DigitalGate

Author:

Ayman Hunjol

This research provided a continuation of the requirements for the graduation project in the Faculty of Information Technology.

2020/2021

Table of Contents

CHAPTER ONE 4

Introduction 4

1.1 Problem Statement 4

1.2 Background 4

1.3 Scope 5

1.4 Project Objective 5

1.5 Definition and Abbreviations: 6

1.6 Development and Design 6

1.7 Project outline 6

CHAPTER TWO 7

LITERATURE AND METHODOLOGY 7

2.1 Current System 7

2.2 Proposed System 7

2.3 Project Methodology 10

CHAPTER THREE **10**

SYSTEM REQUIREMENTS **10**

3.1 Requirements Collection 10

3.2 Functional Requirements 10

3.3 Non-Functional Requirements 11

CHAPTER FOUR **12**

DESCRIBING AND ANALYZING SYSTEM REQUIREMENTS **12**

4.1 Introduction 12

4.2 ER Diagram 12

4.3 System Database Schema 12

4.4 UML 13

4.4.1 Actors 13

4.4.2 Static context diagram 14

4.4.3 Packages 14

4.4.4 Full Use Case 15

4.4.4.1 Admin use case 16

4.4.4.2 Customer use case 17

4.4.5 Textual description 17

4.4.6 Sequence Diagrams 27

4.4.7 Activity Diagrams 34

CHAPTER FIVE 36

DESIGN AND SYSTEM IMPLEMENTATION 36

5.1 Introduction 36
5.2 Programming Languages 36
5.3 Why Using These Programming Languages 36
5.4 Why Using MYSQL server 37
5.5 System interfaces 37

CHAPTER SIX 46

System Testing 46

6.1 Testing Specification. 46
6.1.1 Pretend a customer 46
6.1.2 Research Existing Standards and Guidelines 46
6.1.3 Review and test similar programs 46
6.2 Low-Level specification 47
6.3 Test scenario 49

CHAPTER SEVEN 50

SYSTEM EVOLUTION 50

7.1 Conclusion 50
7.2 Future work 50

CHAPTER ONE

Introduction

1.1 *Problem Statement*

E-commerce provides an easy way to sell products to a large customer base. However, there is a lot of competition among multiple e-commerce sites. When users land on an e-commerce site, they expect to find what they are looking for quickly and easily. Also, users are not sure about the brands or the actual products they want to purchase.

Many customers nowadays search for their products on Google rather than visiting specific e-commerce sites. The purpose of any e-commerce website is to help customers narrow down their broad ideas and enable them to finalize the products they want to purchase. The customers have a very broad idea about what they want to buy, so we should help them narrow these broad ideas. For example, suppose a customer is interested in purchasing a mobile. His or her search for a mobile should list mobile brands, operating systems on mobiles, screen size of mobiles, and all other features as facets.

Another way to help them is by showing them some Products that have some similar properties for what they visited on the website which is called 'Recommendation System'. so the user can have a big idea about Products that are available in store or even buy from the Products that are listed by the Recommendation System.

1.2 *Background*

These days, E-Commerce has become something Important in our Life and we use it daily and everywhere. As a result we can find millions of online Stores Worldwide.

Online Marketing reaches each person these days in many ways. Also, it opens opportunities for People to Make Money and hire other People. E-Commerce has many types, for Example:

1. you can sell Physical goods to the People and ship it Worldwide.
2. You can sell digital products to the People Worldwide through an online store and from this point we have generated our Project idea.

Our Project idea Mainly aims to provide the customer with his needs of Digital Products Like: Software Licenses, Subscriptions, Gift Cards.

1.3 *Scope*

First of All, our system aims to provide the customer with his needs from Digital Products on time without any delay so that the customer can be satisfied and buy again more and more.

Also, our store targets individual Customers ONLY. not Companies. To achieve this goal, we have to add some stock to our store then Suggest some Products to the Customer in many ways. For Example:

1. List Some of the Top Seller Products on our Store.
2. When he visits Some Product Page we Suggest him some Similar Products through our Recommendation System.

so we can know which products he likes and let us be able to know which products to suggest to him using our Recommendation System.

1.4 Project Objective

Our Project mainly aims to provide the customer with his needs of Digital Goods even if it was a Software Licenses or Some Service Subscription or even a Gift Cards. Also, we aim to make it easier on the visitor from simple things to the hardest things. For Example:

We aim to let the search Methodology and finding Products More easily for the Visitor So the Visitor saves his time and effort. And that can be done using our Recommendation System. Another Thing we want to achieve is that we want to get a customer Trust which may be the hardest point for any store to get as many customers as possible. To achieve this point, we provided the customer with an Email System so he can contact us at any time he wants in case he wants to ask about the redemption process of some product or if have some problems or any other questions.

Also, we aim to provide the customer with Safe Payment Methods Like: Visa and PayPal. We do this for the reason in the Example as the following: if some customer

does not feel comfortable to Pay Using his Card for any reason, he may wish if he have PayPal option to Pay because it is a safe Way to Pay plus it offers a refund in case there was some problems with the seller even if it was a normal Person or an online Store Like in our Case.

So that the customer can feel comfortable that he can get his money back at any time he feels that something wrong is happening with his purchased Product.

1.5 Definition and Abbreviations:

Admin: The Person who controls the System, list products, change prices, fill the stock, Answer Tickets.

Customer: The Person who comes to buy Products, does searches on products he likes, pays the price of Product.

The System: a group of parts or subsystems whose relations overlap between each other and the system that includes them, and each part of which depends on the other to achieve the goals pursued by this total system.

CRUD: Create, Read, Update, Delete

1.6 Development and Design

➢MYSQL server

➢PHP Laravel framework

➢Bootstrap 4

➢JQuery and Ajax

➢HTML5

1.7 Project outline

The project consists of 8 chapters:

- Chapter one is an introduction that talks about project scope, problem statement, and background and project objective, definition and abbreviations and deployment and design

- Chapter two discusses literature and methodology, explains a difference between current systems and proposed systems,
- Chapter three is a stage that discusses system requirements
- Chapter four is a stage that focuses on describing and system UML designs.
- Chapter five is a stage that focuses on system design and website screen.
- Chapter six is a stage that focuses on system implementation.
- Chapter seven is a stage that focuses system testing
- Chapter eight present the conclusion and future work of the system

CHAPTER TWO

LITERATURE AND METHODOLOGY

2.1 Current System

After studying our System, we found that it's very helpful and used in most companies which have their own websites on the Internet. Each Company wants to get Customers to achieve the Main Goal which is Make Money. BUT the Company has some Problems to let the customer come to their side and keep it as one of its Future Customers. And They use a Non Personalized System which means that every user or visitor will see the same products as the other users. For Example: All users will see the Last Purchased Products or The Last Added Products on the Store. So, to make it more easy and efficient to the user we updated it to be a Personalized System So that each customer will see a different product based on his rating for some products on the website. In Non Personalized System, The System will store data like: The Products Names that were added recently, Their prices, Products sold recently, etc..

2.2 Proposed System

Recommendation System Methodology is represented by offering the Customer Some offers that he may Like or Even Buy it! But it's based on a technique. In the proposed system we will use collaborative filtering algorithms by searching a large group of users or items and finding a smaller list from it with tastes similar to yours. The proposed system implements both: Personalized recommendations and Non-Personalized recommendations.

In general, there are two common scenarios that will trigger our recommendation system within our system:

1. For guests and users who don't interact a lot with our products, the system automatically selects the Non-Personalized recommendation and displays some products to them
2. For the users who interact with products and the system a lot, the system selects personalized recommendations and starts to generate lists for them. Each user may get different products. Of course, the algorithm is based on the history of rating interactions for each user.

An example on how the personalized recommendation system work is as the following: Suppose we have the following array which generated by our system:

```
$products =  array(
   "User1" => array("Product 1" => 2, "Product 2" => 3,
                    "Product 3" => 3, "Product 4" =>; 4,
                    "Product 5" => 2,
                    "Product 7" => 3),
   "User2" => array("Product 13" => 2,
                      "Product 2" => 3,
                      "Product 6" => 3, "Product 4" => 3,
                      "Product 5" => 2, "Product 7" => 1),
   "User3" => array("Product 8" => 5, "Product 9" => 3,
                    "Product 3" => 1),
   "User4" => array("Product 17" => 5, "Product 10" => 3),
   "User5" => array("Product 13" => 1, "Product 9" => 2,
                    "Product 6" => 4, "Product 15" => 3,
                    "Product 11" => 2, "Product 7" => 3),
   "User6" => array("Product 13" => 3, "Product 12" => 1,
                     "Product 6" => 3, "Product 4" => 3,
                     "Product 14" => 2, "Product 7" => 3),
   "User7" => array("Product 16" => 2)
)
```

Let's say you are User3 and the products you have purchased till now are ‘product 8’, ‘product 9’ and product 3 with appropriate rating given. If you come to the website again, system should display the following products for user3:

```
Array(
    [Product 6] =>; 4
    [Product 4] => 4
    [Product 15] => 3
    [Product 7] => 3
    .
    .
    .
```

How did we get these products? Well, it’s all done by calculating similarity between users, if we notice that User3 has 3 products, and product 9 is purchased by User5, so we presume that User3 and User5 has similarity in products taste, and so, we pick the top rated products by User5 and display them for User3. We mainly focus on calculating the similarly distance which is defined as following:

“The state or fact of being similar or Similarity measures how much two objects are alike.”

Similarity distance can be calculated with multiple measures and in our system we focused and utilized “Euclidean distance” in order to calculate the similarity between users and generate recommended products for our system users. We mainly used the following formula in the calculation:

$$\sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}.$$

We build a full class with different functions, we first call getRecommendation function which accepts array of users along with products rated by each one which called preferences and second parameter which is the user who we want to display recommended products for him, basically this function will utilize another function called getSimilairtyDistance which accepts the preferences array, and user1 and user2 then it will calculate the similarly between the given users using euclidean distance and return value we do this for the user who wants to view his recommendations and the users in our database.

2.3 Project Methodology

The project team will use one of the methods used in software engineering (Systems Development Life Cycle) to achieve the system's expected outputs, as the team uses in electronically analyzing and building the system one of the methods used in software engineering (Systems Development Life Cycle), which starts with the system's planning stage, then analyzes the requirements, then designs the system, then develops and operates the system.

The beginning is on collecting the largest quantity of information about similar systems, projects and existing problems and see what problems do they lack and face within their systems and prepare our system to solve these problems, such as how to boost website visitor base, revenues, and profit, and identifying the troubles of the traditional technique in building e-commerce systems.

CHAPTER THREE

SYSTEM REQUIREMENTS

In this chapter, we will mention the requirements which are divided into functional and non-functional.

3.1 Requirements Collection

The requirements were collected by deep observations from similar up running ideas that specialize in Ecommerce. Also, the laws and regulations pertaining to this area were reviewed. Review collecting from within similar websites in order to gather the opinions of people about the system overall to solve problems and introduce new features.

3.2 Functional Requirements

- Admin:

1) View/Add/Update/Delete/Restore products
2) View/Add/Update/Delete/Restore categories
3) View/Add/Delete/Restore serial keys
4) View all orders
5) View/Add/Remove sale on category

6) View all users
7) View his profile
8) Update his account
9) Search for products

- Customer:

1) Update his account
2) View his orders
3) Search for products
4) View products

3.3 Non-Functional Requirements

1) **Speed:** The transactions and querying between Server and Database is very fast. This is performed by designing relational databases, and using schema objects to reduce the time that server needs to perform the process. Moreover, utilizing cache in order to minimize the access to database and increase performance of website as well as page loading.
2) **Security:** The system is well secured, the system is compatible with cloud-flare protection, all the requests are verified and filtered before applying them in the system, the sessions are stored in server side which makes it impossible for the user to interact or edit his sessions, it can be accessed from server-side only.
3) **Efficiency:** The system is able to take in all inputs to give results quickly and easily.
4) **Portability:** The website can be accessed by any device and any operating system. This is achieved by using responsive design that it's compatible with any device.

