Hawassa University

Institute of Technology

Faculty of Electrical Engineering

Department of Electrical and Computer Engineering

Computer Stream

Hosting Company: Sun Computer Engineering plc

Internship Report

By: Matios Sisay

Duration of Internship Stay: 01 Nov 2021 – 18 Feb 2022

University Mentor Name: Mr Nigatu Amanuel

Submission Date: 21 Feb 2022

DECLARATION

I hereby declare that this internship report is my original work, and that all materials utilized in the report from other sources have been properly acknowledged. The report was written in accordance with the university's internship report criteria.

STUDENT NAME: - MATIOS SISAY

SIGNATURE:-____________

DATE:-________________

MENTOR'S APPROVAL NAME: - MR NIGATU AMANUEL

SIGNATURE:-__________________

DATE:-_______________________

ACKNOWLEDGEMENTS

I would like to thank Hawassa University for giving us this internship experience program from the faculty of electrical and computer engineering. I would like to thank Sun Computer Engineering PLC for accepting me with a welcoming and respectful approach and for allowing me to work there. There were some persons who were on my side without them successful completion of my work would be impossible my supervisor Engineer Abiy Tesfaye and Engineer Eyuel I want to thank for guiding and helping me with every move I made. I would also like to thank my advisor in Hawassa University, Instructor Mr Nigatu or his guidance

EXECUTIVE SUMMARY

The department of Electrical and computer engineering had provided a four months internship program for us to work in the real working environment beside the theoretical knowledge supplied to us. This report summarizes the overall experience and different skills that I developed when I was working as an intern. In the beginning of this report I tried to explain the background of the company that I have been working in, services, the organizational structure and different qualities. Latter on I tried to describe the section that I have been working and skills that I developed during my stay with detail documentation about the project I implemented in the company. At last I finalized this report by putting my conclusion and recommendation to the company and to the university.

LIST OF FIGURES

Fig. No **Title** **page No**

LIST OF ACRONYMS

UI	User Interface
MVC	Model View Controller
API	Application Program Interface
E-R	Entity Relationship
HTML	Hyper Text Markup Language
DOM	Document Object Model
DRF	Django Rest Framework
JSX	JavaScript syntax extension
UML	Unified Modeling Language
HTTP	Hyper Text Transfer Protocol
CSS	Cascading Sheet Style

TABLE OF CONTENT

CHAPTER ONE

CHAPTER TWO

CHAPTER THREE

CHAPTER FOUR

CHAPTER FIVE

CHAPTER ONE

BACKGROUND OF SUN COMPUTER ENGINEERING PLC

1.1 HISTORICAL OVERVIEW OF THE COMPANY

Sun Computer Engineering Plc was founded in 2008 by two engineers in accordance with the aim and passion to provide their duties in the Technology industry. For the past six years, it has provided services to a variety of private and governmental organizations. During its inception, Sun Computer Engineering Plc began serving as a learning school for COC exam taker students and teaching on different tech related aspects like networking, website development, office machine maintenance, . As a consequence of its strong dedication and well-experienced professionals to the company's growth, it has now been raised to be competent in the tech industry.

1.2 OBJECTIVE, MISSION AND VISION OF THE COMPANY

Objective

The objective of Sun Computer Engineering PLC is to provide a good service for customers and to make students be effective on the desired skills by doing so increasing the company financial strength.

Mission

To provide Technological services to the highest level of client satisfaction by valuing people, building long-term relationships, being cost-effective, delivering services on time, and guaranteeing safety and reliability.

Vision

To create local opportunity, growth and impact in the community and provide services on large scale including full country wide service by building strong relationship with other companies and building firms.

1.3 SERVICES AND PRODUCTS OF THE COMPANY

Sun Computer Engineering PLC is dedicated to providing products and services in the tech industry that meet the needs of customers by implementing a quality management system that drives continuous improvement throughout the organization. The services and products provided by the company are:

- Networking
- Computer Maintenance
- Website Development
- Security Camera Installation

1.4 MAIN CUSTOMERS OR END USERS OF THE COMPANY

Sun Computer Engineering PLC has worked on a variety of tech projects all around the city. They will provide a variety of services for various bodies after the projects are completed like ongoing maintenance and upgrading. The major clients or end users of the company are:

- Schools
- Governmental organizations

1.5 ORGANIZATIONAL STRUCTURE OF THE COMPANY

The organizational structure of the company is shown below:

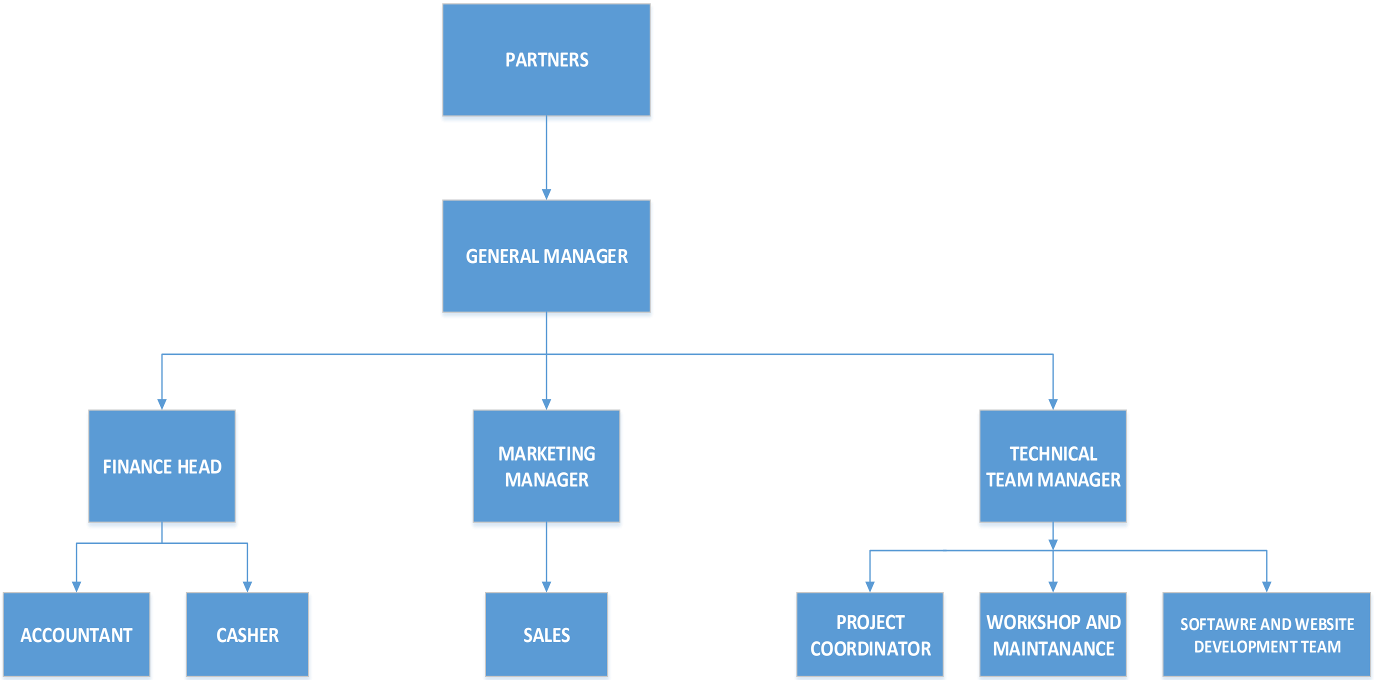


Figure 1.1 Organizational Structure

1.6 WORKFLOW OF THE COMPANY

The work flow of the company follows the hierarchy that I tried to explain on the above organizational structure. The general manager controls every aspect of the company by communicating with the finance head, marketing manager and technical manager. The finance head is responsible for managing the companies' income and expenses and the marketing manager promotes with different marketing strategy. The technical manager is responsible for the service and products that the company provides.

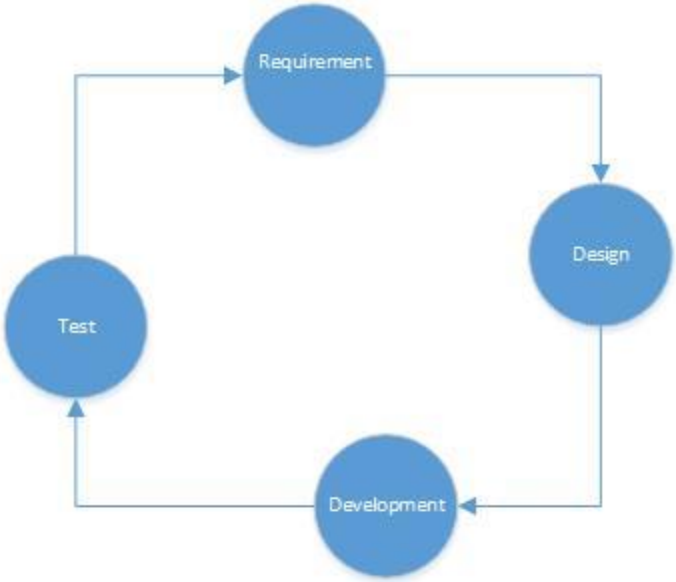


Figure 1. 2 Work Flow

CHAPTER TWO

OVERALL INTERNSHIP EXPERIENCE

2.1 HOW YOU GET INTO THE COMPANY

Before we get out for the internship we were given a detailed orientation by our lectures about how the coming internship program was planned to be and what possible challenges and problems we could face. The orientation mainly focused on helping us realize that no one could believe in us unless we were strictly responsible, ready to learn & grasp knowledge we could get from anyone.

Then using my internship application letter, I requested some companies. One of the company was Sun Computer Engineering PLC. Then I was able to contact the owner of the company Engineer Abiy Tesfaye and ask me about my interest then handed me the acceptance paper.

2.2 SECTION FOR THE COMPANY I HAVE BEEN WORKING IN

I have been working on the website development section of the company which is instructed and controlled by Engineer Abiy Tesfaye. I was able to be exposed to different website development tools and gain many valuable knowledge from videos that was given by the section leader and by being able to access helpful learning websites that are related to website development.

2.3 HOW DOES THE WORK FLOW IN THE SECTION LOOK LIKE

The work flow structure of the Sun Computer Engineering PLC web development team is easy to adapt and not complex due to lack of large team in the development section. I have been assigned to work on the project and discus with the company web developer Engineer Abiy Tesfaye after certain implementation is done or when I face difficulties performing the task and weekly discussions to check my progress on the project with further instructions.

2.4 PIECE OR WORK TASKS I HAVE BEEN EXECUTING

Downloading and installing software tools like:

Visual Studio Code: it’s a code editor and can be used in all platforms and environment.

