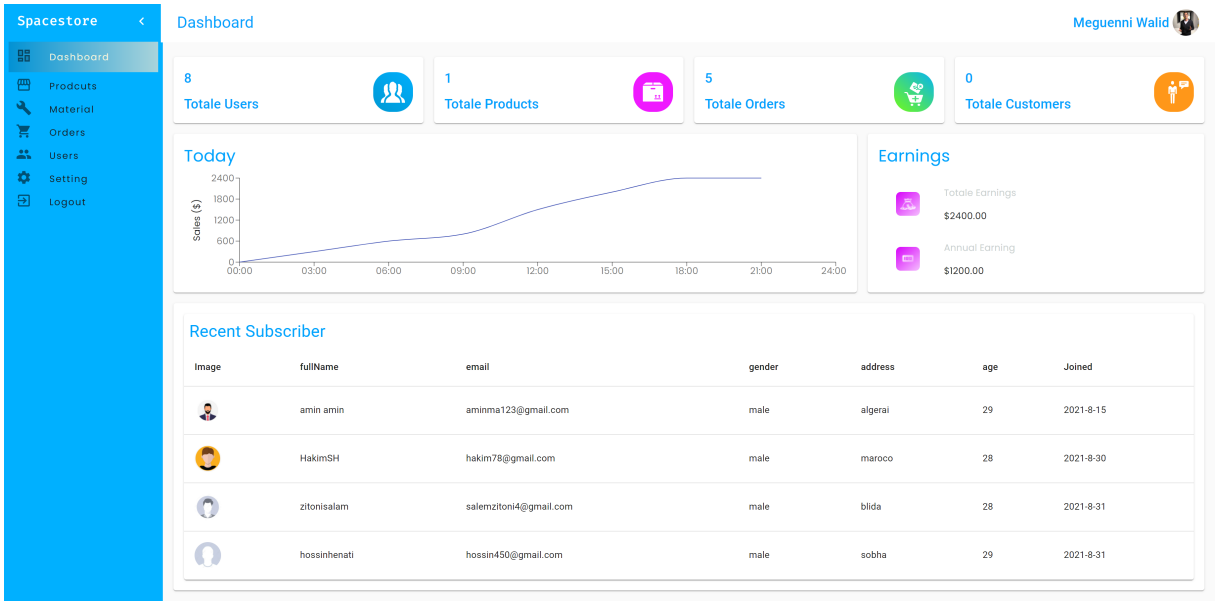

Figure 3.11: Dashboard Page

3.5.8 Products Page

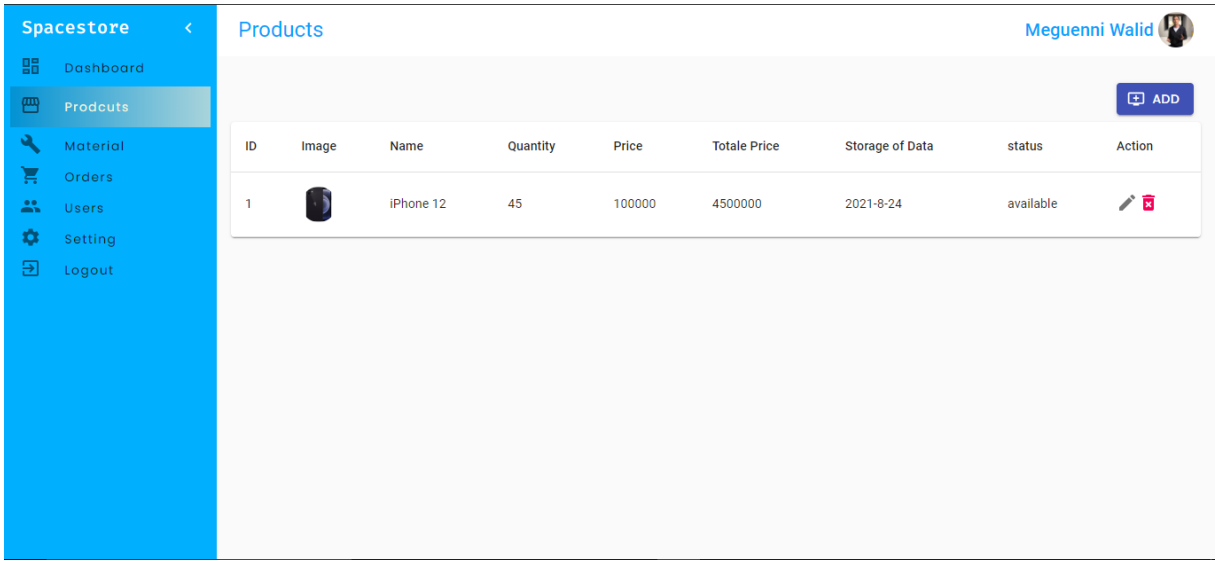

Figure 3.12: Products Page

3.5.9 Materials Page

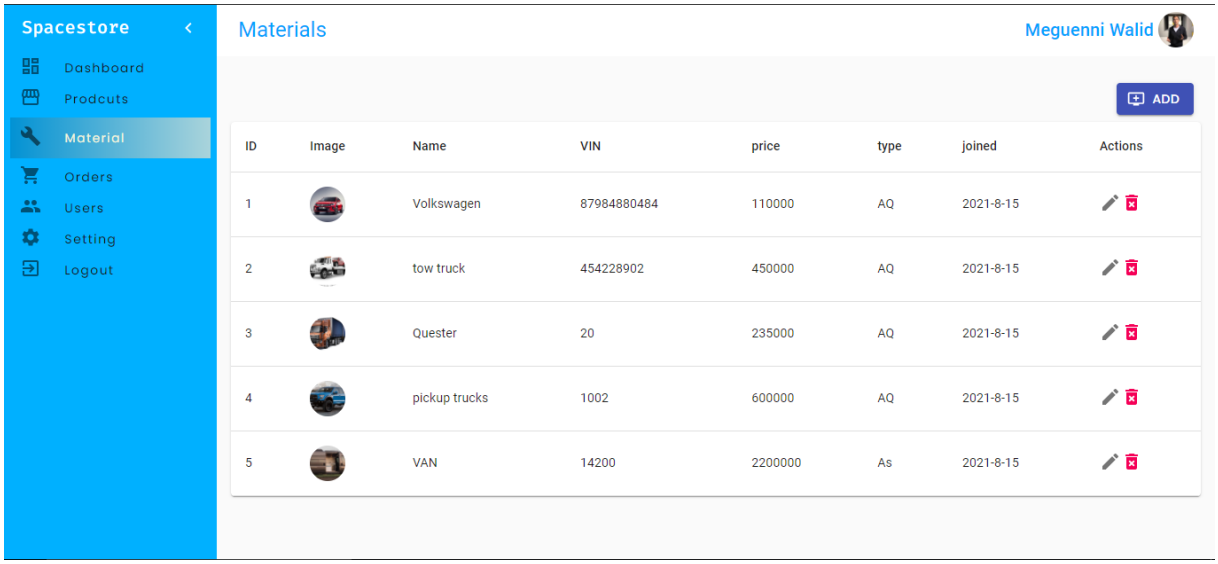

Figure 3.13: Materials Page

3.5.10 Orders Page

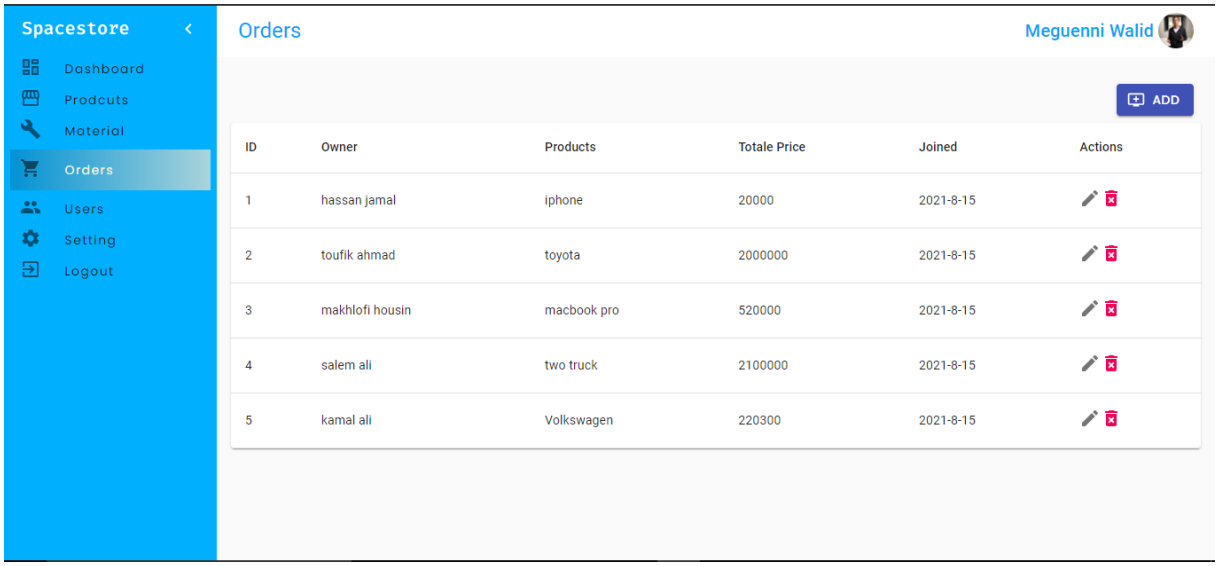

Figure 3.14: Orders Page

3.5.11 User Page admin

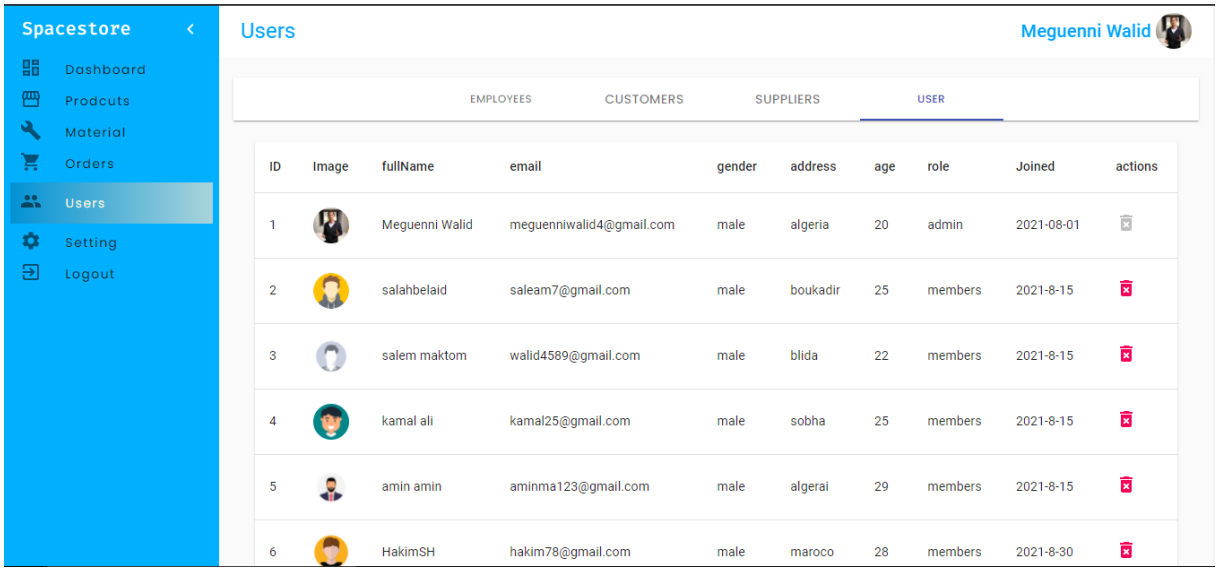

Figure 3.15: User Page admin

3.5.12 User Page members

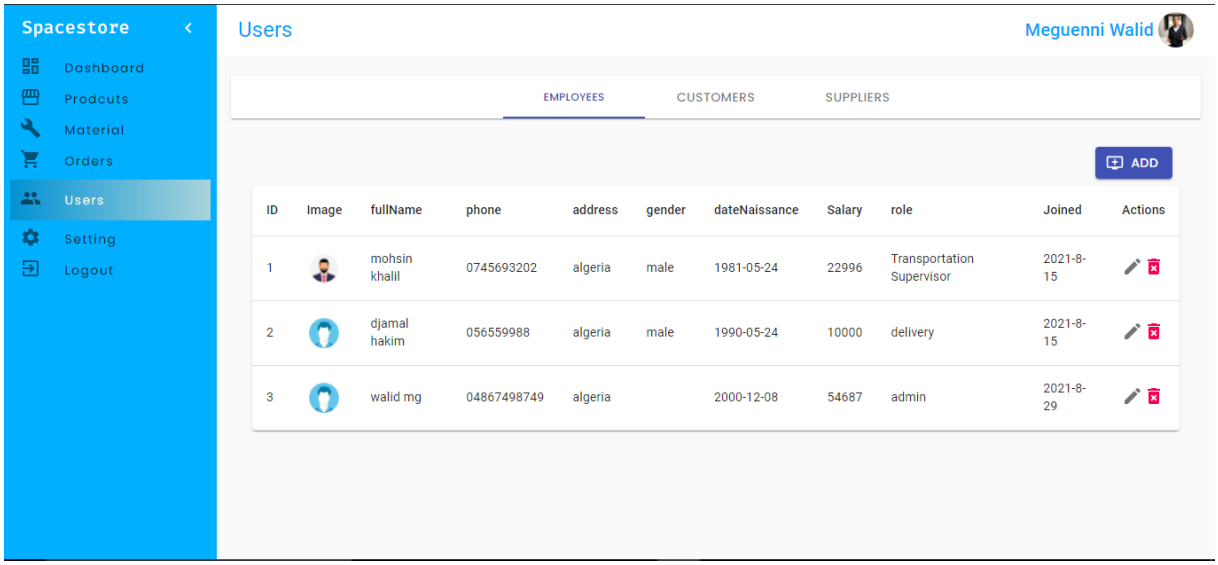

Figure 3.16: User Page members

3.5.13 Setting Page

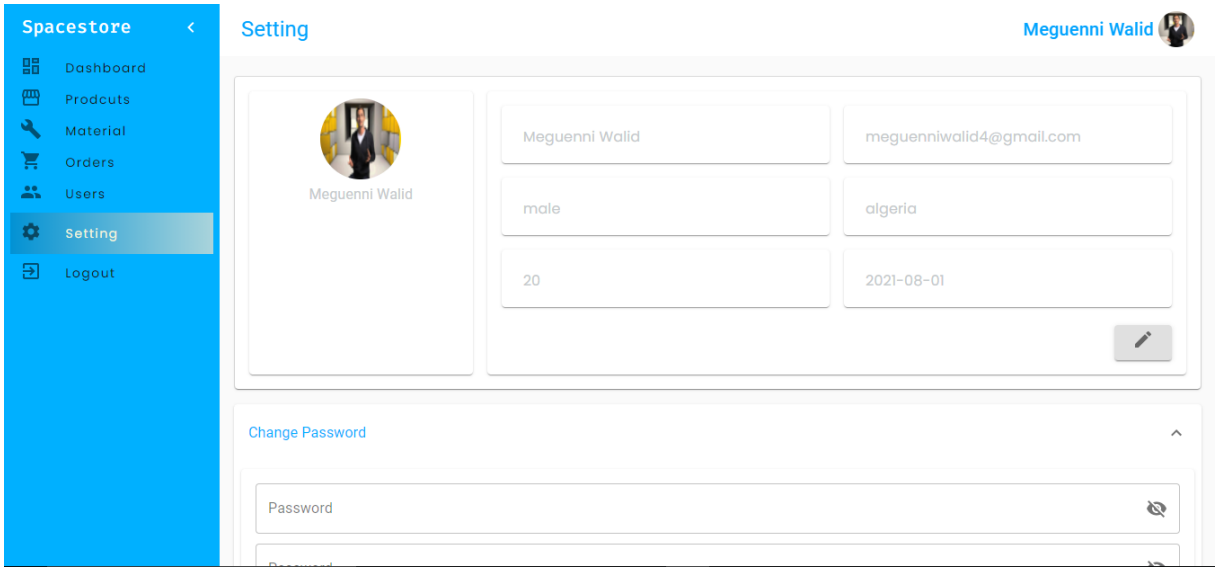

Figure 3.17: Setting Page

3.5.14 404 Not found Page

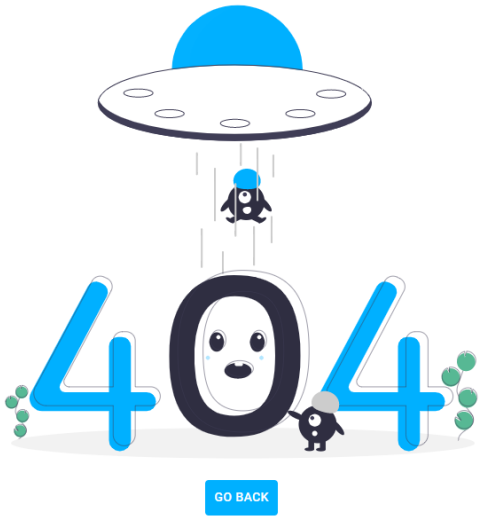

Figure 3.18: 404 Not found Page

Conclusion

As a general summary, we briefly say that the work is located in the field of management where we were completely immersed in the world of the World Wide Web, throughout this work we have described what is the management of the Inventory in general and the