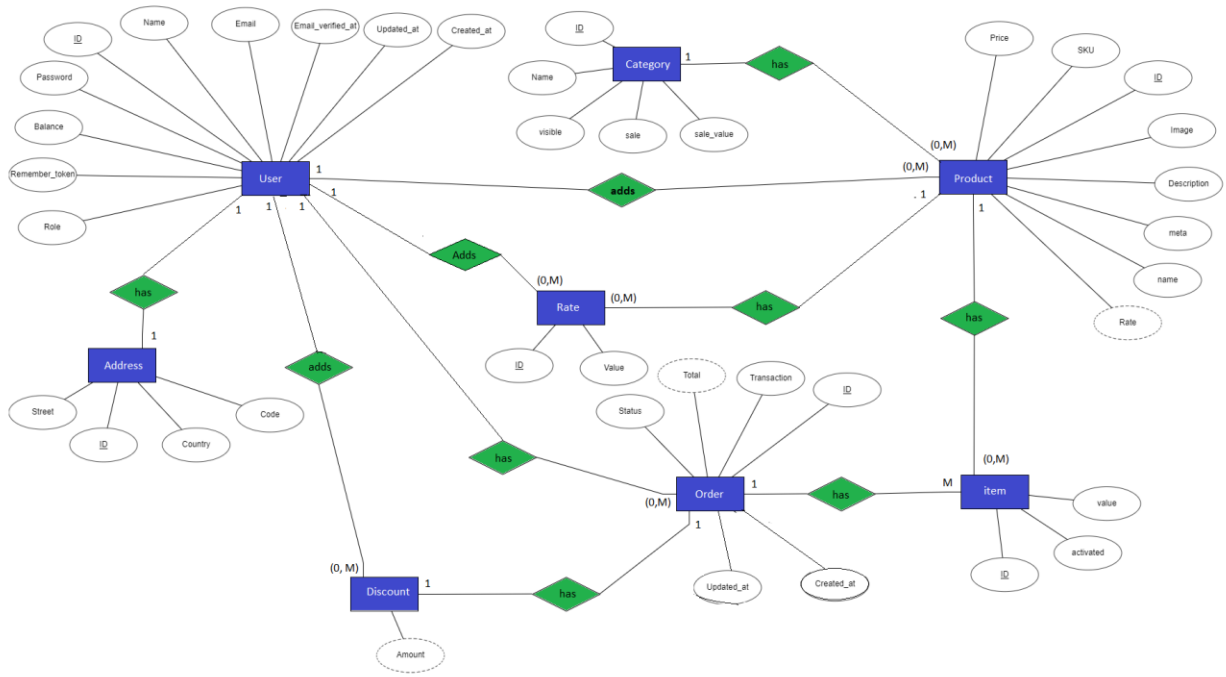
CHAPTER FOUR

DESCRIBING AND ANALYZING SYSTEM REQUIREMENTS

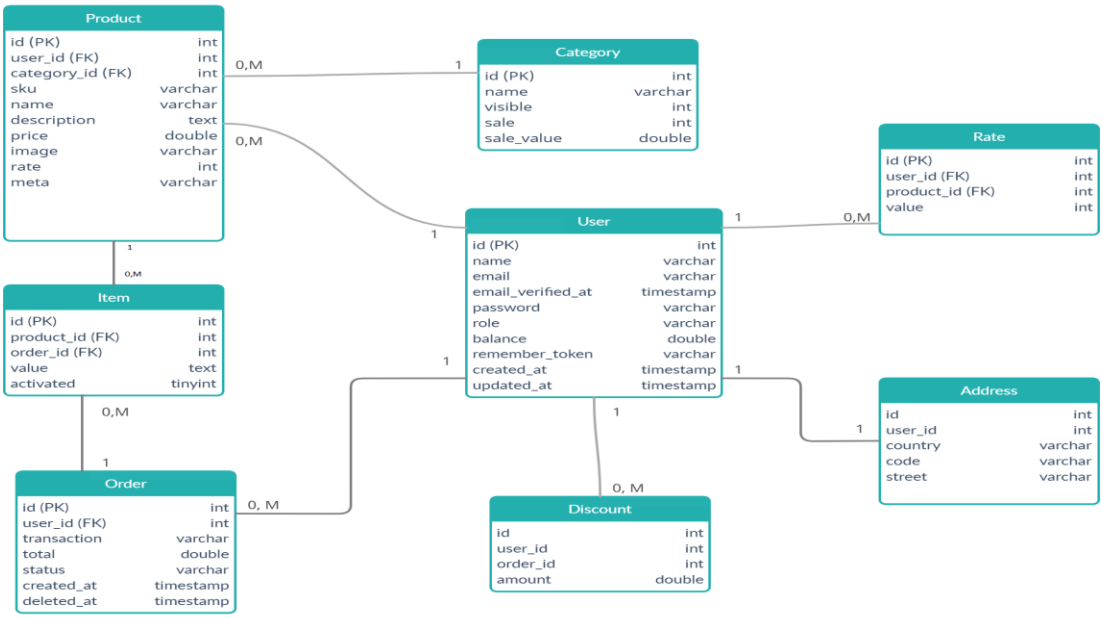
4.1 Introduction

This chapter describes the system generally and its mechanism of operation. It presents associate analysis of the system necessities that embrace a collection of functions that the system performs and the way to act with them. The system relationships are going to be drawn through drawings that illustrate the system's work, that facilitate the method of clearly understanding the system.

4.2 ER Diagram



4.3 System database schema



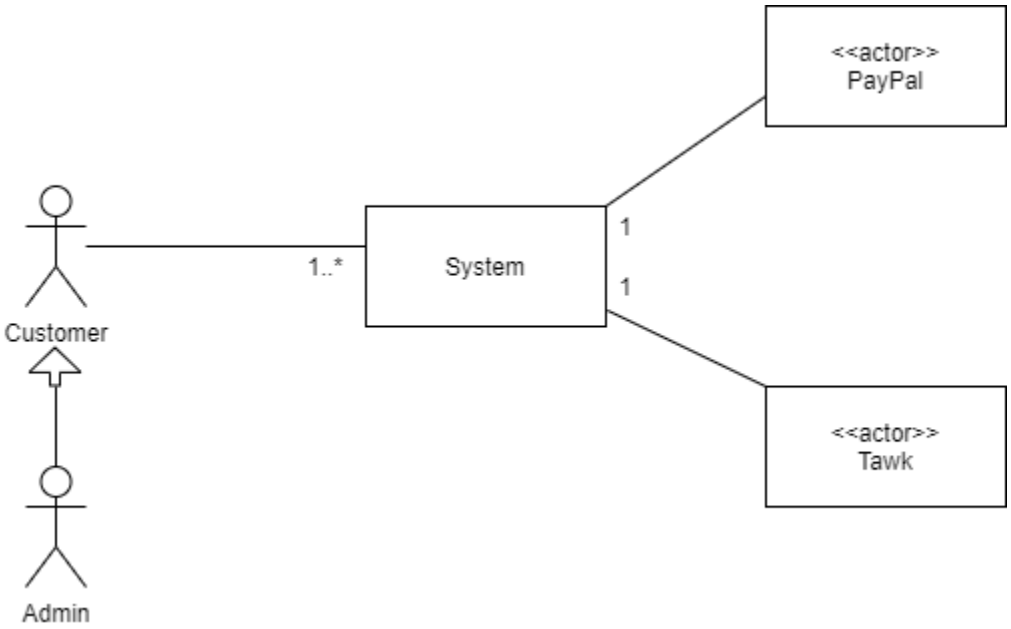
4.4 UML

4.4.1 Actors

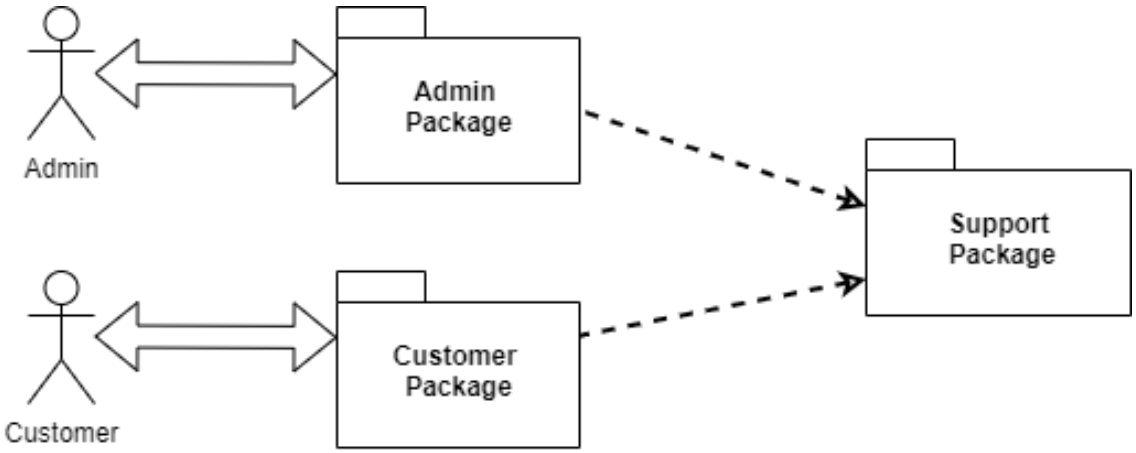
Actor	Actor Type	Actor Description
Admin	Primary	The person who controls everything and can change system properties, can insert products, items, categories and manage orders, invoices, update account settings
Customer	Primary	The person who can view products, look for products, buy products and view his purchases and update his account settings
PayPal	Secondary	The online bank which accepts payments from Customers to a system