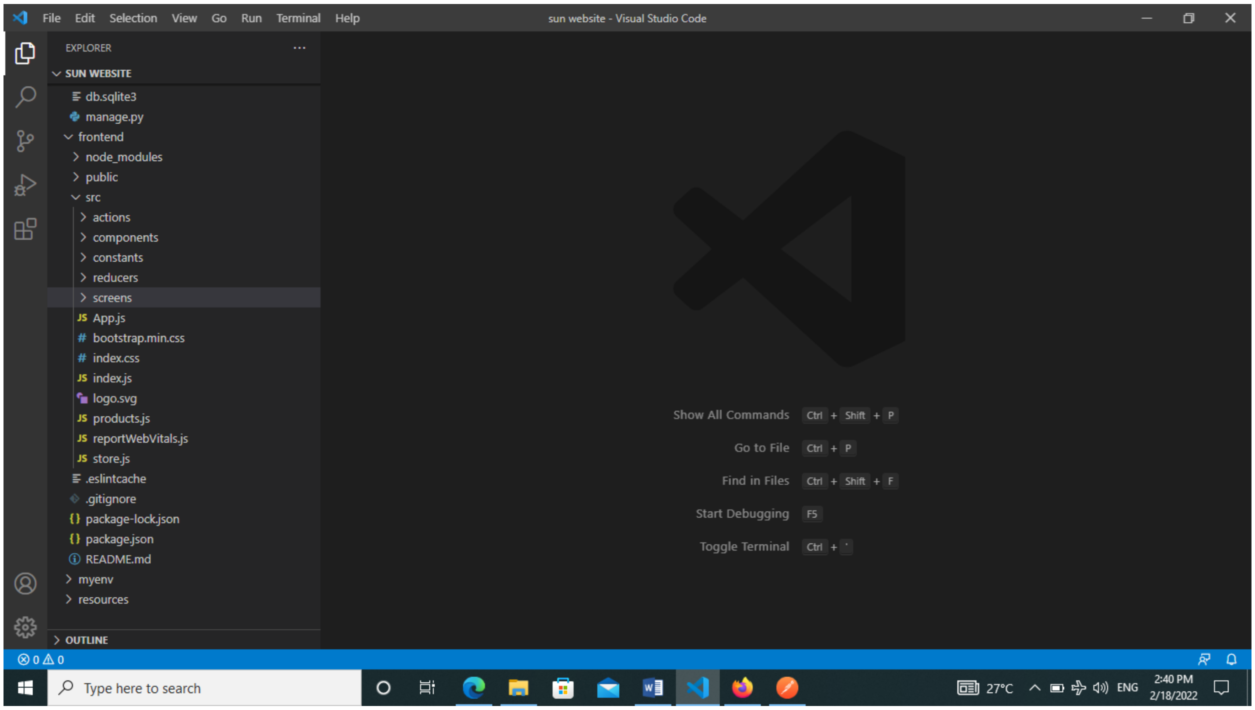
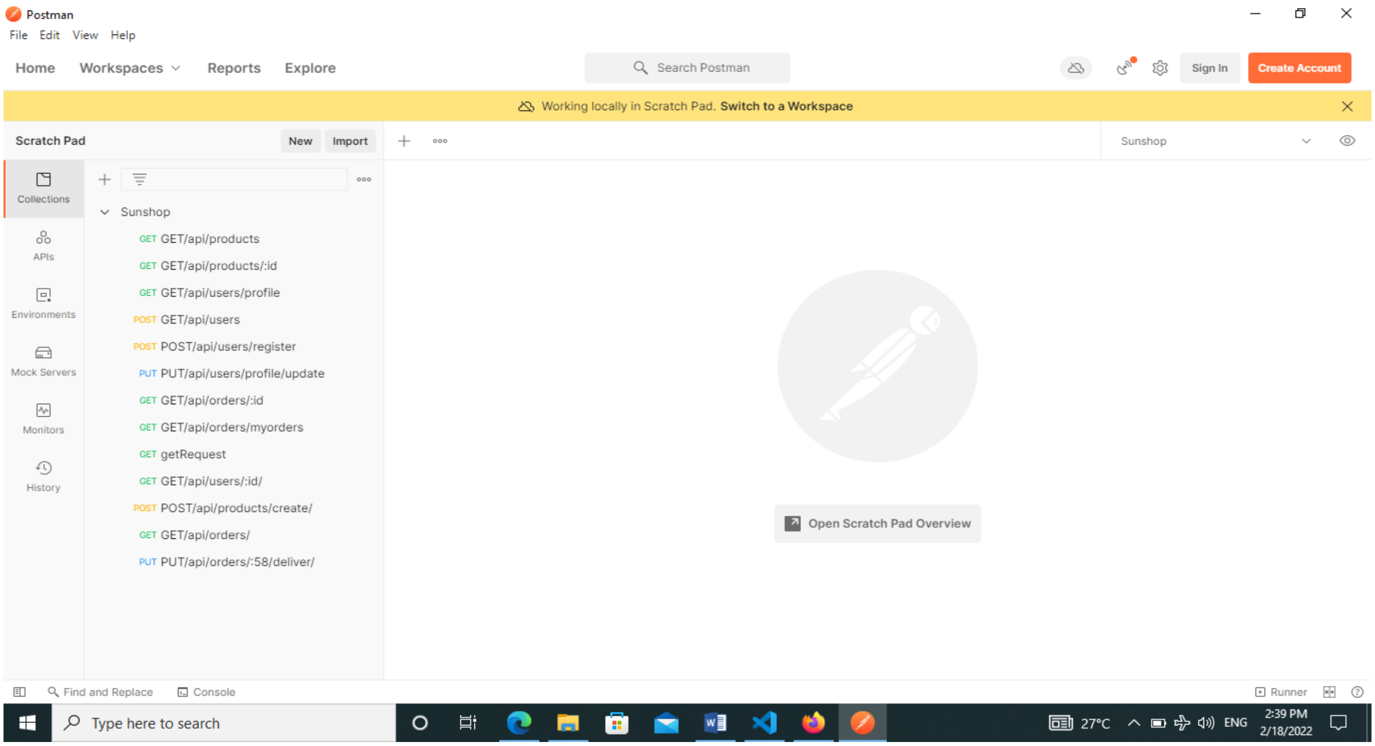


Figure 2.1 Visual Studio Code

- **Postman**: is development tool which helps to build, test and modify APIs.



Sunshop
GET GET/api/products
GET GET/api/products/:id
GET GET/api/users/profile
POST GET/api/users
POST POST/api/users/register
PUT PUT/api/users/profile/update
GET GET/api/orders/:id
GET GET/api/orders/myorders
GET getRequest
GET GET/api/users/:id/
POST POST/api/products/create/
GET GET/api/orders/
PUT PUT/api/orders/:58/deliver/

Figure 2.2 Postman

- **Node.js:** Node.js is an open-source, cross-platform, back-end JavaScript run runtime environment.
- **Python:** Python is a programming language and windows does not come with a python interpreter installed. We need to explicitly install it and add it to environmental variables.

After I set up my environment and desired tools these are the major tasks I have been working:

- ## Create React App

The project started by creating a react app called frontend and we need to install Node.JS software before we start to create the react app. After having a Node.js software in our computer we just follow some steps:

Step 1 - Install create-react-app globally with node package manager (npm).

```
npm install -g create-react-app
```

Step 2 - Then run the generator in your chosen directory.

```
create-react-app frontend
```

Step 3 - Navigate to the newly created directory and run the start script.

```
cd frontend

npm start
```

Now, we can see our react-app in local host server http://127.0.0.1.300.

▪ Adding react bootstrap

The LUX theme from bootswatch.com was used in the frontend and by adding the CSS file of the LUX theme into the source folder and importing different stylings from the theme by installing a node module using npm (node package manager):

```
npm install react-bootstrap
```

▪ Implementing react router

```
npm install react-router-dom react-router-bootstrap
```

▪ Create Django project

We need to have Python software before we start using Django, we need to create an environment for the applications. We can set the environmental variable of Windows using the following command in our root working folder.

```
Pip install virtualenv

virtualenv myenv (name of desired environment)
```

To activate the environment, we use the following command

```
myenv\scripts\activate
```

Virtualenv keeps all important files and installs all in a one environment rather than globally define.

Now we can create our Django project using

```
Pip install django

django-admin startproject backend (name of desired directory)
```

This will create a directory in your current directory with the name of the project. To use the development server, we need to use the manage.py runserver command. We must launch the command prompt and put ourselves in the project root (use the cd command to browse folders) to execute the command:

```
Python manage.py runserver
```

This command starts the Django development server.

Fetching data from backend to frontend

By using Axios node module and providing the necessary URL we can make requests to get a data from the backend. The command used to install the module is provided under:

```
npm install axios in the react folder
```

Migration

Migrations are Django’s way of propagating changes you make to your models (adding a field, deleting a model, etc.) into your database schema. The command used are:

```
python manage.py makemigrations

python manage.py migrate
```

The migrate command looks at the INSTALLED_APPS setting and creates any necessary database tables according to the database settings

Creating an admin user

First, we need to create a user who can login to the admin site by using the following command:

```
python manage.py createsuperuser
```

Enter your desired username and press enter.

```
Username: admin
```

You will then be prompted for your desired email address:

```
Email address: admin@example.com
```

The final step is to enter your password. You will be asked to enter your password twice, the second time as a confirmation of the first.

```
Password: **********

Password (again): *********

Superuser created successfully.
```

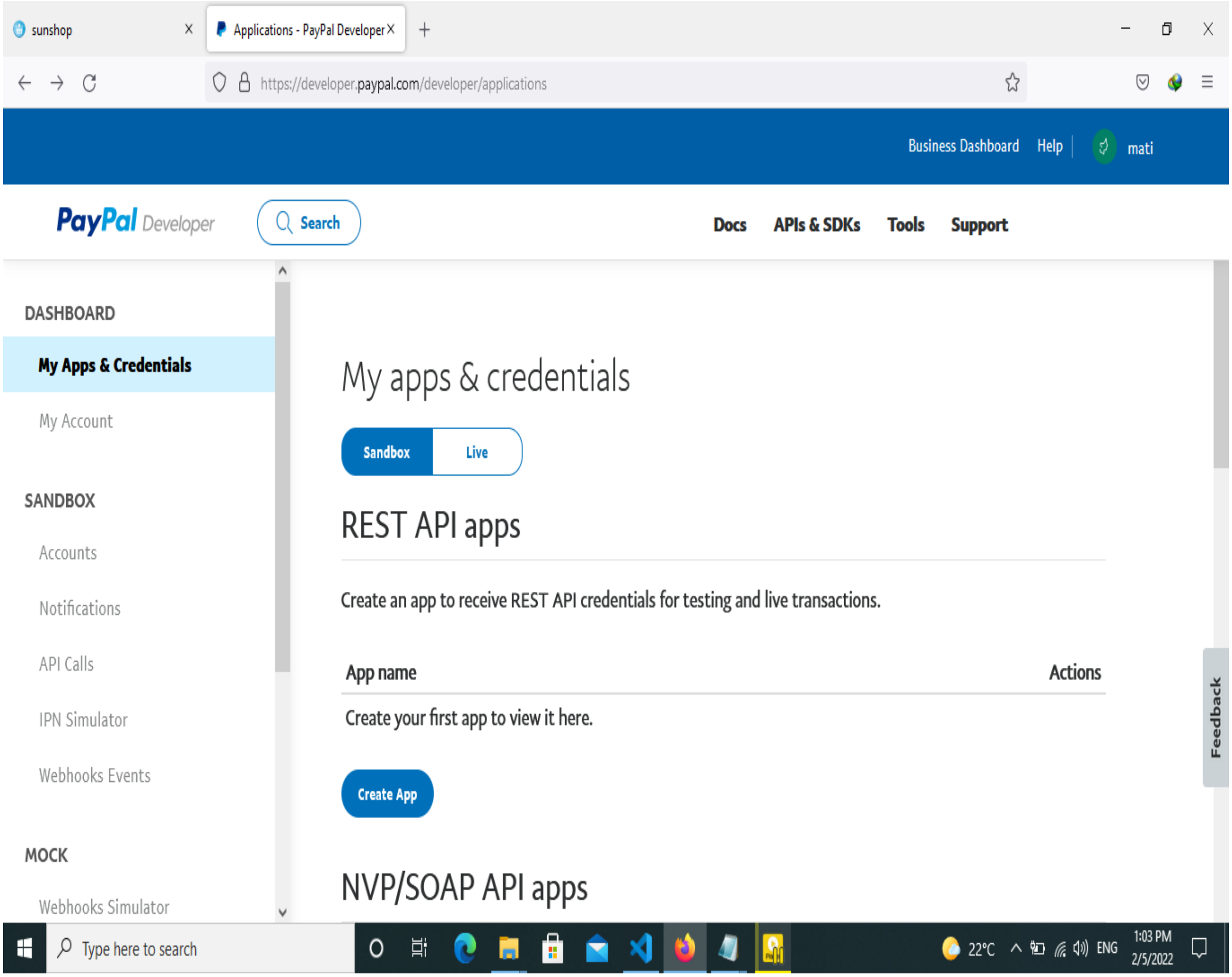
Installing Redux

```
npm install redux react-redux redux-thunk redux-devtools extension
```

Testing Online Payment System

The PayPal sandbox is a self-contained, virtual testing environment that simulates the live PayPal production environment. The sandbox provides a shielded space where you can initiate and watch while your apps process PayPal API requests without touching any live PayPal accounts.

When you initiate a transaction through a sandbox account, PayPal creates a mock transaction that behaves exactly like a transaction in the live environment. By using fictitious sandbox accounts with their associated authentication credentials in your PayPal API calls, we tested and debug the apps without referencing any real PayPal users or live PayPal accounts.