management of Inventory electronic filing in particular, we have identified a problem for it Update on the utility of electronic filing management, eliminate the paper and reduce human errors by increasing the effectiveness of the system through the so-called Validation.
This project has adopted a solution to facilitate Storage and verify the data and make it easier for new beginners by saving money.
Our main goal was to develop a web platform for simplification And make this service convenient. Our platform has been modified to It is convenient and easy to use and they are all recognized qualities.
To achieve the application, we used MERN Stack which means Mongo DB Express JS React JS Node JS. Express JS To carry out the various treatments and functions of the application In addition to React JS and ..... to realize their appearance graphics, while not forgetting MongoDB as a database management system. and mongo atlas as cloud for deploying our data.
This is all in addition to following one of the famous design patterns MVC to made code clear.

Finally, this study gives us a vision about the future of technology and the user of the field of the inventory management system and its impact to facilitate daily life for everyone with its appropriate system. This gave us the hypothesis or the possibility of creating a mobile application used by suppliers, managers ... and other stakeholders of all categories in other words make this system multi-platform

Bibliography

[1] 1/NicolePontius. *Inventory Management System*. URL: https://www.camcode.com/asset-tags/what-is-an-inventory-management-system/.

[2] 12/wikipedia. *Unified Modeling Language*. URL: https://en.wikipedia.org/wiki/Unified_Modeling_Language.

[3] 2/wikipedia. *Use case*. URL: https://en.wikipedia.org/wiki/Use_case.

[4] 3/tutorialspoint. *UML - Use Case Diagrams*. URL: https://www.tutorialspoint.com/uml/uml_use_case_diagram.htm.

[5] 4/Wikipedia. *Class diagram*. URL: https://en.wikipedia.org/wiki/Class_diagram.