		bank account

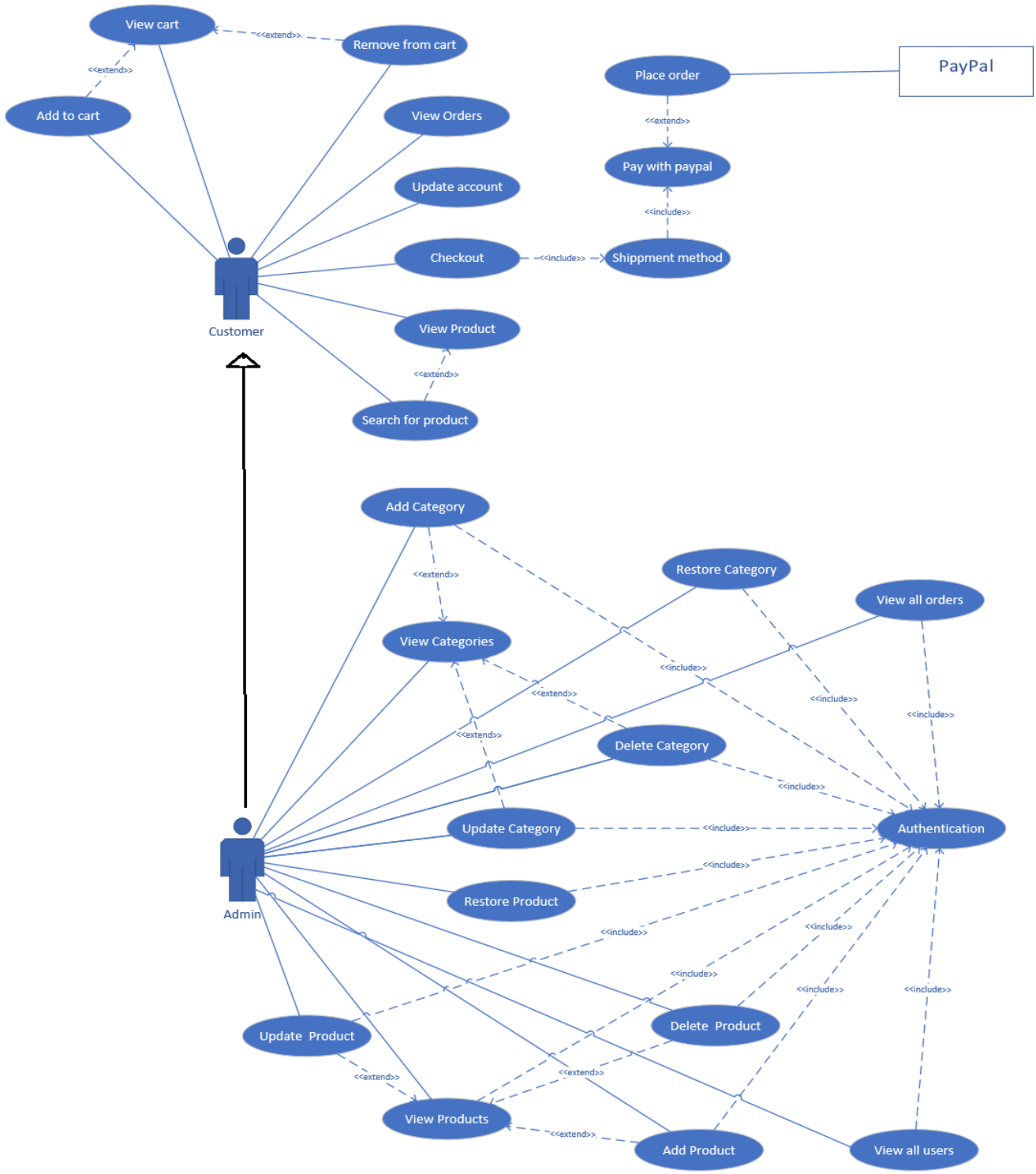
4.4.2 Static context diagram



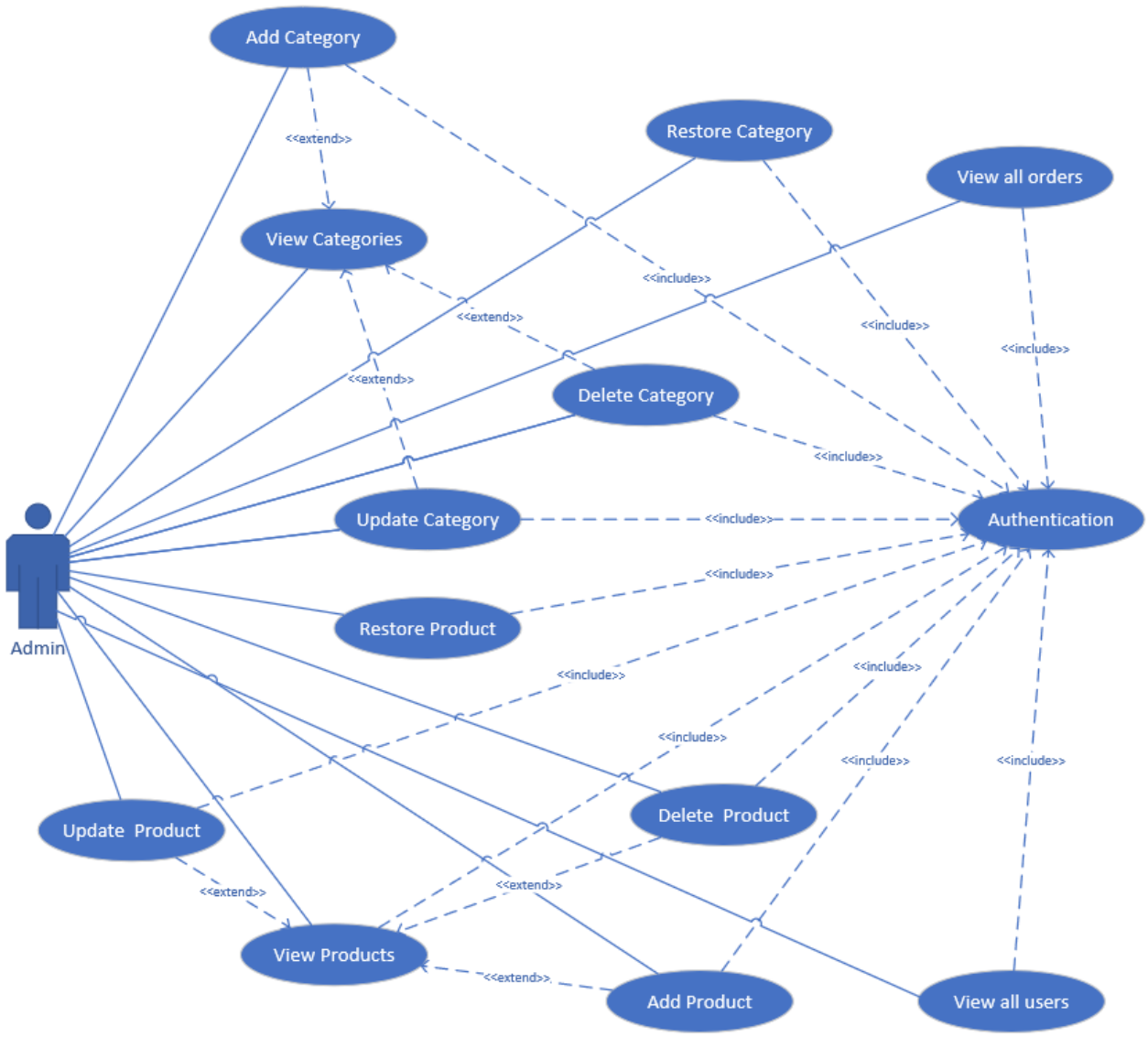
4.4.3 Packages



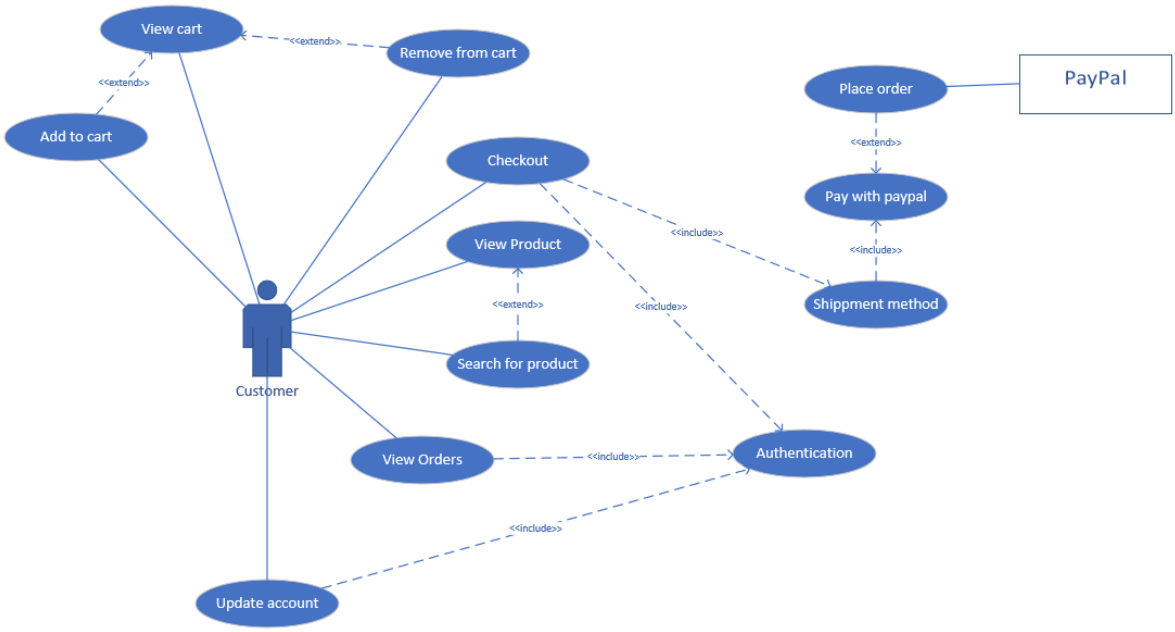
4.4.4 Full Use Case



4.4.4.1 Admin use case



4.4.4.2 Customer use case



4.4.5 Textual description

Identification summary #1
Title: Add Product
Summary: The admin can add a product to the system
Actors: Admin(primary)

Preconditions:

1- Admin must be logged into his account

2- System is running and database is loaded

3- Internet access

4- Admin dashboard is loaded

Main success scenario

Admin	System
1- The admin clicks on the “Add new” button	2- System display modal contains a form and asks the admin to fill the requirements
3- The admin fills the requirements and fields	
4- The admin clicks on the “Add” button	5- System checks and verify if the data meets the requirements
	6- System sends add query to the database
	7- System sends alter to the user that operation was performed successfully

Alternative Sequences:

A1: The admin forgets to fill or mistype requirements

The A1 sequence starts at point 5 of the main success scenario

8- System informs the user that the filled requirements are wrong

9- System asks the admin to re-enter the requirements

Identification summary #2
Title: Update Product
Summary: The admin can update the existing product on the system
Actors: Admin(primary)

Preconditions:

1- Admin must be logged into his account

2- System is running and database is loaded

3- Internet access

4- Admin dashboard is loaded

Main success scenario

Admin	System
1- The admin clicks on the “Update Product” button corresponding to the required product row	2- System display modal contains a form and fills it with the product data retrieved from the database
3- The admin changes the fields he desires	
4- The admin clicks on the “Submit” button	5- System checks and verify if the data meets the requirements
	6- System sends update query to the database
	7- System sends alter to the user that operation was performed successfully

Alternative Sequences:

A1: The admin forgets to fill or mistype requirements

The A1 sequence starts at point 5 of the main success scenario

8- System informs the user that the filled requirements are wrong

9- System asks the admin to re-enter the requirements

Identification summary #3
Title: Delete Product
Summary: The admin can delete product on the system
Actors: Admin(primary)

Preconditions:

1- Admin must be logged into his account

2- System is running and database is loaded

3- Internet access

4- Admin dashboard is loaded

Main success scenario

Admin	System
1- The admin clicks on the “Unlock deleting” button	2- System shows confirmation dialog whether to unlock the ability to delete or not
3- The admin clicks on “Yes” Button	4- System enable the delete button for all products
5- The admin clicks on the “Delete Product” button corresponding to the required product row	6- System sends a request to the database with a delete query
	7- System deletes all un-used serial keys associated with the product
	8- System sends alter to the user that operation was performed successfully and shows the number of deleted serial keys

Identification summary #4
Title: Restore Product
Summary: The admin can restore a deleted product back to the system
Actors: Admin(primary)

Preconditions:

1- Admin must be logged into his account

2- System is running and database is loaded

3- Internet access

4- Admin dashboard is loaded

Main success scenario

Admin	System
1- The admin clicks on the “Recycle bin” drop menu	
2- The admin clicks on the “Restore products” button	3- The system redirects the admin to restore the products page
4- The admin clicks on the “Restore Product” button corresponding to the required product row	5- System returns the product from recycle bin to the system
	6- System returns the deleted serial keys that associated with the product from recycle bin to the system
	7- System sends alter to the user that operation was performed successfully and shows the number of restored serial keys

Identification summary #5
Title: Add Category
Summary: The admin can add a category to the system
Actors: Admin(primary)

Preconditions:

1- Admin must be logged into his account

2- System is running and database is loaded

3- Internet access

4- Admin dashboard is loaded

Main success scenario

Admin	System
1- The admin clicks on the “Add new” button	2- System display modal contains a form and asks the admin to fill the requirements
3- The admin fills the requirements and fields	
4- The admin clicks on the “Add” button	5- System checks and verify if the data meets the requirements
	6- System sends add query to the database
	7- System sends alter to the user that operation was performed successfully

Alternative Sequences:

A1: The admin forgets to fill or mistype requirements

The A1 sequence starts at point 5 of the main success scenario

8- System informs the user that the filled requirements are wrong

9- System asks the admin to re-enter the requirements

Identification summary #6
Title: Update Category
Summary: The admin can update the existing Category on the system
Actors: Admin(primary)

Preconditions:

1- Admin must be logged into his account

2- System is running and database is loaded

3- Internet access

4- Admin dashboard is loaded

Main success scenario

Admin	System
1- The admin clicks on the “Update” button corresponding to the required product row	2- System display modal contains a form and fills it with the category data retrieved from the database
3- The admin changes the fields he desires	
4- The admin clicks on the “Submit” button	5- System checks and verify if the data meets the requirements
	6- System sends update query to the database
	7- System sends alter to the user that operation was performed successfully

Alternative Sequences:

A1: The admin forgets to fill or mistype requirements

The A1 sequence starts at point 5 of the main success scenario

8- System informs the user that the filled requirements are wrong

9- System asks the admin to re-enter the requirements

Identification summary #7
Title: Checkout
Summary: The admin can update the existing Category on the system
Actors: Customer(primary)

Preconditions:

1- Customer must be logged into his account

2- System is running and database is loaded

3- Internet access

Main success scenario

Admin	System
1- The customer clicks on cart icon from navigation	2- System redirects user to cart page and display all cart items with details
3- The customer clicks on pay with PayPal button	4- System verified the cart and sends it to paypal API
	5- System redirects user to PayPal website
	6- System creates order with data received from PayPal api
	7- System sends invoice to the customer email
	8- System redirects user to success page and display notification of order complete

Alternative Sequences:

A1: The customer cancels the request in paypal

The A1 sequence starts at point 5 of the main success scenario

6- System redirects user to cancelled page

A2: The customer has balance more than total of items

The A1 sequence starts at point 3 of the main success scenario

4- System displays button to allow user to add his balance as discount

5- Customer adds balance

6- Customer clicks on pay with balance

7- System subtract total of items from user balance

8- System creates order

9- System redirects user to success page

Identification summary #8
Title: Update account
Summary: The admin can update the existing Category on the system
Actors: Customer(primary)

Preconditions:

1- Customer must be logged into his account

2- System is running and database is loaded

3- Internet access

4- Customer must be in his profile web page

Main success scenario

Admin	System
1- The customer choose what fields to update	
2- The customer clicks on update button	3- System detects which fields has been changed
	4- System applies the changes and update the account details
	5- System display notification to the user of operation status

Identification summary #9
Title: Search
Summary: The admin can update the existing Category on the system
Actors: Customer(primary)

Preconditions:

1- Customer must be logged into his account

2- System is running and database is loaded

3- Internet access

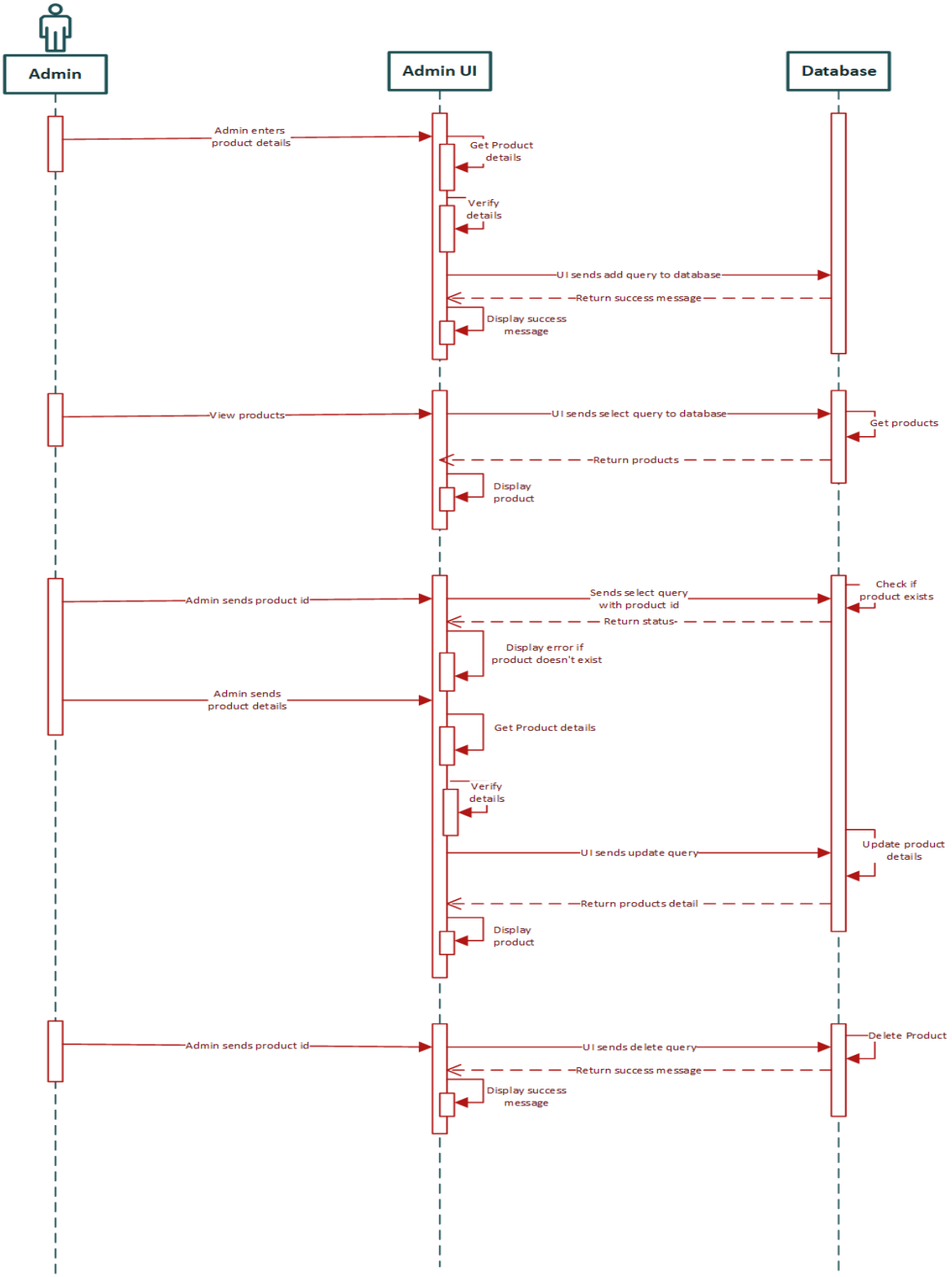
4- Customer must be in his profile web page

Main success scenario

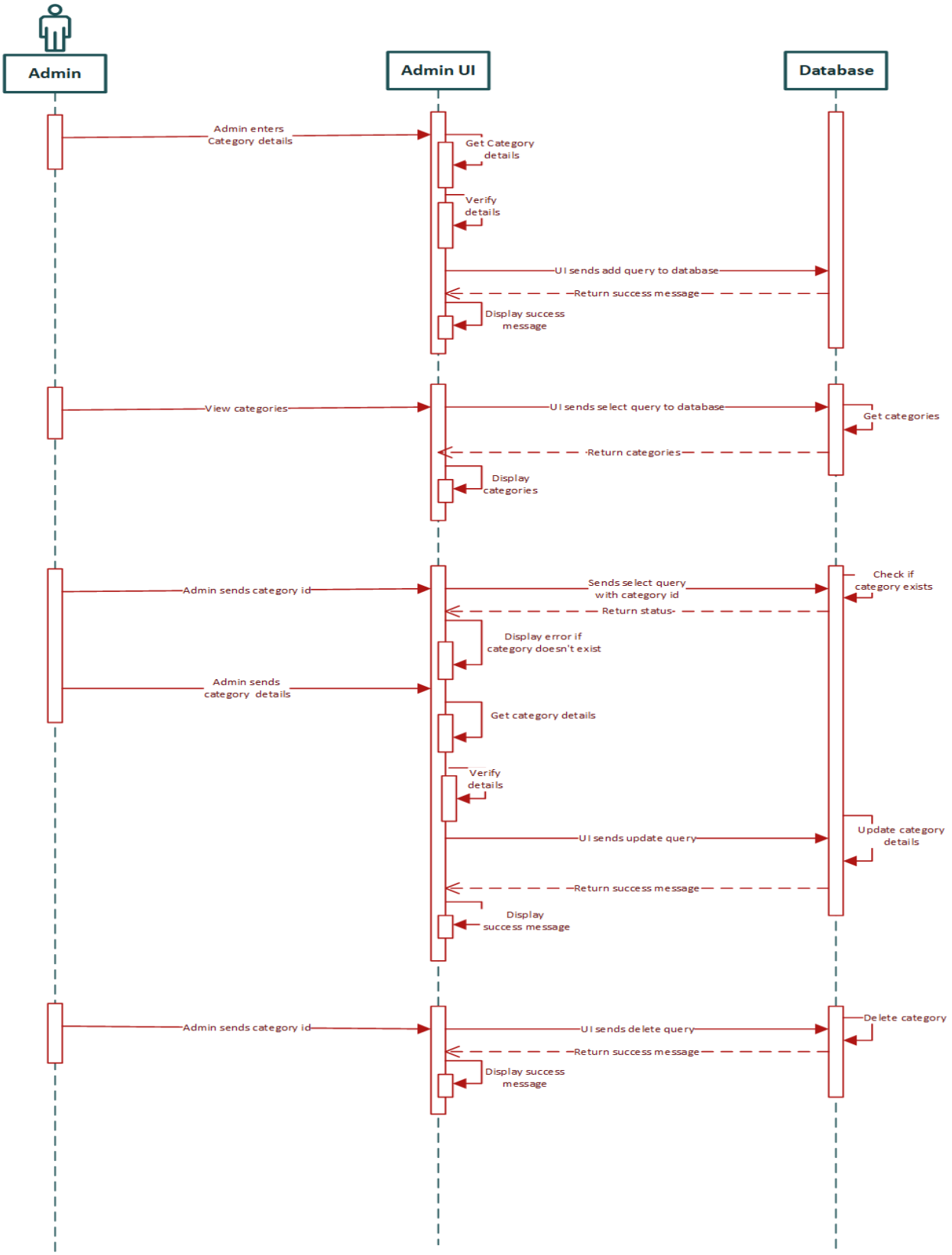
Admin	System
1- The customer choose what fields to update	
2- The customer clicks on update button	3- System detects which fields has been changed
	4- System applies the changes and update the account details
	5- System display notification to the user of operation status