My apps & credentials

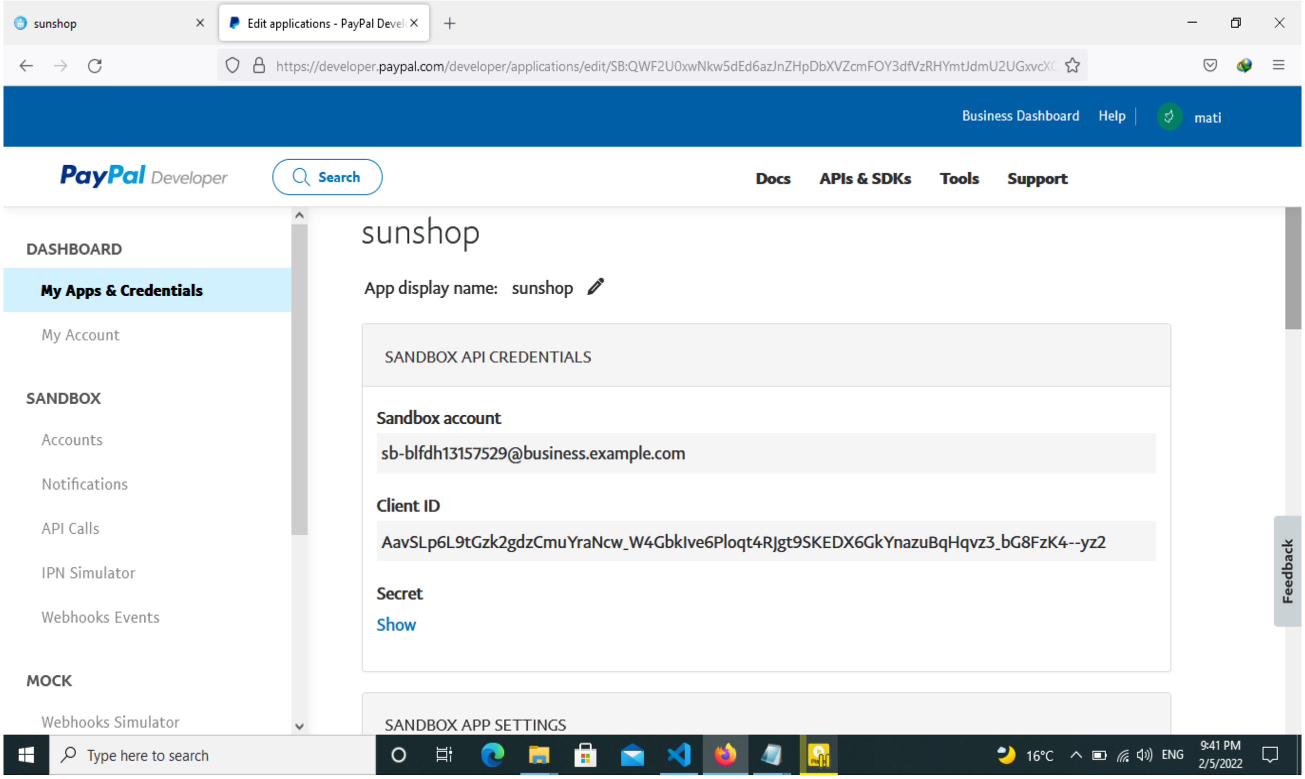
REST API apps

Create an app to receive REST API credentials for testing and live transactions.

App name	Actions
Create your first app to view it here.	

NVP/SOAP API apps

Figure 2.3 Developers PayPal home page



sunshop

App display name: sunshop

SANDBOX API CREDENTIALS

Sandbox account

sb-blfdh13157529@business.example.com

Client ID

AavSLp6L9tGzk2gdzCmuYraNcw_W4GbkIve6Ploqt4RJgt9SKEDX6GkYnazuBqHqvz3_bG8FzK4--yz2

Secret

Show

SANDBOX APP SETTINGS

Figure 2.4 Sandbox account used to test

2.5 PROCEDURES I HAVE BEEN USING

Since I started the project I have been assigned on to work needs to be effective and functioning. I used different approaches and procedures. The following procedure was used to overcome difficulty and lastly, I achieved the project in a good and best manner.

1. First I gathered information about E-commerce system and my advisor in the company helped me by providing very useful videos.
2. Then I started setting up desired environments and tools
3. After that I started the design, coding and configuring my system work flow

2.6 HOW GOOD I HAVE BEEN IN PERFORMING TASKS

First I feared that I lack practical skills since it was my first time in a work environment, as well as theoretical skill. But, when I proceeded I were able to do the project and solve the problem in a good manner be able to make the company satisfied.

2.7 CHALLENGES I HAVE BEEN FACING

Challenges I faced while performing tasks are:

- Being new for the working area
- Implementing new programing language that I have never used before

2.8 MEASURES I HAVE TAKEN

In order to overcome the challenge that I face these are the important measures I have taken.

- Using different learning website and watching YouTube videos
- Asking for help to my advisor in the company

CHAPTER THREE

OVERALL BENEFITS GAINED FROM THE INTERNSHIP

3.1 IMPROVING PRACTICAL SKILLS

During the four-month internship program, I gained a great deal of practical experience. I can now apply realistic methods to complete any activity. I can give some examples of the many practical skills I have gained:

- Proper installation and preparing required environment.
- Working with different web development tools.
- Adding helpful extensions that makes my work easier.
- Discovering different learning websites which I used to upgrade my knowledge and also to solve different errors.
- Being able to have a good work ethics.
- Apply class room theoretical part to real work situations.

3.2 UPGRADING THEORETICAL KNOWLEDGE

The internship program is quite beneficial in terms of theoretical knowledge growth. Beyond the courses I took during my four-year Electrical and Computer engineering school, I've considered a lot of things. In addition, I use the courses to improve my practical skills for my work tasks. The following are some areas where my theoretical understanding has increased.

- Learn more about various courses especially programming related.
- Become more knowledgeable about general work functions of web development process and how to test before going live.
- Gain an increased awareness about how to be full stack developer.

3.3 IMPROVING INTERPERSONAL COMMUNICATION SKILLS

Interpersonal communication refers to how a person communicates with other persons with whom he or she may come into contact. Most of my contacts was with Engineer Abiy Tesfaye who is the web developer of the company and I was able to have a good communication with him and develop a better skill to communicate with people in the work space.

3.4 IMPROVING YOUR TEAM PLAYING SKILLS

It is a person's ability to work cooperatively with others. This was a simple and quick skill to learn. It all started when I was on campus. Some of the assignments were completed in groups. This enables me to deal with a variety of personalities among the people I've worked with. I improved my team work qualities during my internship time through:

- Showing genuine commitment.
- Being patient on different situations.
- Being reliable and responsible.
- Giving respect for other

3.5 IMPROVING YOUR LEADERSHIP SKILLS

The internship program is an important occasion to practice the assets of leadership. I have learned a lot about leadership and I saw how important it is for a successful project. Generally leadership skills prove the following core point:

- Commitment and Passion.
- Accountability. Taking responsibility when something happen
- Good Communicator.

Generally during the time of the internship period, I take risk by being leader of the project and from this I obtain a good leadership experience.

3.6 UNDERSTANDING ABOUT WORK ETHICS RELATED ISSUES

The benefit of the internship program was interesting in the development of ethical work habit which include understanding and applying rules and regulation of the company. Ethical conducts of workers is very important for developing healthy working environment and increasing productivity. Accountability, Character building, Teamwork, Appearance, Productivity, Organizational Skills, Communication Skills, Cooperation, and Respect are the basic qualities of a person having a good work ethics and I think I have been able to achieve those qualities in my internship experience in the company.

3.7 ENTREPRENEURSHIP SKILLS

An entrepreneur is a positive thinker who can see solutions from surrounding cloud of problems. An entrepreneur is also able to take risk of the outcome of the things he/she does. Since I already recognized that my internship was more about building professional skills than gaining content knowledge, I took my time to learn about the company, not just the tasks at hand. I also used to communicate and actively engage with management and leadership without sounding academic or overly formal which helped me to learn a lot on improving critical thinking and problem-solving skills. One of the vision and mission of Sun Computer Engineering PLC is to motivate and help the entrepreneurs those have the idea and need to join the market. These experiences are unforgettable and I believe that it shaped my personality and behavior.

CHAPTER FOUR

PROJECT: E-COMMERCE SYSTEM

4.1 INTRODUCTION

E-commerce originated in a standard for the exchange of business documents, such as orders or invoices, between suppliers and their business customers. Electronic Commerce involves selling and buying services and goods via the internet using electronic device like phone and personal computers, it took much faster and closer to connect supplier and the one with customers need. This project attempts to examine several problems related with the distribution channel and at the end of the paperwork and manual purchasing instead it adopts online ordering products. The web-based system has two main components: frontend and backend. In frontend, react + redux using JavaScript was implemented. While in the backend a python framework called Django was used. The frontend build how a website looks and holds our user view data and helps us to make a desirable website. The backend development uses backend programming languages to fulfill those requests on the server side. It build how a website works. The backend work on managing the database, as well as the site's users, security, and site performance issues. There are many different hardware and software tools that are used to develop websites in accordance of the programming language, desired task and personal preference. The only hardware tool used to develop this project is personal computer and the software tools include Visual Studio Code, Postman, Node.js and Python.

4.2 STATEMENT OF PROBLEM

The existing system in the company is manual way of addressing customers and providing products information's via cellphone and text messages which is affecting the business of the company in economical and marketing aspects these are:

- Not being competent in the electronics market due to many e-commerce website developed in our country.
- Providing less detailed information's to the customers
- Improper stock management.
- Different unnecessary cost for phone bills and marketing purpose.
- Making the company a little bit of low profile in the eyes of the customers.

4.3 OBJECTIVE

4.3.1 General Objective

The overall objective of this project is to establish a system for online product distribution and to improve the performance and efficiency of marketing management in order to overcome the intermediaries between the customers and the company.

4.3.2 Specific Objectives

In order to achieve the general objective, the following specific objectives are identified.

- Providing online ordering system.
- To provide information about various products in different category with price.
- To minimize human error
- To easily provide accessibility of stock report
- To easily provide accessibility of sales report
- To easily provide accessibility of customer management
- To test payment integration with sandbox account before deployment

4.4 SCOPE

This project aim to provide online product distribution system with information about Products and give access for registration of new user, login functionality for registered User, updating profile and access to online ordering system.

Limitation of the project include sandbox account was used to test online payment Integration.

4.5 METHODOLOGY

Methodology specifies the method and technology used to develop the software system such as, the methods used to gather data, approach used to design the software system, software and hardware requirements used to implement the system. The main goal of this study is to develop an e-commerce system by using Agile software development method and by using software frameworks like Django, Django rest framework, react and redux.

Agile software development method

Agile software development refers to software development methodologies centered on the idea of iterative development. The ultimate value in agile development is that is enables teams to deliver value faster, with greater quality and predictability and greater aptitude to respond to change. The benefits of agile are:

- Highly responsive to customers development requests
- Faster time to market
- Project visibility and transparency
- Risk reduction

Agile helps teams deliver high quality software on time and on budget.

Django

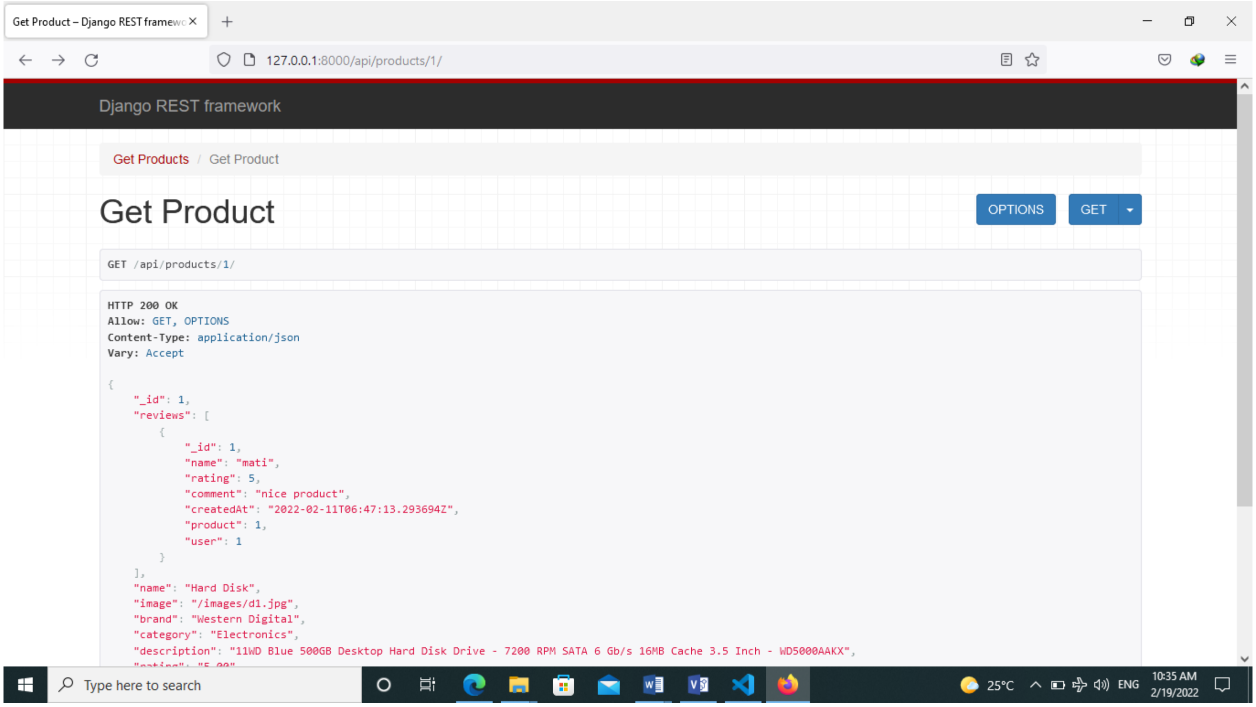
Django was born in 2003 in a press agency of Lawrence, Kansas. It is a web framework that uses Python to create websites. Its goal is to write very fast dynamic websites. This framework was created to accelerate the development phase of a site, but not exclusively. Indeed, this framework uses the MVC pattern, which enables us to have a coherent architecture.