[6] 5/Wikipedia. *Sequence diagram*. URL: https://en.wikipedia.org/wiki/Sequence_diagram.

[7] 6/wikipedia. *Node js*. URL: https://en.wikipedia.org/wiki/Node.js.

[8] 7/tutorialspoint. *Node.js - Intro*. URL: https://www.tutorialspoint.com/nodejs/nodejs_introduction.htm.

[9] 8/w3schools. *Node.js Tuto..* URL: https://www.w3schools.com/nodejs/.

[10] 9/wikipedia. *Express.js*. URL: https://en.wikipedia.org/wiki/Express.js.

[11] 90/tutorialspoint. *ExpressJS Tut..* URL: https://www.tutorialspoint.com/expressjs/index.htm.

[12] 91/wikipedia. *MongoDB*. URL: https://en.wikipedia.org/wiki/MongoDB.

[13] 92/wikipedia. *Representational state transfer*. URL: https://en.wikipedia.org/wiki/Representational_state_transfer.

[14] 93/tutorialspoint. *RESTful Web Services*. URL: https://www.tutorialspoint.com/restful/restful_introduction.htm.

[15] 931/wikipedia. *HTML5*. URL: https://en.wikipedia.org/wiki/HTML5.

[16] 932/tutorialspoint. *CSS Tuto*. URL: https://www.tutorialspoint.com/css/index.htm.

[17] 933/wikipedia. *CSS*. URL: https://en.wikipedia.org/wiki/CSS.

[18] 934/wikipedia. *JavaScript*. URL: https://en.wikipedia.org/wiki/JavaScript.

[19] 94/wikipedia. *React (JavaScript library)*. URL: https://en.wikipedia.org/wiki/React_(JavaScript_library).

[20] 95/w3schools. *React Tutorial*. URL: https://www.w3schools.com/react/.

[21] 96/freecodecamp. *Meet Material-UI*. URL: https://www.freecodecamp.org/news/meet-your-material-ui-your-new-favorite-user-interface-library-6349a1c88a8c/#:~:text=In%20a%20nutshell%2C%20Material%2DUI,that%20implement%20Google%27s%20Material%20Design.&text=With%20over%2035%2C000%20stars%20on,libraries%20for%20React%20out%20there..

[22] 97/tutorialspoint. *MVC Framework*. URL: https://www.tutorialspoint.com/mvc_framework/mvc_framework_introduction.htm.

[23] 98/wikipedia. *Visual Studio Code*. URL: https://wikipedia.org/wiki/Visual_Studio_Code.

[24] 99/wikipedia. *Figma (software)*. URL: https://en.wikipedia.org/wiki/Figma_(software).