4.4.6 Sequence Diagrams

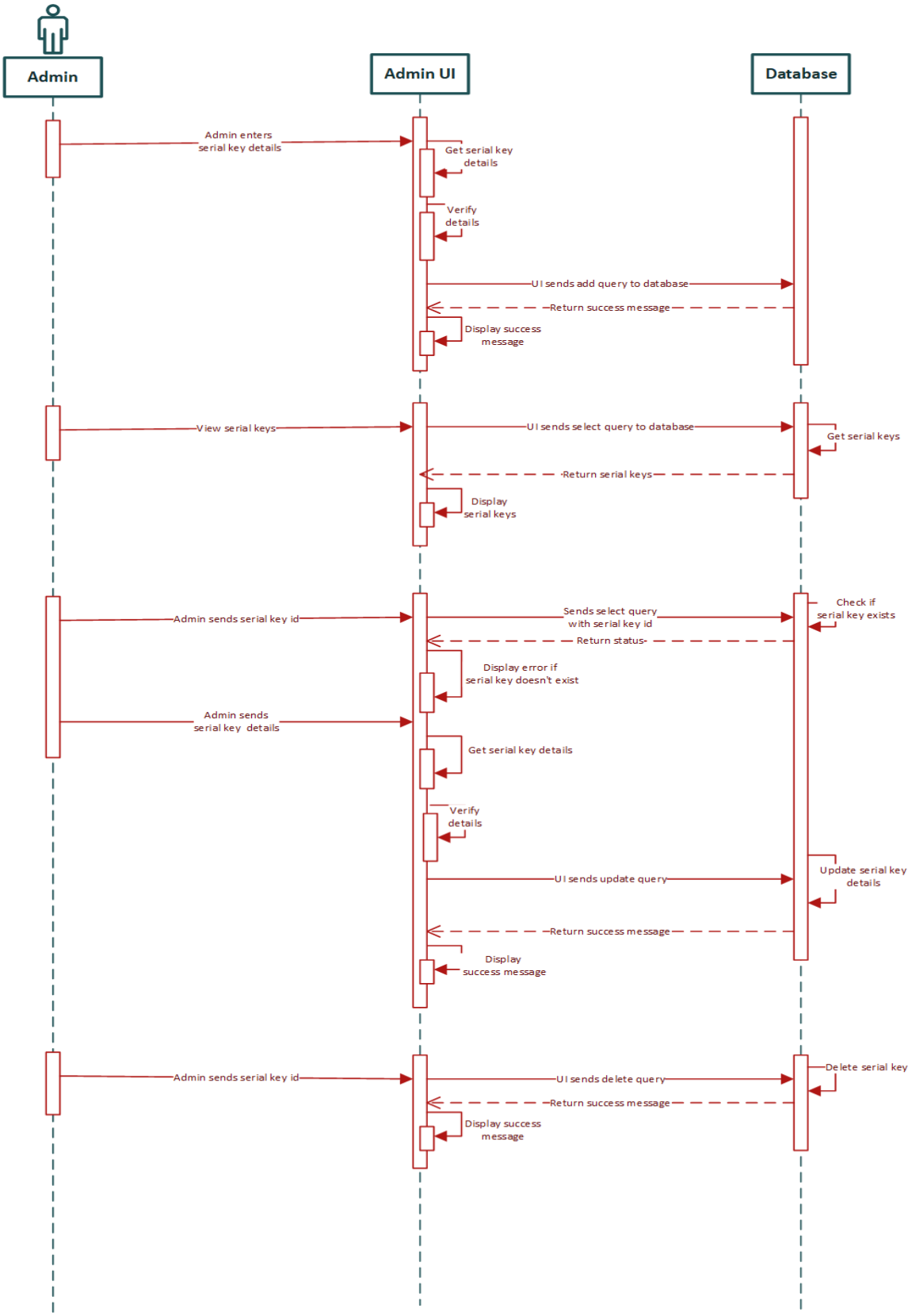
1) Product CRUD sequence



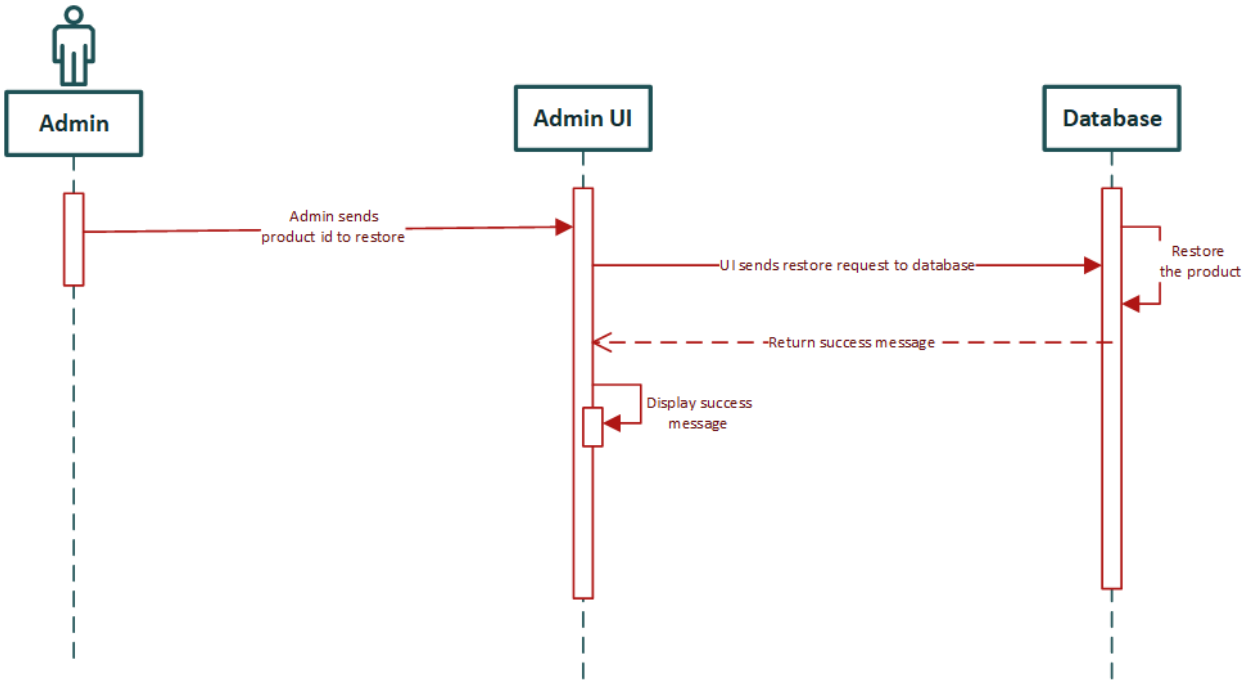
2) Category CRUD sequence



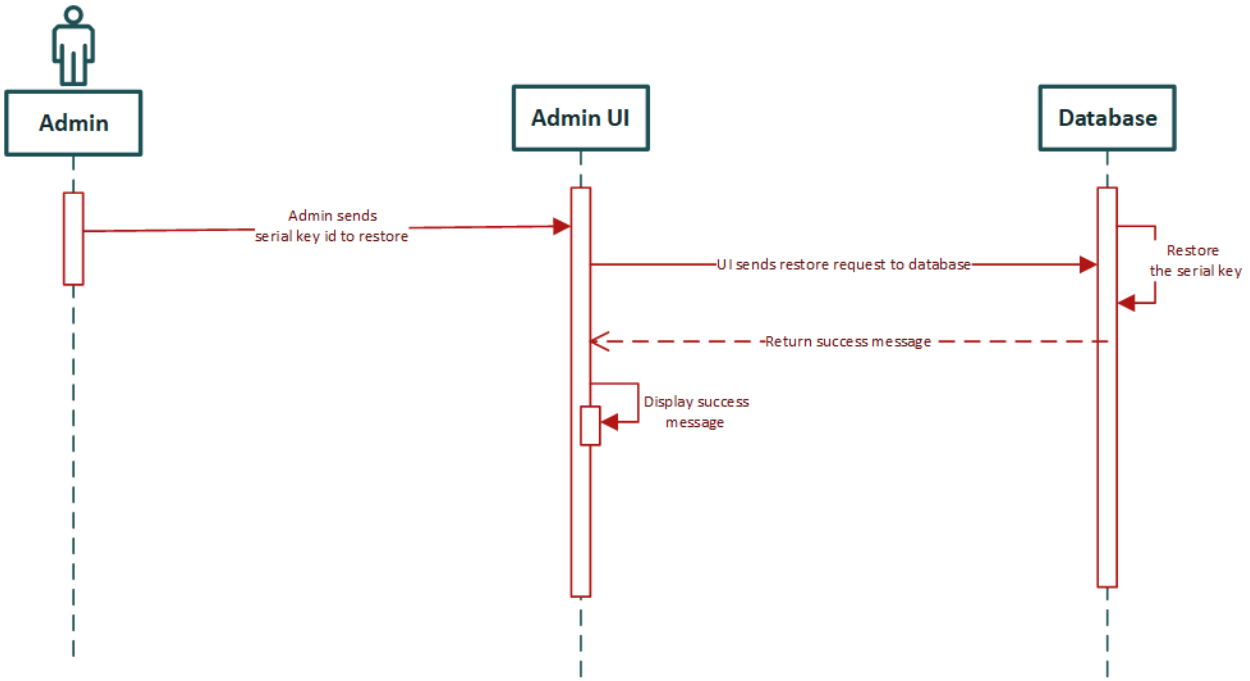
3) Serial key CRUD sequence



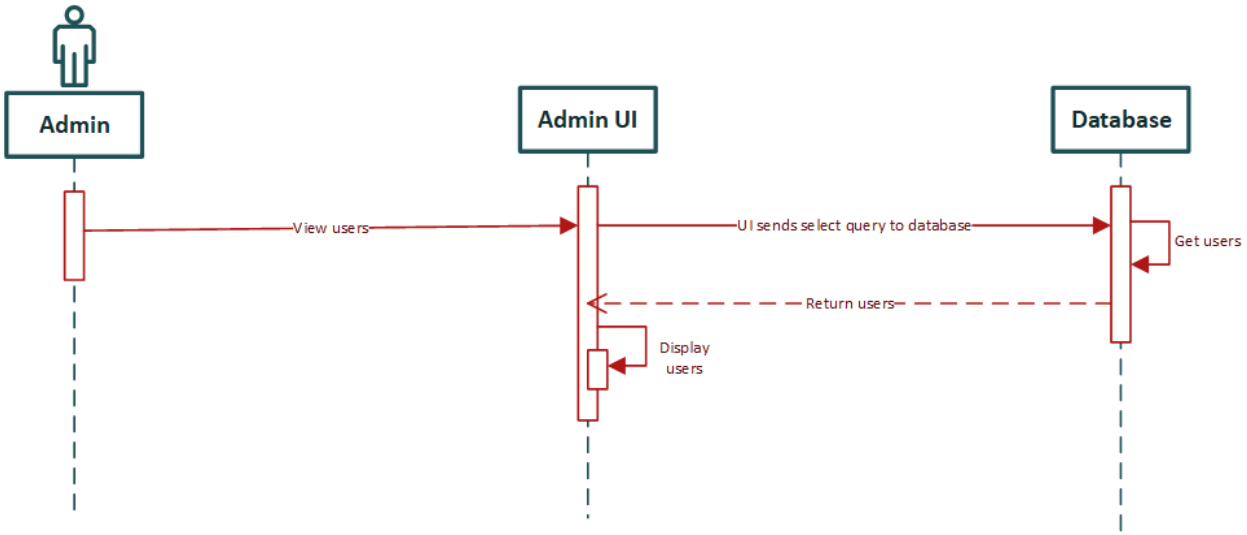
4) Restore Product



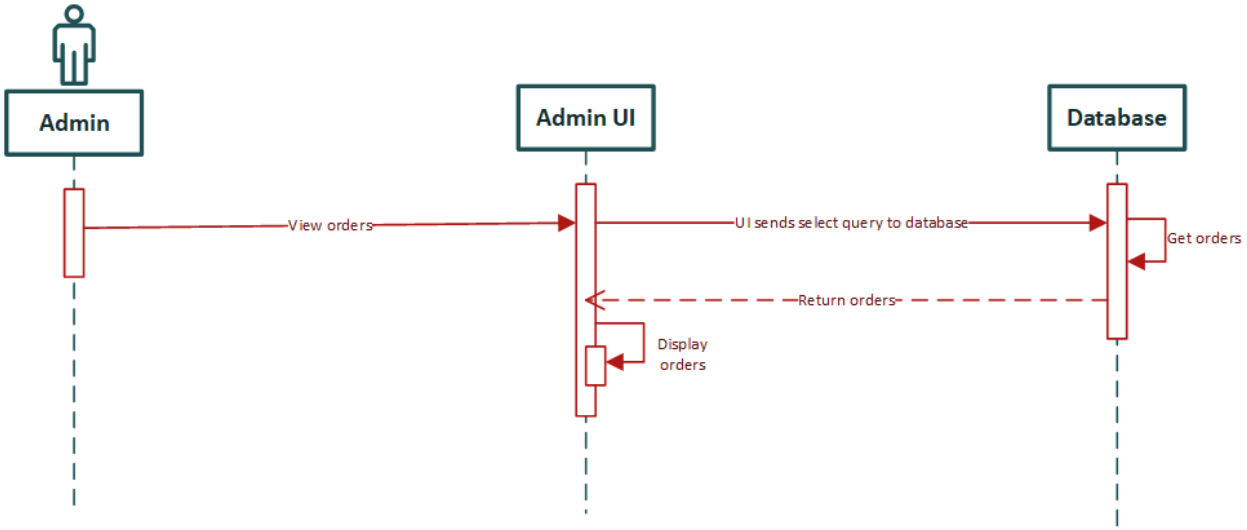
5) Restore Serial key



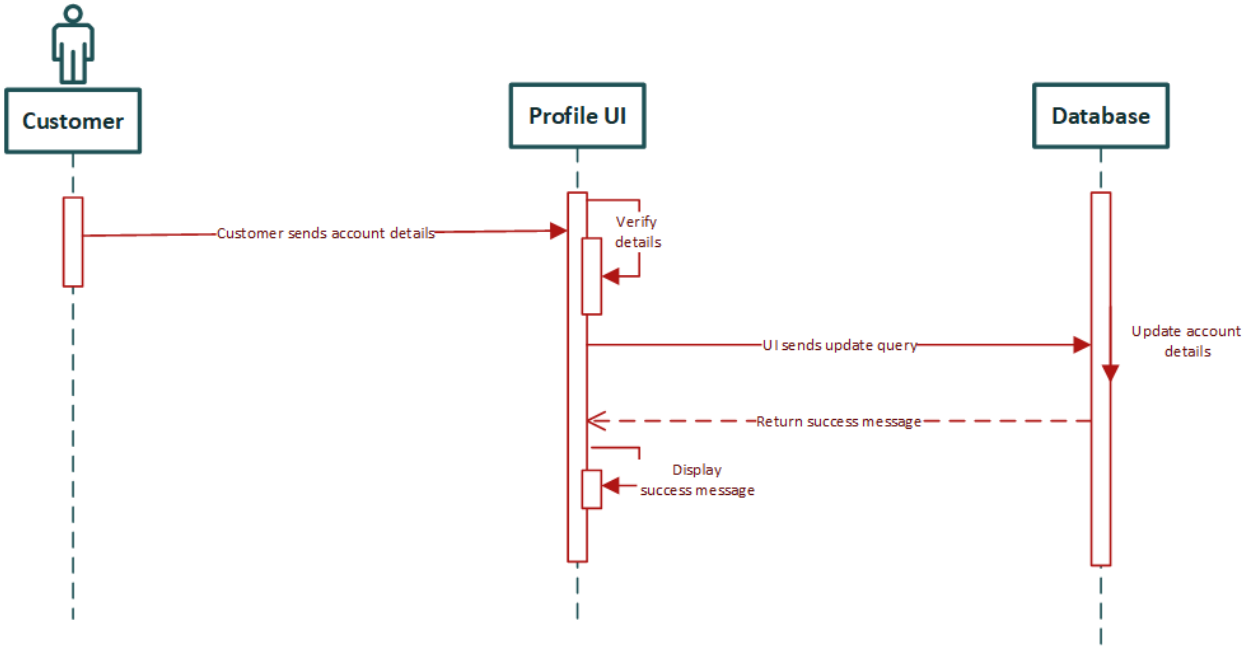
6) View users



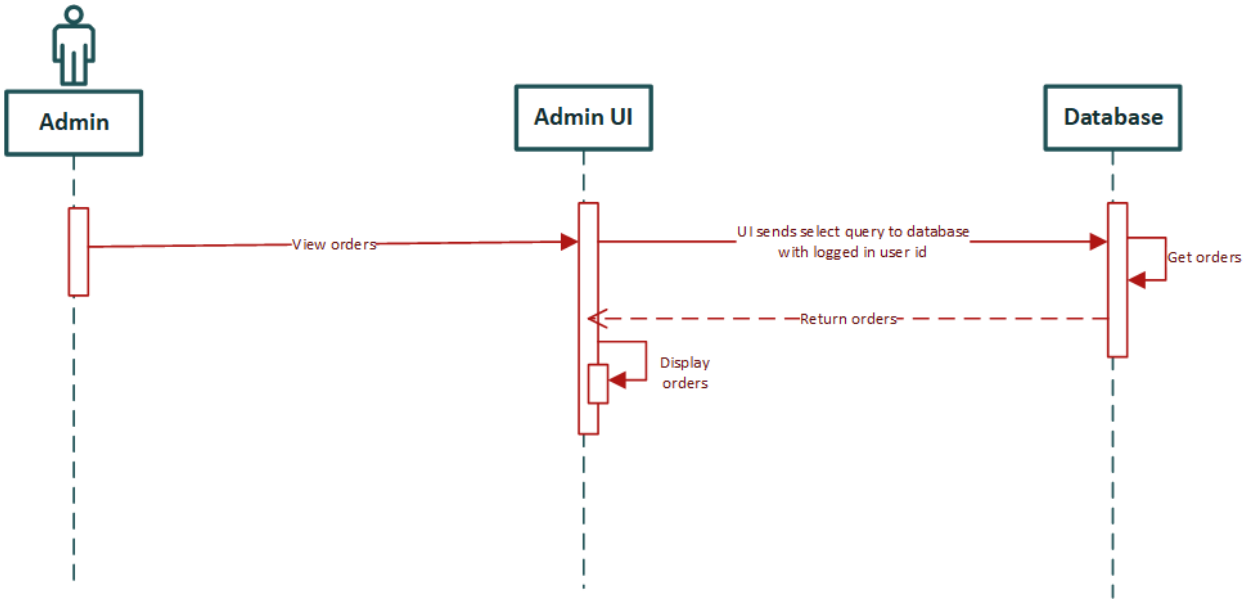
7) View orders