For clarity, a dynamic Web site is one in which pages aren't simply HTML documents sitting on a server's file system somewhere. In a dynamic Web site, rather, each page is generated by a computer program a so-called "Web application" that the Web developer, create. A Web application may, for instance, retrieve records from a database or take some action based on user input.

A framework is a set of software that organizes the architecture of an application and makes a developer's job easier. A framework can be adapted to different uses. It also gives practical tools to make a programmer's job faster. Thus, some features that are regularly used on a website can be automated, such as database administration and user management. Once a programmer handles a framework, it greatly improves their productivity and the code quality.

Django Rest Frame Work

Django rest frame work is a powerful and flexible toolkit for building web API. The biggest reason to use Django rest framework is because it makes serialization so easy. Serialization in Django rest framework are responsible for converting objects into data types understandable by JavaScript and frontend frameworks. Django rest frame work provides a good user interface to see our data.


Get Product – Django REST framew
127.0.0.1:8000/api/products/1/
Django REST framework
Get Products / Get Product
Get Product
OPTIONS
GET
Type here to search
25°C
ENG
10:35 AM
2/19/2022

```
GET /api/products/1/

HTTP 200 OK
Allow: GET, OPTIONS
Content-Type: application/json
Vary: Accept

{
    "_id": 1,
    "reviews": [
        {
            "_id": 1,
            "name": "mati",
            "rating": 5,
            "comment": "nice product",
            "createdAt": "2022-02-11T06:47:13.293694Z",
            "product": 1,
            "user": 1
        }
    ],
    "name": "Hard Disk",
    "image": "/images/d1.jpg",
    "brand": "Western Digital",
    "category": "Electronics",
    "description": "11WD Blue 500GB Desktop Hard Disk Drive - 7200 RPM SATA 6 Gb/s 16MB Cache 3.5 Inch - WD5000AAKX",
```

Figure 4.1 Django REST frame work UI

The MVC framework

Before the MVC framework existed, web programming mixed the database access code and the main code of the page. This returned an HTML page to the user. Even if we are storing CSS and JavaScript files in external files, server-side language codes are stored in one file that is shared between at least three languages: Python, SQL, and HTML.

The MVC pattern was created to separate logic from representation and have an internal architecture that is more tangible and real. The Model-View-Controller (MVC) represents the three application layers that the paradigm recommends:

- **Models**: These represent data organization in a database. In simple words, we can say that each model defines a table in the database and the relations between other models.
- **Views**: These contain all the information that will be sent to the client. They make views that the final HTML document will generate. We can associate the HTML code with the views.
- **Controllers**: These contain all the actions performed by the server and are not visible to the client. The controller checks whether the user is authenticated, or it can generate the HTML code from a template

Database Layer: The models are stored in database tables in SQLite.

A key advantage of such an approach is that components are loosely coupled. Each distinct piece of a Django-powered Web application has a single key purpose and can be changed independently without affecting the other pieces. For example, a developer can change the URL for a given part of the application without affecting the underlying implementation. A designer can change a page's HTML without having to touch the Python code that renders it. A database administrator can rename a database table and specify the change in a single place rather than having to search and replace through a dozen files.

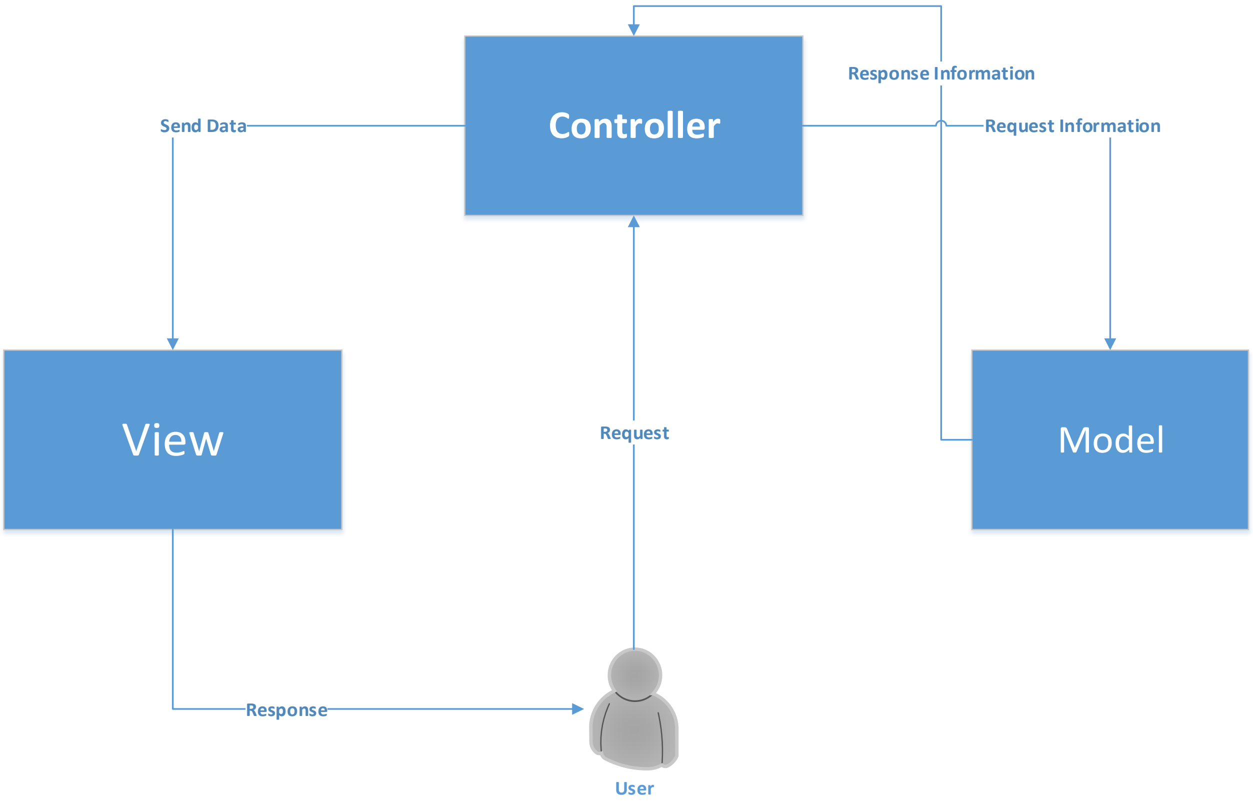


Figure 4. 2 MVC model

The following are the steps that are followed in an application with the MVC pattern:

1. The client sends a request to the server asking to display a page.
2. The controller uses a database through models. It can create, read, update, or delete any record or apply any logic to the retrieved data.
3. The model sends data from the database to the controller; for example, it sends a product list if we have an online shop.
4. The controller injects data into a view to generate it.
5. The view returns its content depending on the data given by the controller to the user.

Why Django

The following is a non-exhaustive list of the advantages of using Django:

- Django is supported by a good community. This is a very important asset because it allows you to resolve issues and fix bugs very fast.
- Django is fully customizable. Developers can adapt to it easily by creating modules or overridden framework methods.
- Scalability which means the capacity to be changed in size or scale and ability of computing process to be used or produced in a range of capabilities. Django support hundreds of millions of requests as we have seen in the cases of Instagram, Bitbucket and more.
- Modularity
- Using Python in this framework allows you to have benefits from all Python libraries and assures a very good readability.

React

React.js is a JavaScript library that was created by Facebook. It is often thought of as the “view” in a model-view-controller (MVC) user interface. This makes sense when you consider the fact that the only function that must be implemented in React is the “render” function. The render function provides the output that the user sees (the “view”).

React is a library for building composable user interfaces. It encourages the creation of reusable UI components, which present data that changes over time. React implements one-way reactive data flow, which reduces the boilerplate and is easier to reason about than traditional data binding. React "reacts" to state changes in your components quickly and automatically to re-render the components in the HTML DOM by utilizing the virtual DOM. The virtual DOM is an in-memory representation of an actual DOM. By doing most of the processing inside the virtual DOM rather than directly in the browser's DOM, React can act quickly and only add, update, and remove components which have changed since the last render cycle occurred.

Why React

React provides different features the major features include:

- **JSX:** JSX is JavaScript syntax extension. It isn't necessary to use JSX in React development, but it is recommended.
- **Virtual DOM:** This characteristic of react helps to speed up the app development process and offers flexibility.
- **Component Based Architecture:** React is all about components. You need to think of everything as a component. This will help you maintain the code when working on larger scale projects.
- **Unidirectional data flow and Flux:** React implements one-way data flow which makes it easy to reason about your app. Flux is a pattern that helps keeping your data unidirectional.

React is based on components and states. This is what makes react such a popular library. When you want to create an application, you usually break it into simpler parts. When programming with React, you will want to break your interface into its most basic parts, and those will be your React components.

REDUX

Redux is a predictable state container for JavaScript apps. As the application grows, it becomes difficult to keep it organized and maintain data flow. Redux solves this problem by managing application’s state with a single global object called Store. Redux fundamental principles help in maintaining consistency throughout your application, which makes debugging and testing easier.

Redux components are as follows:

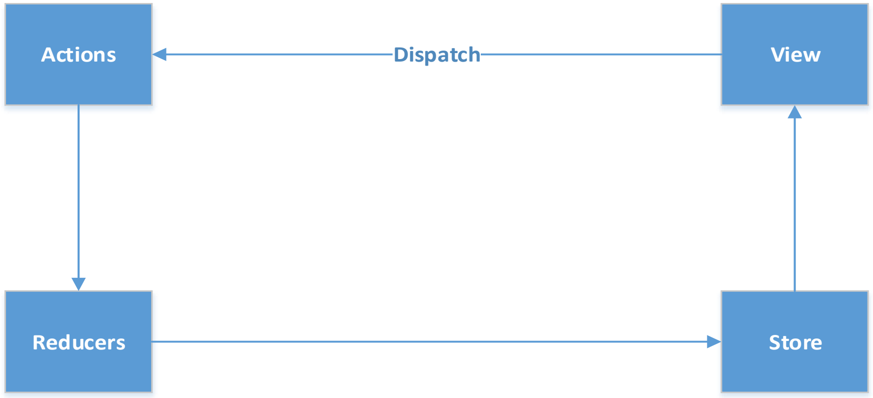


Figure 4. 3 Redux components

An action is a plain object that describes the intention to cause change with a type property. They are the only source of information for the store as per Redux official documentation. It carries a payload of information from your application to store.

They are the simple functions used to initiate an HTTP call using any HTTP platform, such as fetch or Axios, and result in a response from the server. Each action created in the application has an intention to fetch various information from the server using the API. Reducers are a pure function in Redux. Pure functions are predictable. Reducers are the only way to change states in Redux. It is the only place where you can write logic and calculations. Reducer function will accept the previous state of app and action being dispatched, calculate the next state and returns the new object.

Reducers are normal functions used to manage a whole application's state. They use the name of the action to be generated along with the action data. Based on the action data, the state object can be manipulated using different switch cases.

A store holds the whole state tree of your application. The only way to change the state inside it is to dispatch an action on it. A store is not a class. It's just an object with a few methods on it.

Redux follows the unidirectional data flow. It means that your application data will follow in oneway binding data flow. As the application grows & becomes complex, it is hard to reproduce issues and add new features if you have no control over the state of your application. Redux reduces the complexity of the code, by enforcing the restriction on how and when state update can happen.

4.5.1 DATA GATHERING

To successfully develop the desired web based online product distribution data was collected from the marketing team how they have been selling products and what type of method they were using to contact customers.

4.5.2 HARDWARE REQUIREMENT

The only hardware requirement to develop the system is personal computer which will enable us to host a local server in order to see our progress and test the frontend functionalities by using necessary web development software tools.

4.5.3 SOFTWARE REQUIREMENT

The software tools that are used to design the system are: Visual Studio Code, Postman, Node.js, and Python.

Visual Studio Code: Visual Studio Code is a code editor redefined and optimized for building and debugging modern web and cloud applications. It is compatible with all platforms and environment.

Postman: Postman is an API development tool which helps to build, test and modify APIs. It has the ability to make various types of HTTP requests like GET, POST, PUT, PATCH, saving environments for later use, converting the API to code for various languages like JavaScript, Python.

Node.js: Node.js is an open-source, cross-platform, back-end JavaScript run runtime environment that runs on the V8 engine and executes JavaScript code outside a web browser.

Python: Python is a programming language and windows does not come with a python interpreter installed. We need to explicitly install it and that installer should give us the option to append the proper paths to environmental variables and execute python files.

4.5.4 DESIGN

In this category detailed description of the system or model for product distribution system is discussed. It requires to pass via a series of steps starting from developing the desired web app to final output of the web, including testing online payment system. Descriptions about the system design and analysis are presented.

4.5.4.1 System Architecture

The system has three main components: frontend, backend and database. The frontend is implemented using react + redux for creating user view page and create our react app, use react bootstrap, implement react router and test our server using the local host. Node.js JavaScript is used to create the front end. In the backend python Django is implemented. I start by creating working environment, installing Django and Django rest frame work library for application program interface (API). After both the frontend and backend are created, Axios is used to fetch data from backend to the frontend.