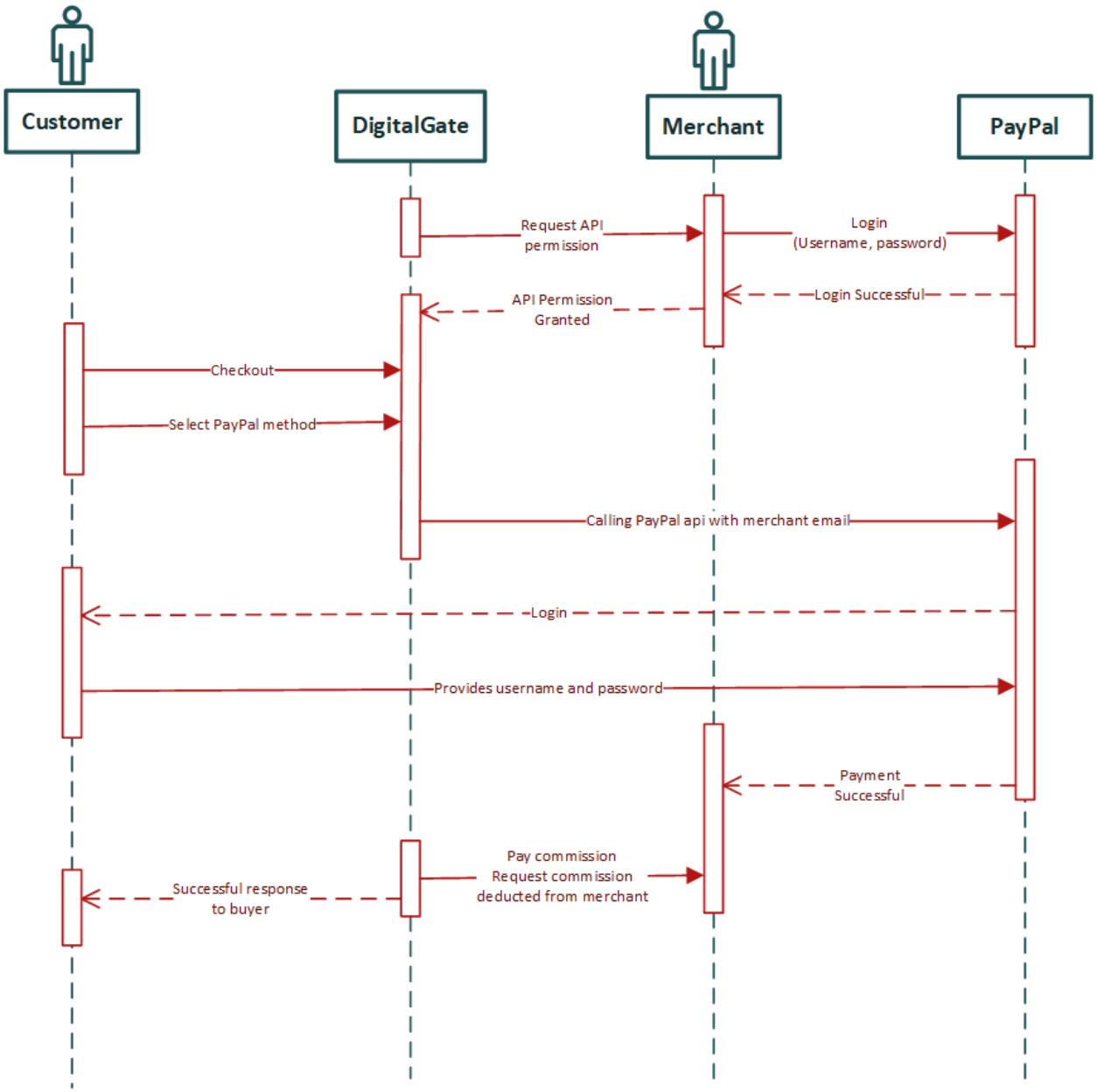
8) Update account



9) View Purchases

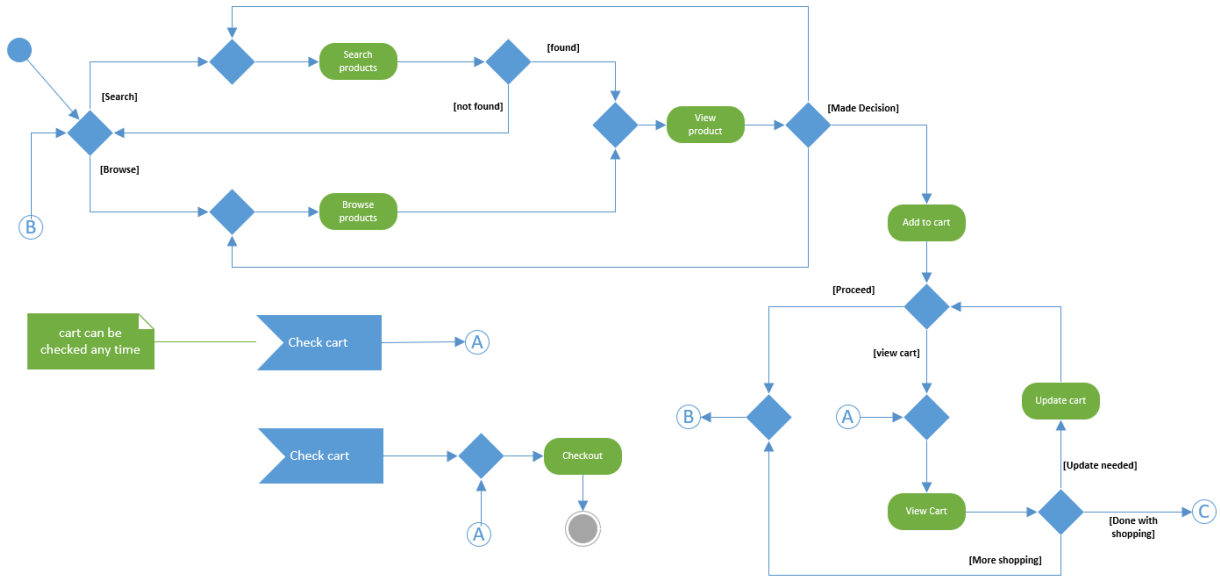


10) Checkout

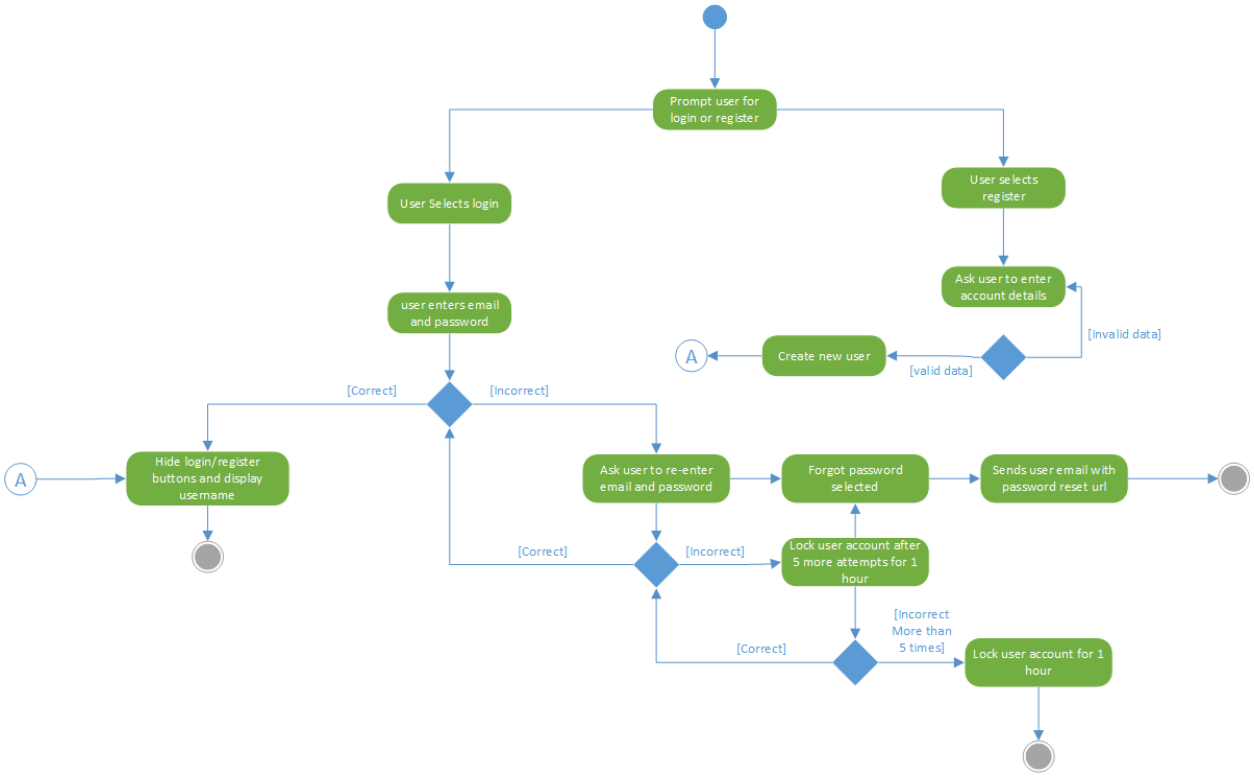

Customer
DigitalGate
Merchant
PayPal
Request API permission
Login (Username, password)
Login Successful
API Permission Granted
Checkout
Select PayPal method
Calling PayPal api with merchant email
Login
Provides username and password
Payment Successful
Pay commission Request commission deducted from merchant
Successful response to buyer

4.4.7 Activity Diagrams

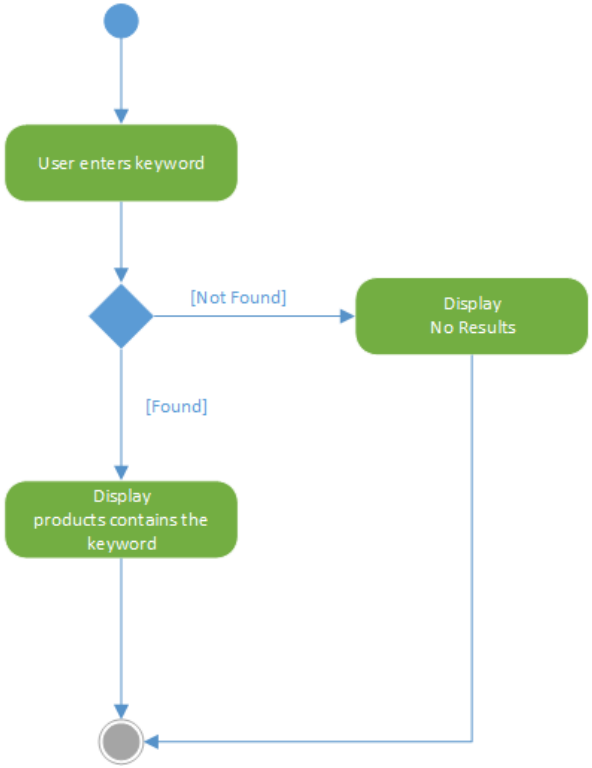
1) Shopping



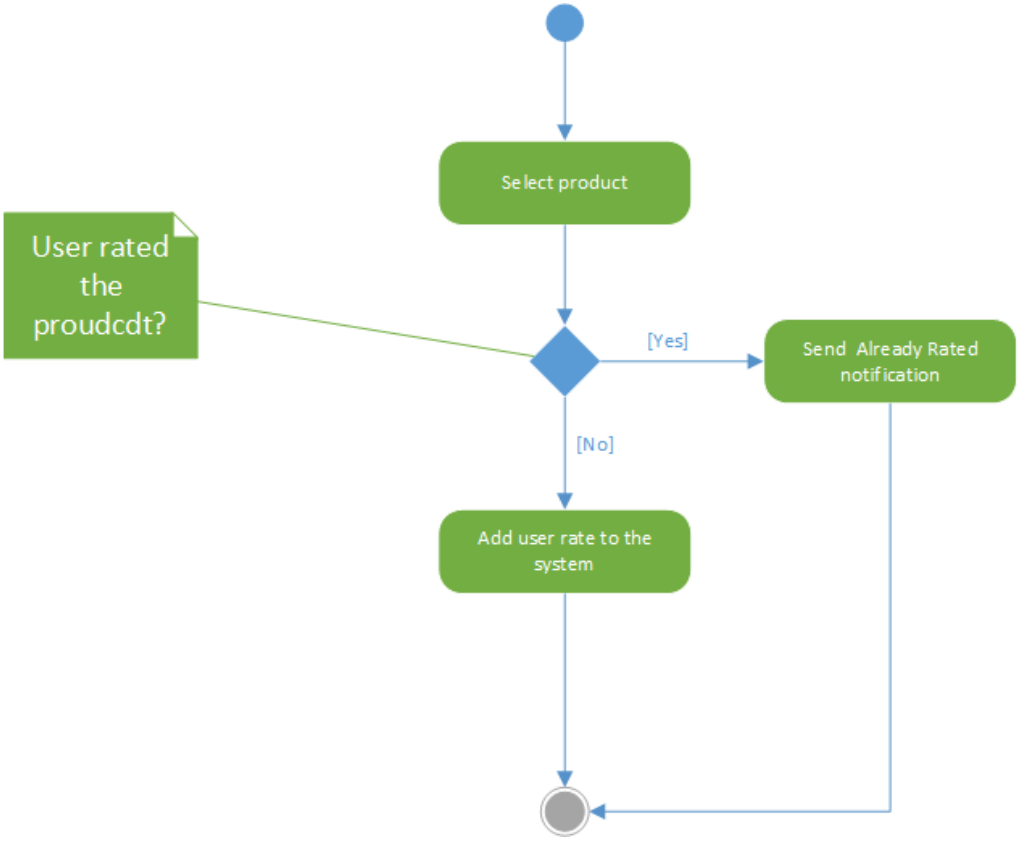
2) Login/Register



3) Search



4) Rate product



CHAPTER FIVE

DESIGN AND SYSTEM IMPLEMENTATION

5.1 Introduction

This section of the document discusses programming languages that we use in the system implementation, why we use these languages and system implementation. the system design stage is considered one of the important stages in the process of building and developing the system, because it gives a complete idea of all parts of the system with illustrations, and it also facilitates the process of building the system properly for programmers, and the tastes and desires of the system users must be taken into account when designing and their desires, ease of use of the system, and taking into account all The categories used for the system, and in this chapter we will cover the design of the system screens and its database.

5.2 Programming Languages

System interfaces are designed with Bootstrap 4, JQuery and Ajax which are libraries used in web development to build interactive elements on websites as well as static infrastructures built with languages like HTML5. And we used PHP Laravel framework for the server-side development

5.3 Why Using these Programming Languages

For javascript the purpose of using JQuery and Ajax are to make it much easier to use JavaScript. JQuery takes a lot of common tasks that require many lines of JavaScript code to accomplish, and wraps them into methods that you can call with a single line of code. AJAX helps us to load data from the server without a browser page refresh. As for styling, we used Bootstrap 4 which is a CSS library to make HTML5 elements responsive and automatically change website layout upon changing device dimensions. The reasons why we choose laravel framework as back-end are:

* Laravel provide quick and functional core that can be extended
* Clean and simple routing
* Effective ORM and database layer
* Easy integration with third-party libraries.
* Active and growing community that can provide quick support and answers
* Supporting unit tests out of the box

* Async queue and background jobs for the long running tasks

5.4 Why Using MYSQL server

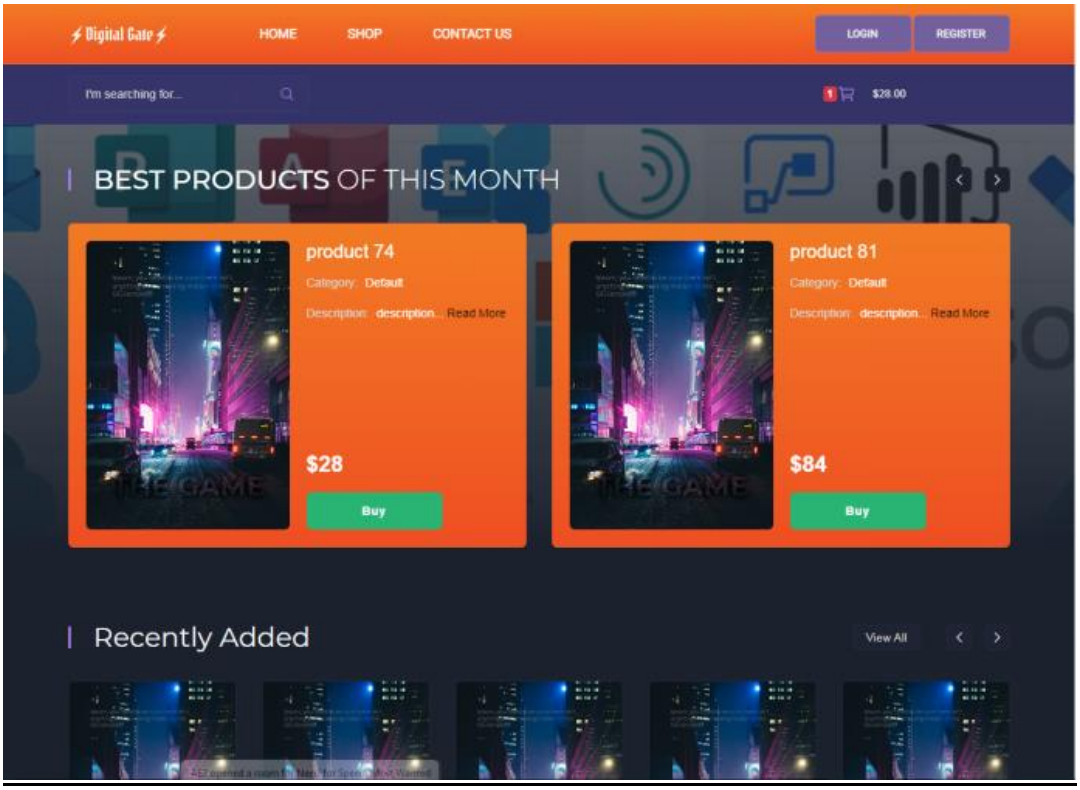
We use MySQL server because it is easy to use, and can run MySQL smoothly on several popular operating systems including Windows, Linux and Mac OS X. And supports a number of storage engines.

5.5 System interfaces

The system consists of several main sections, including: the users section, A system for displaying and arranging products and serials, as well as supervision and registration. In this the main screens section, through which the registration process will be explained, Order products and use cutting arrangements.

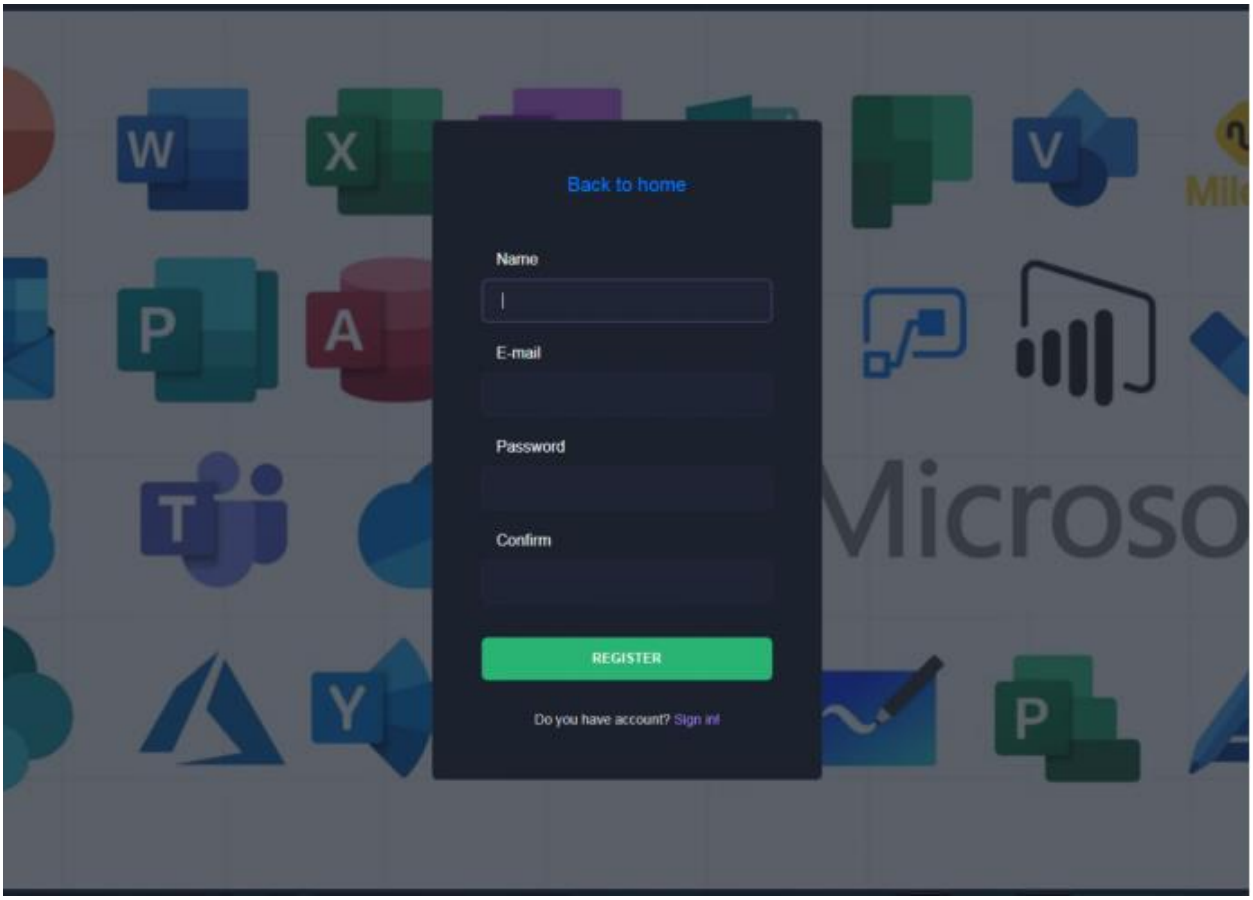
1- <u>Home Page:</u>

The main interface of the system: the user can review all newly added products upon entering the site. As shown in the figure.