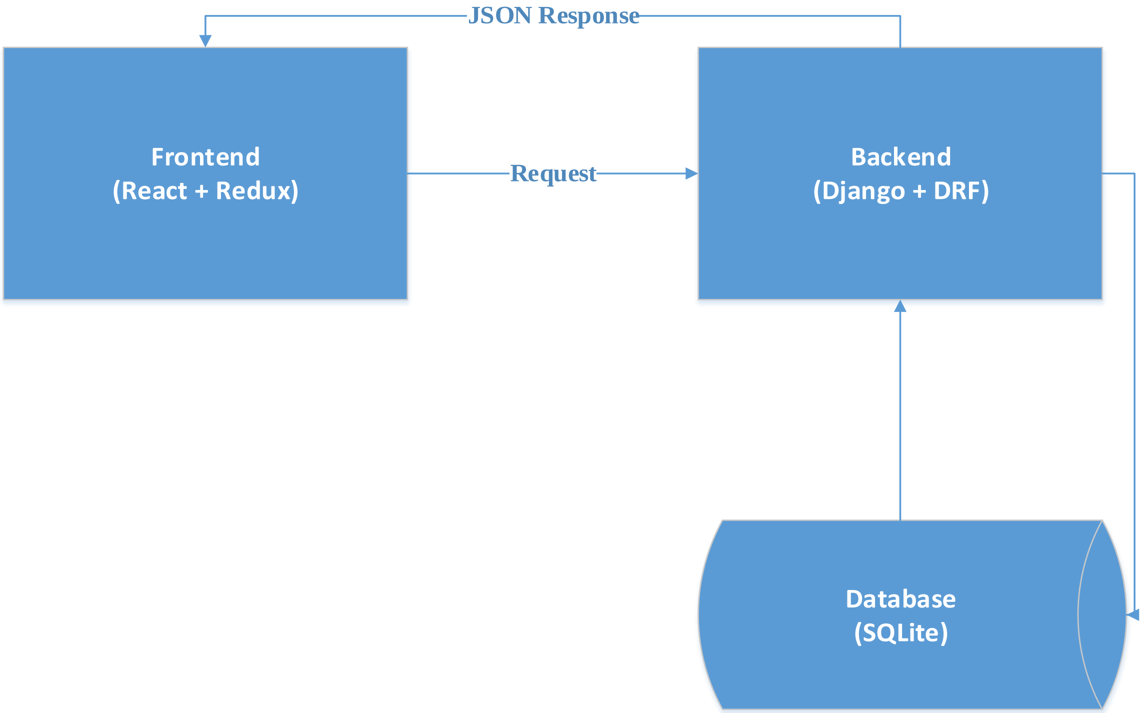


Figure 4. 4 Over all architecture of the system

The frontend runs on a local server host port: http://127.0.0.1.3000 , so fist I started the project by rendering static data from the react folder and once the backend was in developed was hosted on different port: http://127.0.0.1.8000. After having both servers and running them on a web browser and once we have that backend with API and database now we can use different method to make data request from the front end using Axios that will be sent to the backend. In our backend we are going to get that request inside of view. Now, in that view, we will send that to our database which is SQLite. The database will get that data and send it back into our view. And before we return that data to the frontend, we just need to serialize our data, because right now that is a python query set and we can only get JSON data in our frontend. So, we serialize that data and then return back to the frontend. These are request methods that are used in this project:

- **GET Method:** a request method which is used to retrieve intended data from the database. If we just want to get our products, we just send a get request GET /api/products/ to port 8000.
- **POST Method:** a request method that enable us to create a user or product in the database. If we want to create a product, we send a post request POST /api/product/create/ to the backend port
- **PUT Method:** a request method that allow us to update existing product or user. If we want to update a product, we send a put request PUT /api/product/update/id to the backend port.
- **DELETE Method:** a request method which is used to remove product or a user from the data base. If we want to delete a product, we send a delete request DELETE /api/product/create/ to the backend port.

4.5.5 System Analysis

This approach uses a unified modeling language (UML) in which it depends on the visual modeling of the system. UML is the industry-standard language for specifying, visualizing, constructing, and documenting the artifacts of software systems.

4.5.5 .1 Behavioral diagram

Behavioral diagram describe the functionality of the software system. It shows how the system interact with the user and it self. The behavioral diagrams used in this project are:

1. Use case diagram
2. Activity diagram

1. Use case diagram

Use case diagrams are graphical representation of overview of behavior of the system and shows the relationship between the actors and use cases in the system. The most important role of a use case diagram is to communicate the systems functionality and behavior to the customer or end user.

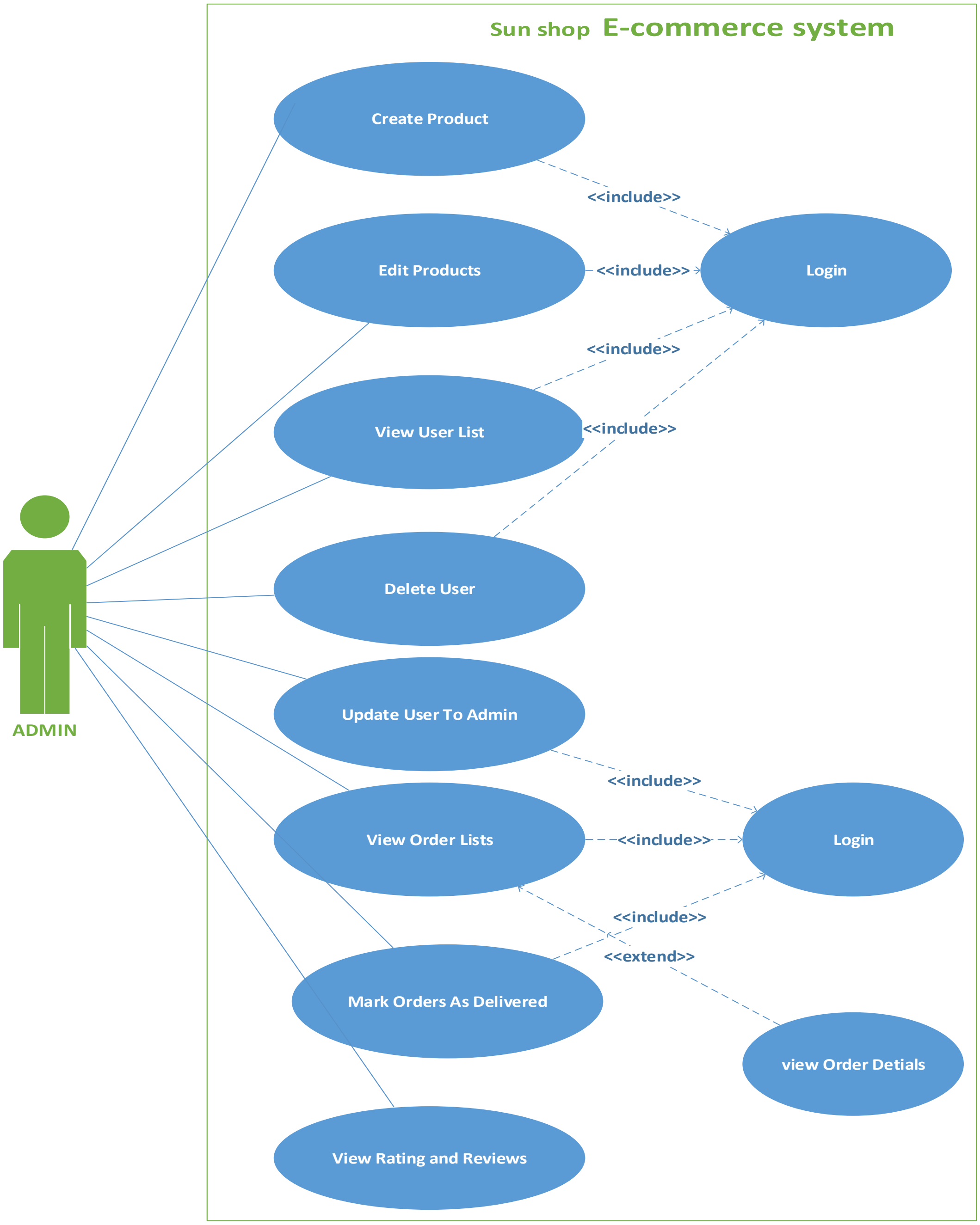


Figure 4. 5 admin use case diagram

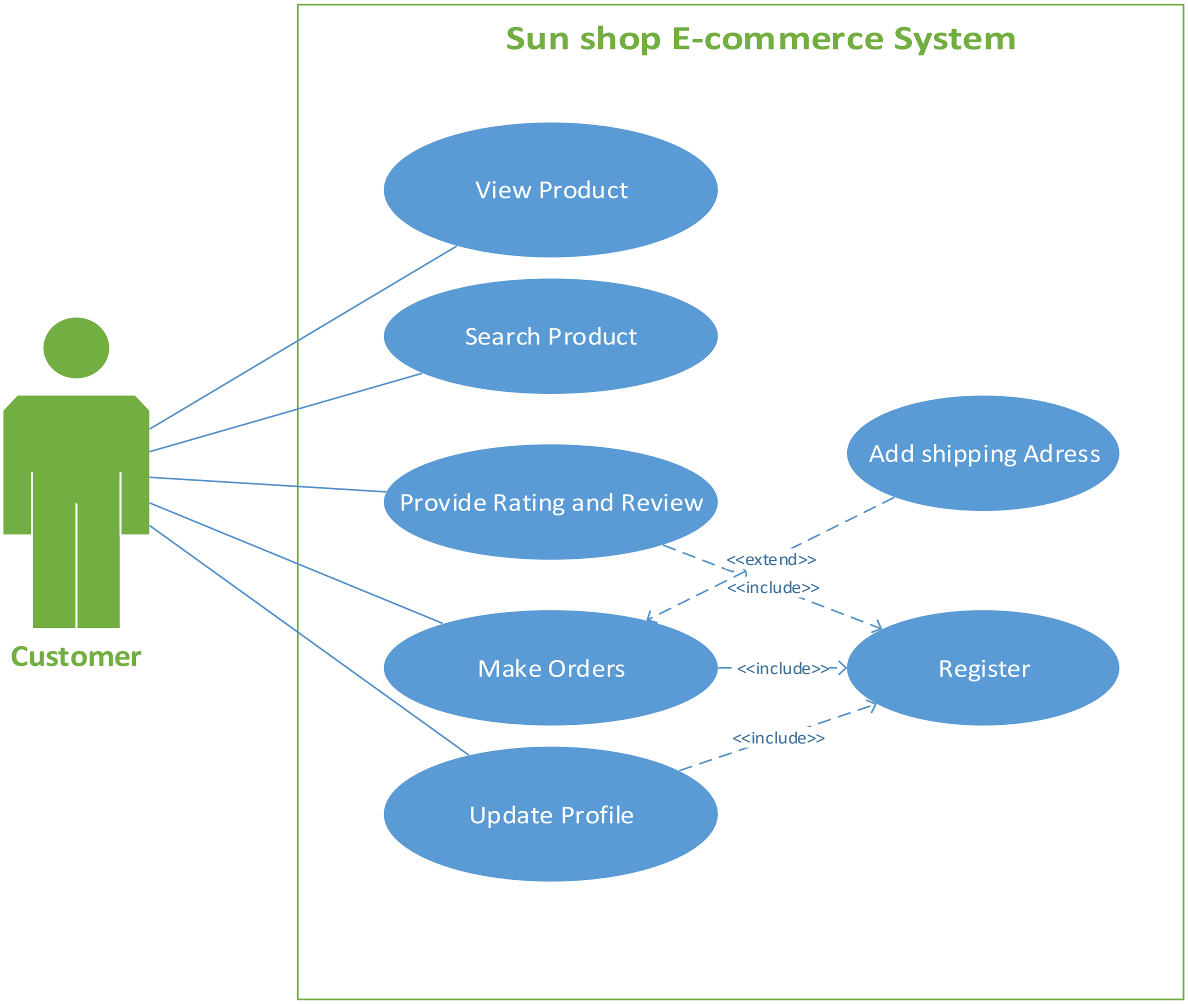


Figure 4. 6 customer use case diagram

2. Activity diagram

Activity diagrams are graphical representations of workflows of stepwise activities and actions with support for choice, interaction and concurrency. In the Unified Modeling Language, activity diagrams show the overall flow of control.

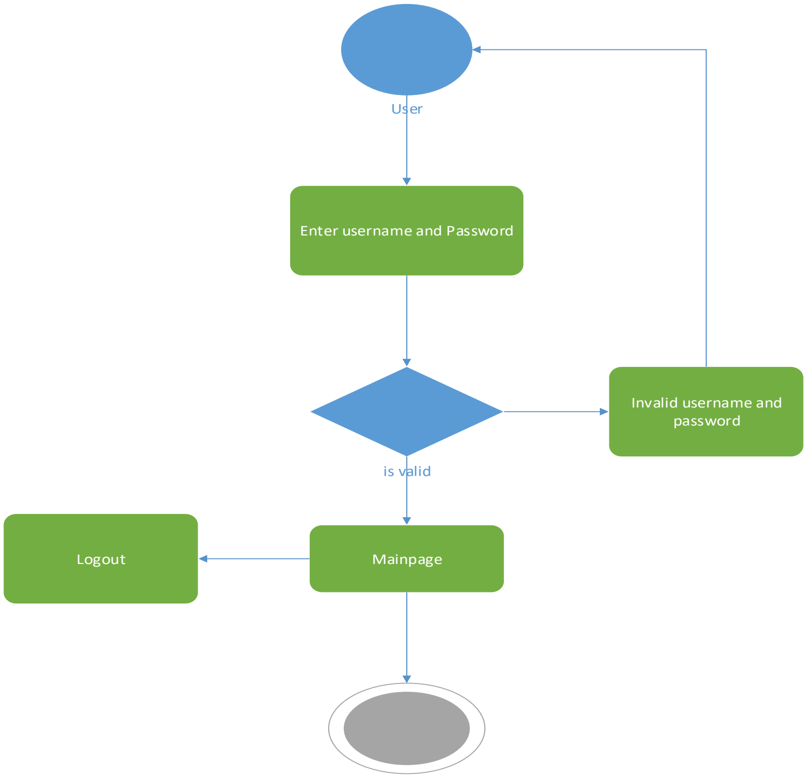


Figure 4. 7 Activity diagram for login

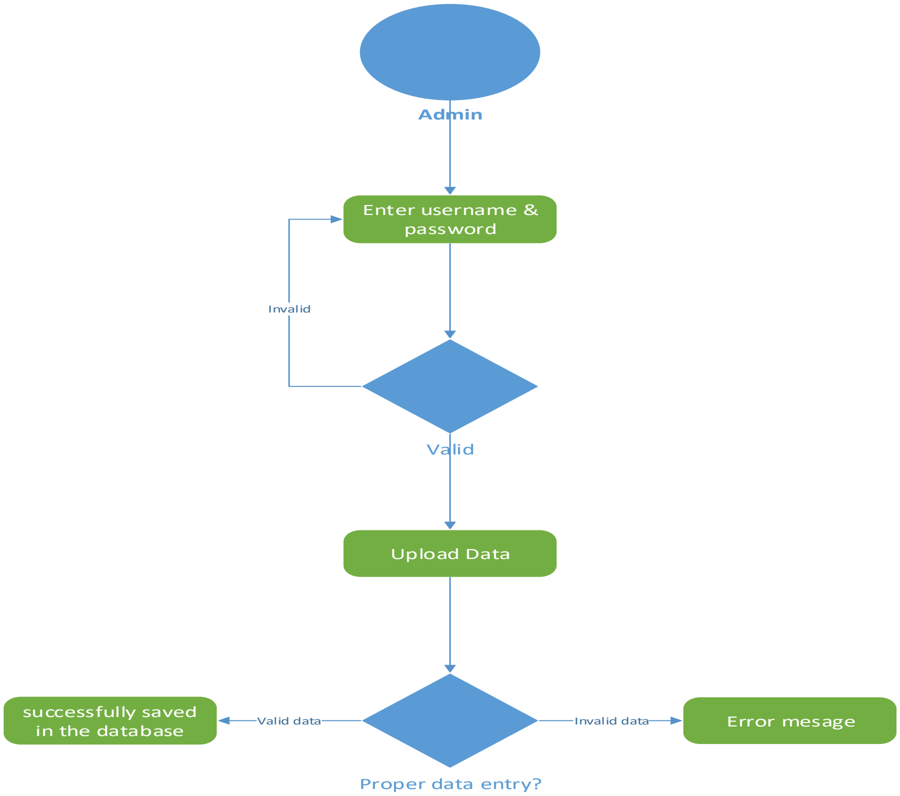


Figure 4. 8 Activity diagram for upload new item

4.5.5 .2 Structural diagrams

Structural diagram provides a logical overview of all or parts of a software system. It acts as a look inside a given structured classifier, defining, its configuration classes, interfaces, packages, and the relationships between them at a micro-level. The structural diagram used in this project is Class diagram.

- **Class diagram**

Class diagram is a structural diagram which is used for general conceptual modeling of the structure of the application, and for detailed modeling, translating the model into programming code. Class diagrams can also be used for data modeling.

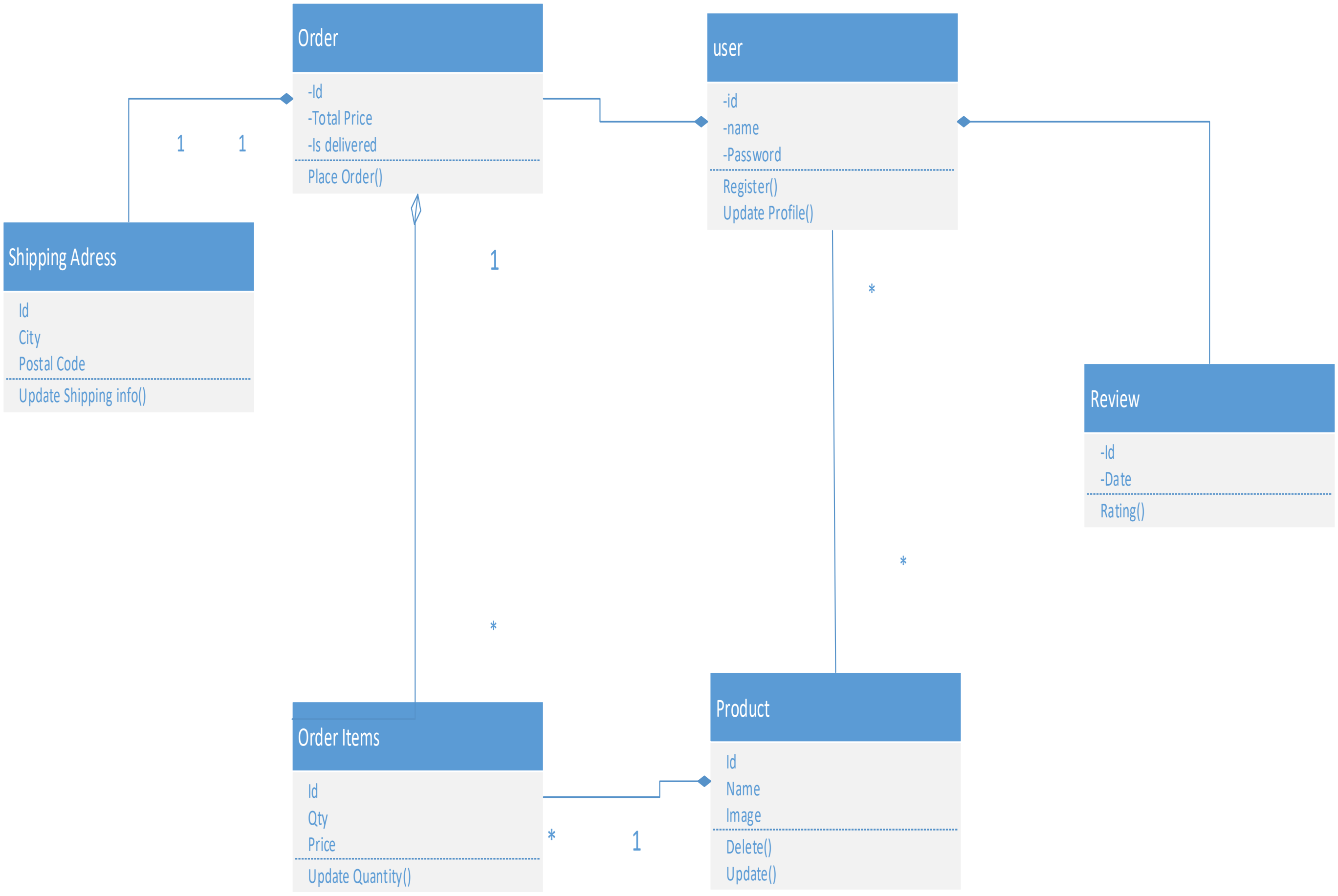


Figure 4.9 Class diagram

4.5.5 .3 Entity-Relationship Diagram

An Entity-Relation Diagram shows the relationship among entity sets in terms of database management system, an entity is a table or attribute of a table in a database, so by showing relationship among tables and their attributes, ER diagram shows the complete logical structure of a database.

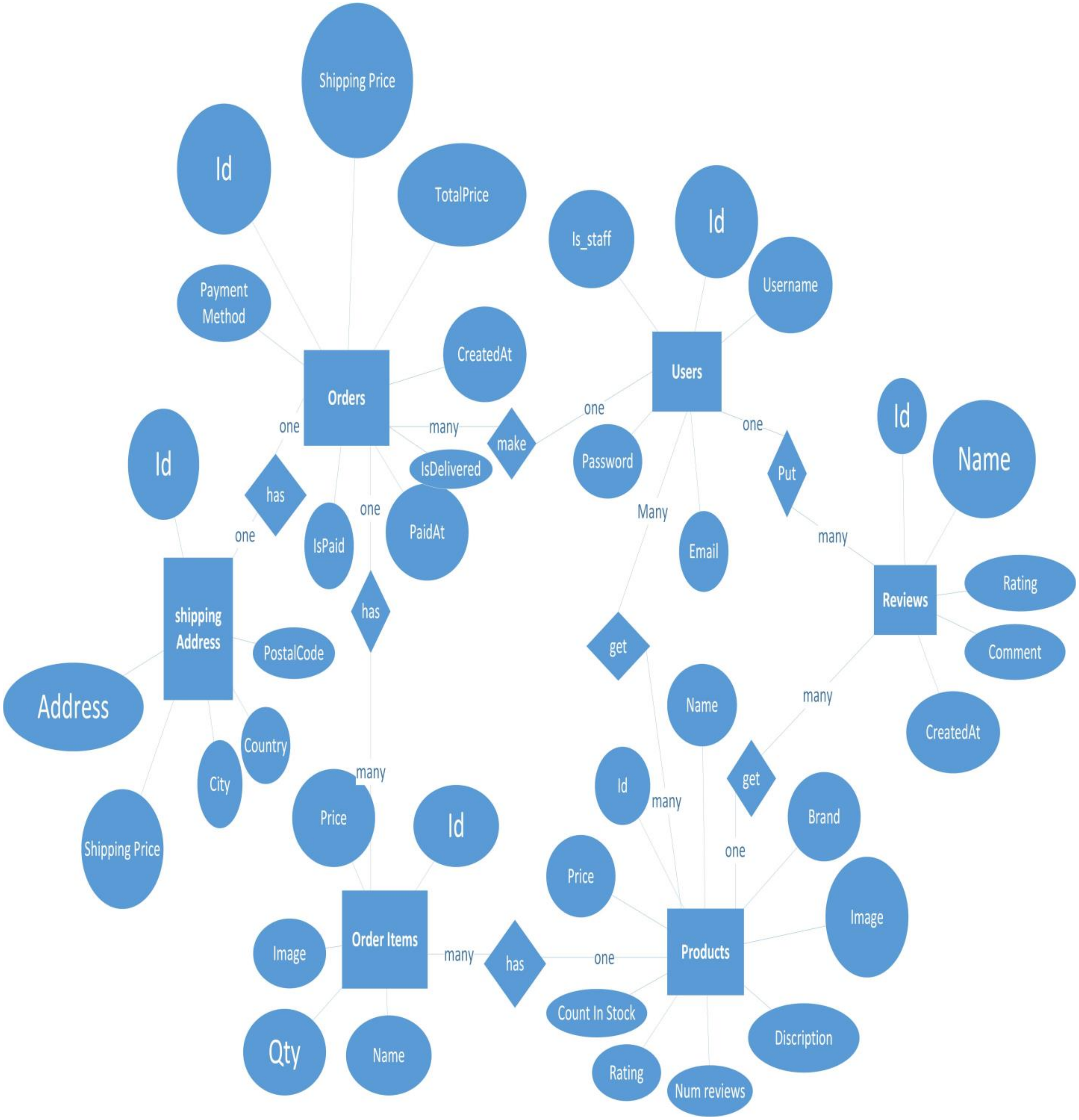


Figure 4.10 E-R Diagram

4.6 RESULT AND DISCUSSION

In this section, the results of the developed web system have been explained and presented. The user interface was implemented using react with a little bit of CSS and HTML. All the stated objectives was implemented fully functional. Below listed screenshots are the user interfaces with the functionalities.

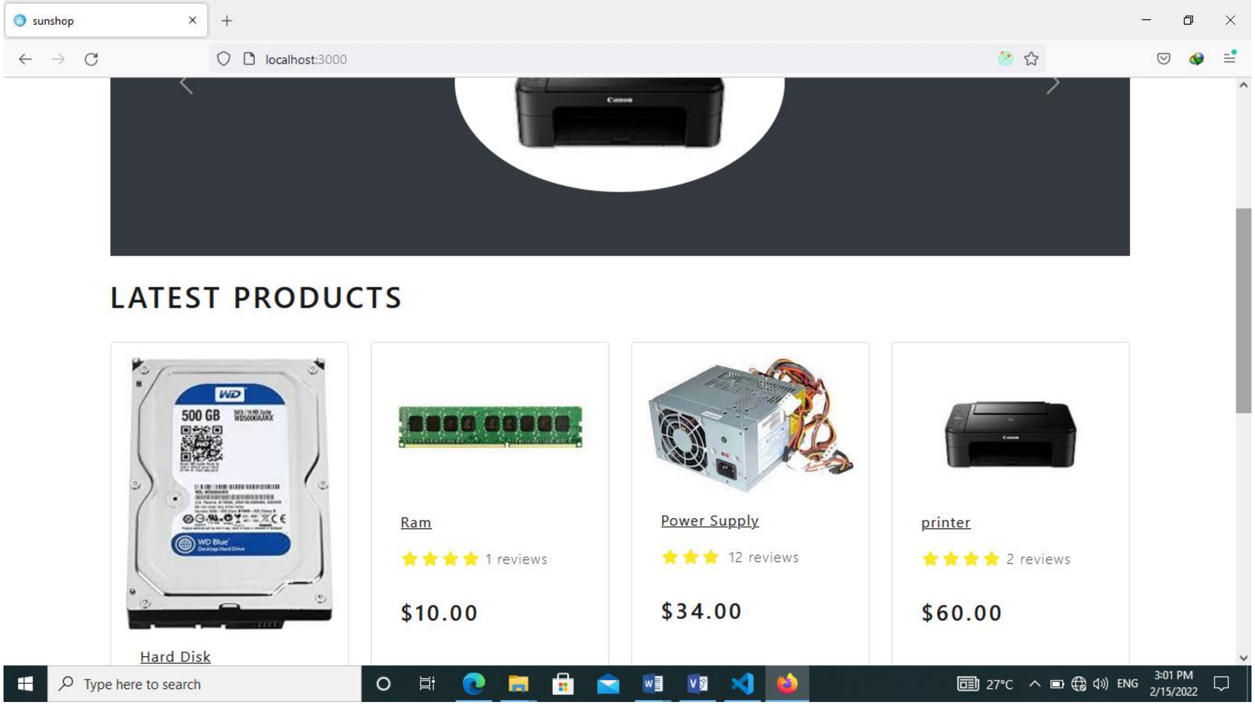
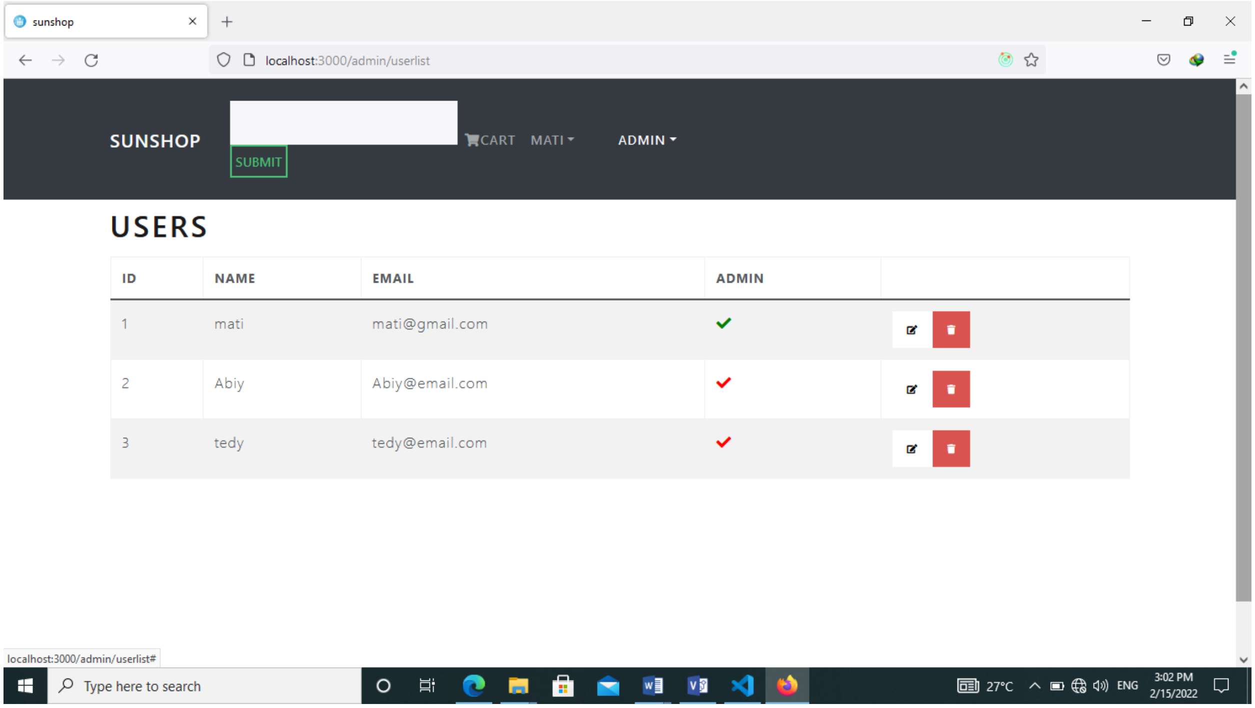


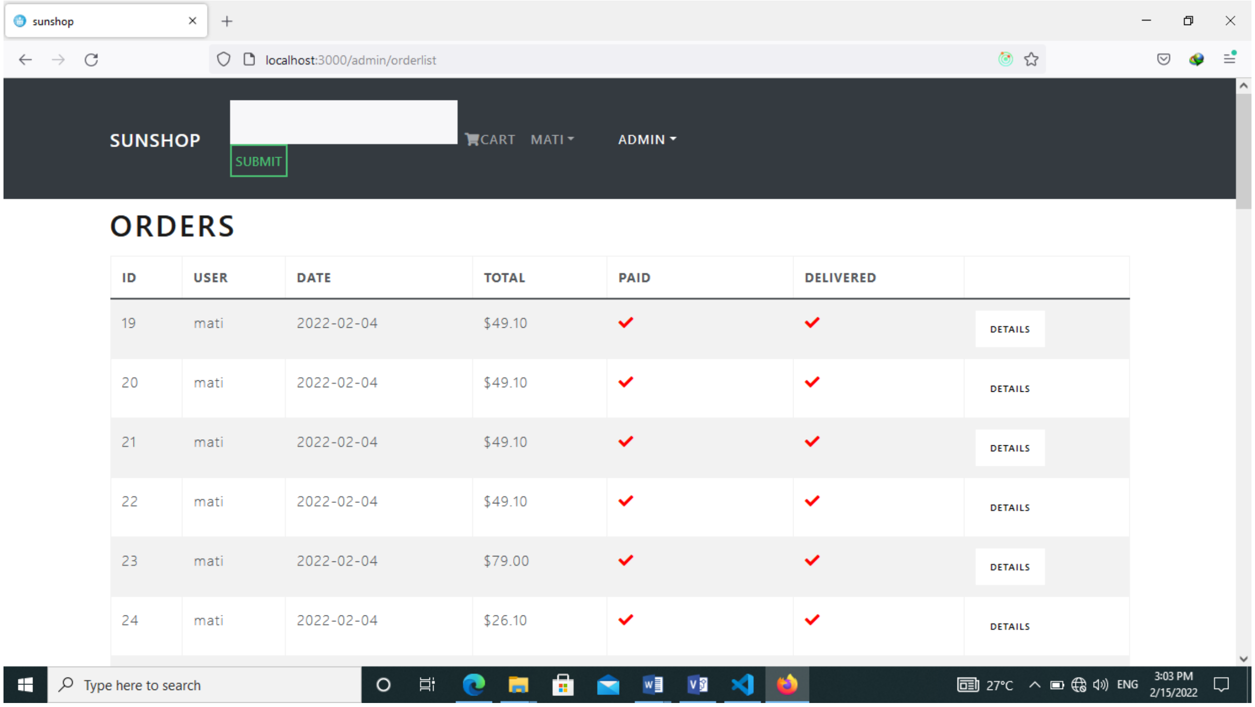
Figure 4.11 Home page



USERS

ID	NAME	EMAIL	ADMIN	
1	mati	mati@gmail.com	✓	
2	Abiy	Abiy@email.com	✓	
3	tedy	tedy@email.com	✓	

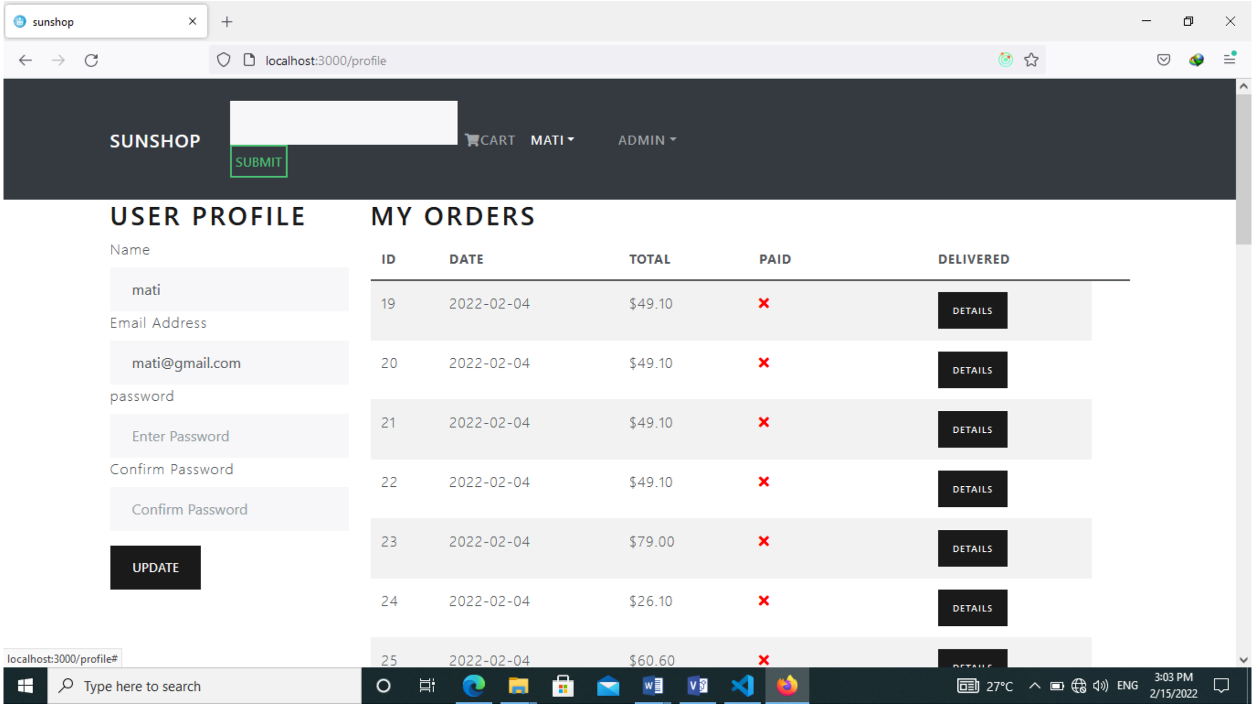
Figure 4.12 User list page



ORDERS

ID	USER	DATE	TOTAL	PAID	DELIVERED	
19	mati	2022-02-04	$49.10	✓	✓	DETAILS
20	mati	2022-02-04	$49.10	✓	✓	DETAILS
21	mati	2022-02-04	$49.10	✓	✓	DETAILS
22	mati	2022-02-04	$49.10	✓	✓	DETAILS
23	mati	2022-02-04	$79.00	✓	✓	DETAILS
24	mati	2022-02-04	$26.10	✓	✓	DETAILS

Figure 4.13 Order List page



USER PROFILE

Name

mati

Email Address

mati@gmail.com

password

Enter Password

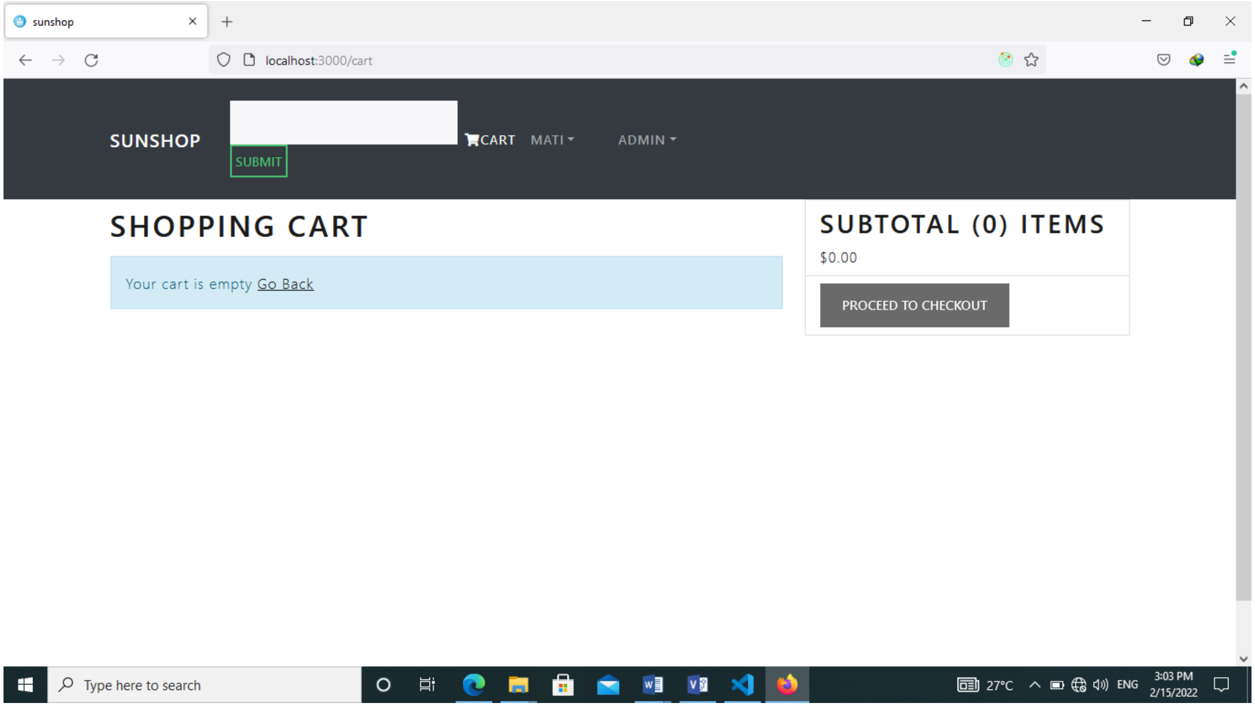
Confirm Password

Confirm Password

MY ORDERS

ID	DATE	TOTAL	PAID	DELIVERED
19	2022-02-04	$49.10	×	DETAILS
20	2022-02-04	$49.10	×	DETAILS
21	2022-02-04	$49.10	×	DETAILS
22	2022-02-04	$49.10	×	DETAILS
23	2022-02-04	$79.00	×	DETAILS
24	2022-02-04	$26.10	×	DETAILS
25	2022-02-04	$60.60	×	DETAILS

Figure 4.14 Profile page



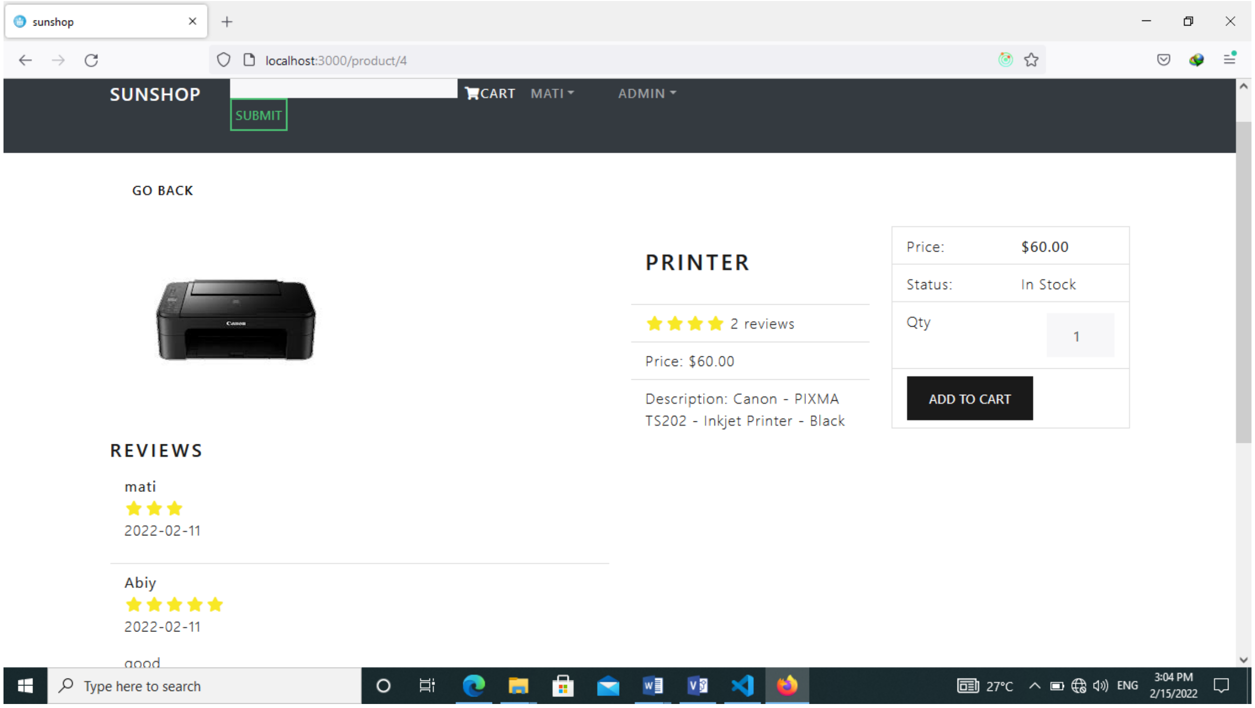
SHOPPING CART

Your cart is empty Go Back

SUBTOTAL (0) ITEMS

$0.00

Figure 4.15 Cart page



PRINTER

2 reviews

Price: $60.00

Description: Canon - PIXMA TS202 - Inkjet Printer - Black

Price:	$60.00
Status:	In Stock
Qty	1

REVIEWS

mati
2022-02-11

Abiy
2022-02-11
good

Figure 4.16 Product page

4.7 CONCLUSION AND RECOMMENDATION

4.7.1 Conclusion

Technology has made significant progress over the years to provide consumers a better online shopping experience and will continue to do so for years to come. This project summarizes a research project in which an online product distribution service was proposed, designed, developed, demonstrated and implemented. The main barriers for online product distribution system adoption were the barriers between distributers, information systems usage skills, delivery etc.

4.7.2 Recommendation

This project has provided basic and required functionalities in order to make the system better the following tasks are recommended:

- Integrating with a real payment system
- Making the user interface more attractive

CHAPTER FIVE

CONCLUSION AND RECOMMENDATION

5.1 CONCLUSION

This internship program provided us to introduce and give a knowhow on how the professional life looks like, and to make a bridge that links the student life with the professional life, in addition to observe and fill the gap between the theoretical and practical knowledge. It definitely will be sensible to scale practice up and to replicate it in other disciplines as well.

5.2 RECOMMENDATION TO THE COMPANY

By studying the company organization structure and work flow I would like to recommend the company the following terms:

- Hire highly professional skilled man powers, this enables the firm to attend projects with a lot of speed and accuracy.
- Use modern desktops and equipment, this will enable the projects to be finished with a lot of speed and will be captivating students to learn on those materials.
- Arrange the necessary materials and office for interns, this will help the intern to incited and work with responsibility.

5.3 RECOMMENDATION TO HAWASSA UNIVERSITY

It is well known that without skilled man power the development of a country and company is unthinkable. Therefore, the university have to keep paying a great attention for the internship program and to give enough support for the intern to complete its training successfully.

References

1. Samuel Dauzon, Aidas Bendoraitis, Arun Ravindran Packt Publishing Ltd, 2016. “Django: Web Development with Python”.

2. Suvash Sedhain, Published at2006. “Web framework for Python Django Book”,

3. Django Documentation Link ‘docs.djangoproject.com’

4. React documentation Link ‘reactjs.org’

5. Learning websites like https://stackoverflow.com, https://pluralsight.com

Appendices

The entity tables from the E-R Diagram

No	Field name	Data type	constraints
1	_id	Int	Primary key
2	username	Char	none
3	first_name	Char	none
4	last_name	Char	none
5	email	Char	none
6	password	Char	none
7	Is_staff	boolean	none
8	Is_activeuser	boolean	none

Table 4. 1 user entity table

No	Field name	Data type	constraints
1	_id	Int	Primary key
2	order	Int	none
3	address	char	none
4	city	Char	none
5	postalCode	Char	none
6	country	Char	none
7	shippingPrice	Decimal	none

Table 4. 2 shipping address entity table

No	Field name	Data type	Constraints
1	_id	Int	Primary key
2	product	Int	none
3	name	Int	none
4	qty	Char	none
5	price	Int	none
6	Image	Decimal	none
7	order	char	none

Table 4. 3 order item entity table

No	Field name	Data type	constraints
1	_id	Int	Primary key
2	user	Int	none
3	paymentMethod	Char	none
4	taxPrice	Decimal	none
5	shippingPrice	Decimal	none
6	Totalprice	Decimal	none
7	iaPaid	Boolean	none
8	paidAt	Date	none
9	IsDelivered	Boolean	none
10	deliveredAt	Date	none
11	createdAt	Date	none

Table 4. 4 order entity table

No	Field name	Data type	constraints
1	_id	Int	Primary key
2	user	Int	none

3	name	Char	none
4	image	Char	none
5	brand	Char	none
6	category	Char	none
7	description	Char	none
8	rating	Int	none
9	numReviews	Int	none
10	price	Decimal	none
11	countinStock	Int	none
12	createdAt	Date	none

Table 4. 5 Product entity table

No	Field name	Data type	Constraints
1	_id	Int	Primary key
2	user	Int	none
3	product	Int	none
4	name	Char	none
5	rating	Int	none
6	comment	Text	none

7	createdAt	data	none

Table 4. 6 review entity table

Appendices 2

Sample Code Used for Cart

```
import React, { useEffect } from 'react'

import { Link } from 'react-router-dom'

import { useDispatch, useSelector } from 'react-redux'

import { Row, Col, ListGroup, Image, Form, Button, Card } from 'react-bootstrap'

import Message from '../components/Message'

import { addToCart, removeFromCart } from '../actions/cartActions'

function CartScreen({ match, location, history }) {

    const productId = match.params.id

    const qty = location.search ? Number(location.search.split('=')[1]) : 1

    const dispatch = useDispatch()

    const cart = useSelector(state => state.cart)

    const{ cartItems } = cart

    useEffect(() => {

        if (productId) {

            dispatch(addToCart(productId, qty))

        }

    }, [dispatch, productId, qty])

    const removeFromCartHandler = (id) => {
```

```
    dispatch(removeFromCart(id))
}
const checkoutHandler = () =>{
    history.push('/login?redirect=shipping')
}
return (
    <Row>
        <Col md={8}>
            <h1>Shopping Cart</h1>
            {cartItems.length === 0 ?(
                <Message variant='info'>
                    Your cart is empty <Link to ='/'>Go Back</Link>
                </Message>
            ) : (
                <ListGroup variant='flush'>
                    {cartItems.map(item => (
                        <ListGroup.Item key={item.product}>
                            <Row>
                                <Col md={2}>
```

```
  <Image src={item.image} alt={item.name} fluid rounded />
</Col>
<Col md={3}>
  <Link to={`/product/${item.product}`}>{item.name}</Link>
</Col>
<Col md={2}>
  ${item.price}
</Col>
<Col md={3}>
  <Form.Control
    as="select"
    value={item.qty}
    onChange={(e) => dispatch(addToCart(item.product, Number(e.target.value)))}
  >
    {
      [...Array(item.countInStock).keys()].map((x) => (
        <option key={x + 1} value={x + 1}>
          {x + 1}
        </option>
```

```
                  ))
                }
              </Form.Control>
            </Col>
            <Col md={1}>
              <Button
              type='button'
              variant='light'
              onClick={() => removeFromCartHandler(item.product)}
              >
                <i className='fas fa-trash'></i>
              </Button>
            </Col>
          </Row>
        </ListGroup.Item>
      ))}
    </ListGroup>
  )}
</Col>
```

```
      <Col md={4}>
        <Card>
          <ListGroup variant='flush'>
            <ListGroup.Item>
              <h2>Subtotal ({cartItems.reduce((acc, item) => acc + item.qty, 0)}) items</h2>
              ${cartItems.reduce((acc, item) => acc + item.qty * item.price, 0).toFixed(2)}
            </ListGroup.Item>
            <ListGroup.Item>
              <Button
                type='button'
                className='btn-block'
                disabled={cartItems.length === 0}
                onClick={checkoutHandler}
              >
                Proceed To checkout
              </Button>
            </ListGroup.Item>
          </ListGroup>
```

```
                </Card>

              </Col>

          </Row>

    )

}

export default CartScreen
```