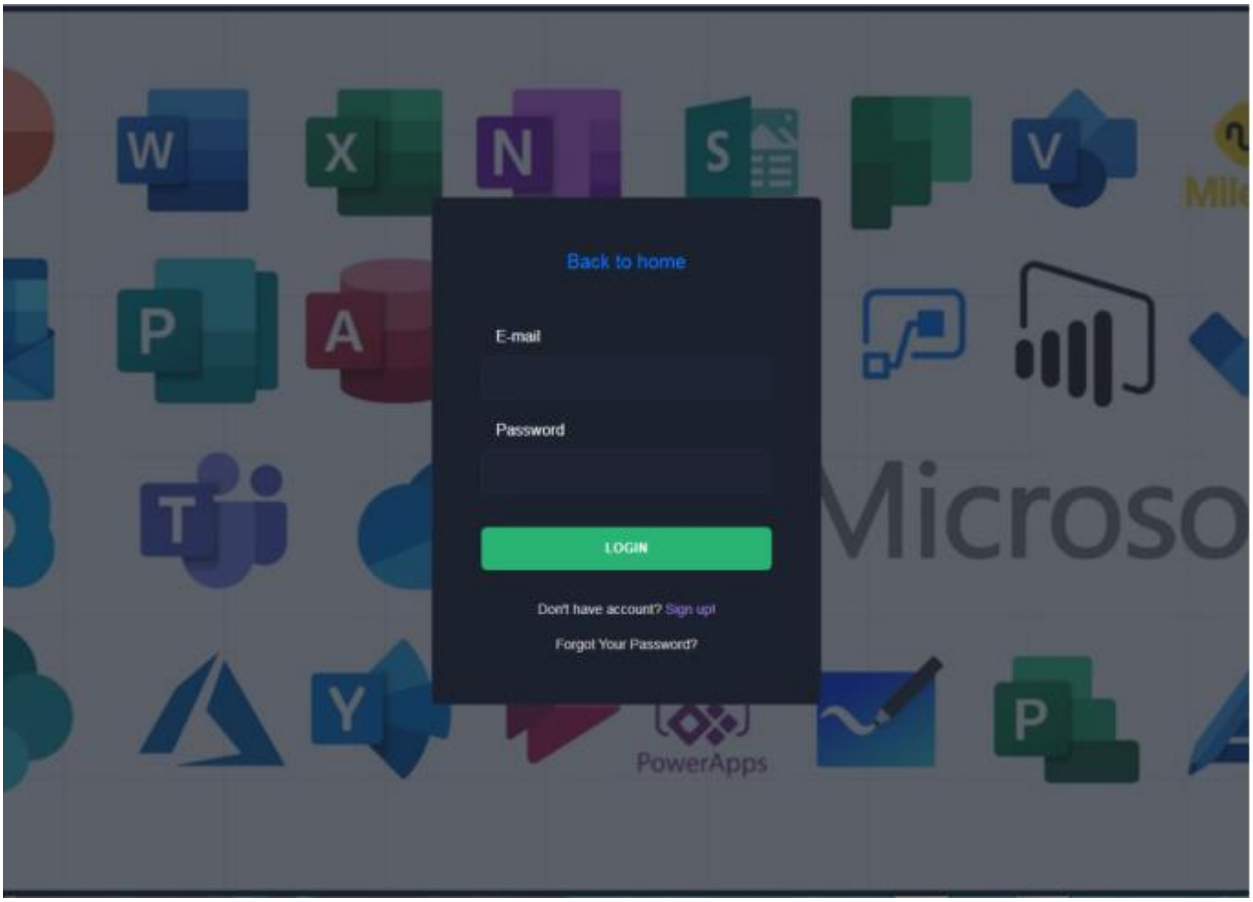
2- Register Page:

User registration section: through it the user can enter his / her data Basic data, so that he can then properly carry out the ordering process. The customer enters his basic data at this stage, and the data is stored in the system. While maintaining the confidentiality of the data he enters. The system works to verify some entries correctly, including: Verify the name, and also the email is only for one user.



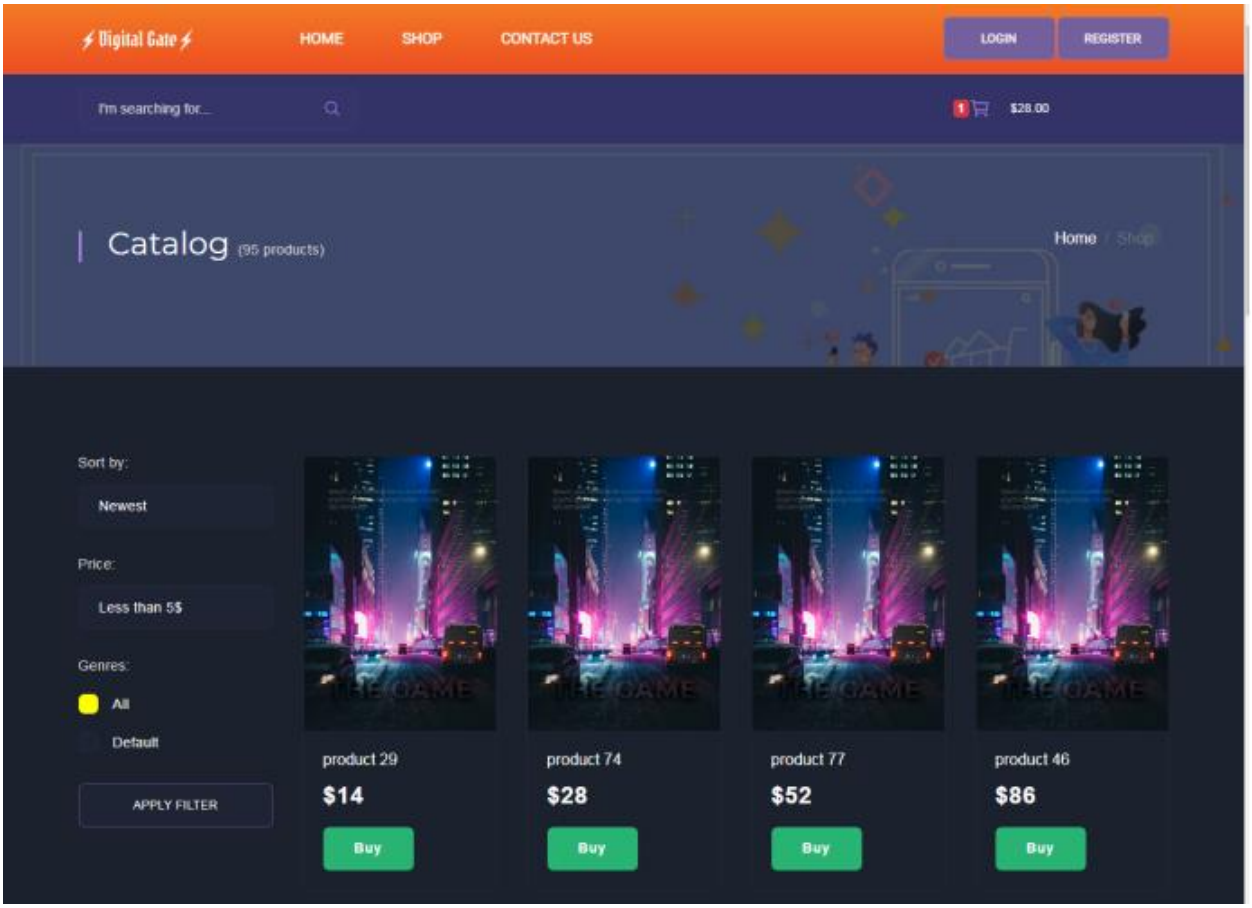
3- <u>Login Page:</u>

Through it, the system administrator and users can log in to the system after entering his email and password, as the system checks the type of user to direct him to his department. Then access to his personal account. As shown in the figure.



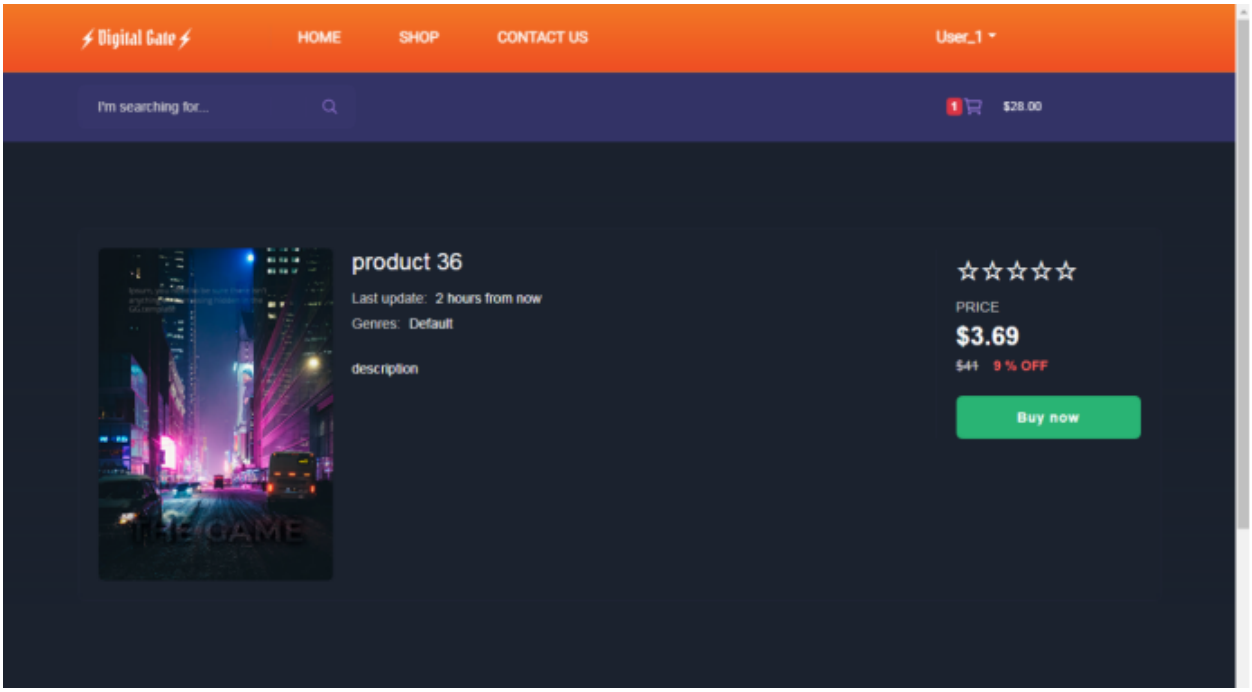
4- <u>Shop Page && Buy Product:</u>

On this page, the user can buy any product. When they click on the "Buy" button, the system adds this product to the shopping cart. Also, there is a notification that shows the number of products added to the cart, whenever you add a product, the counter increases by one.



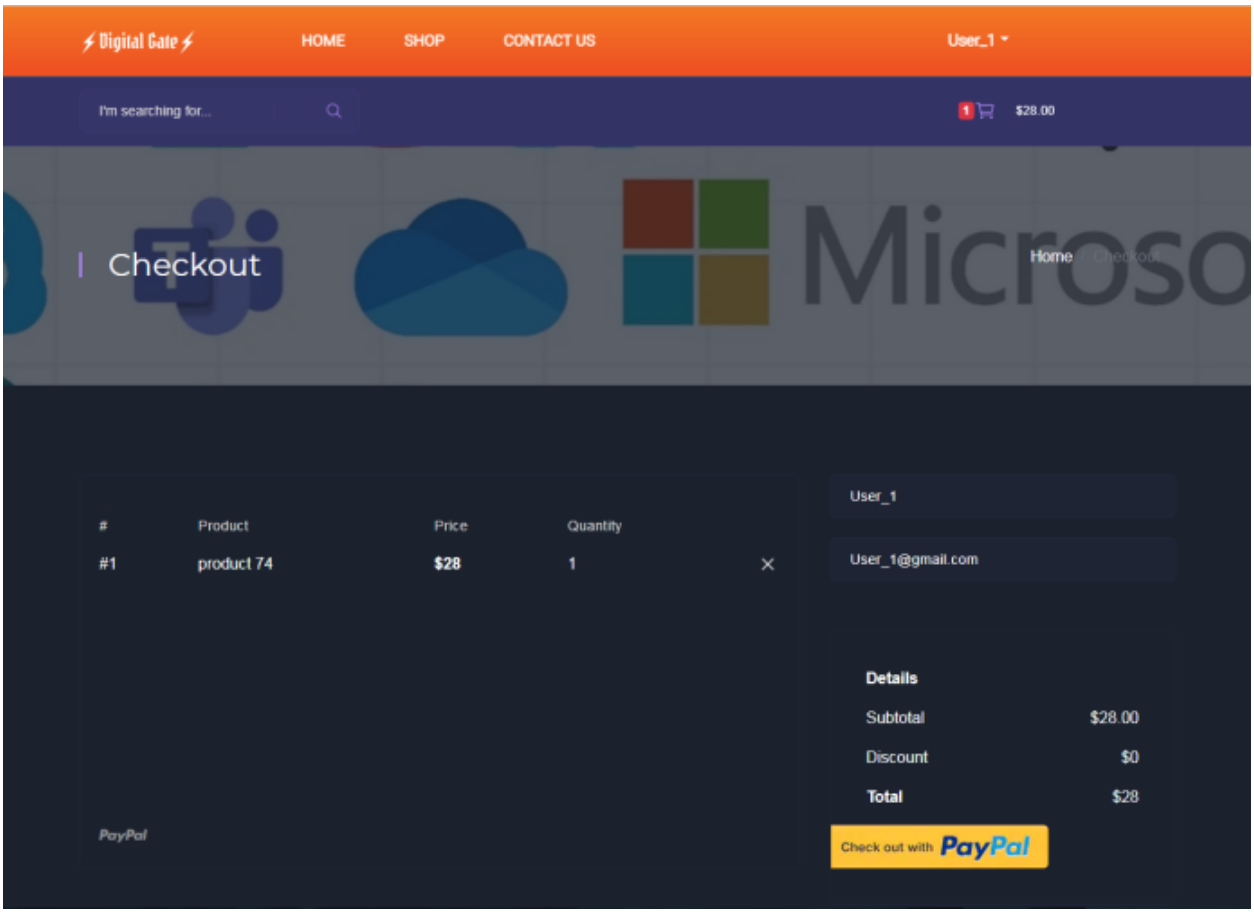
5- <u>Product page:</u>

When entering a specific product, it gives you details about this product, and you can also evaluate the product, and you can also add this product to your cart by clicking on (buy, add).



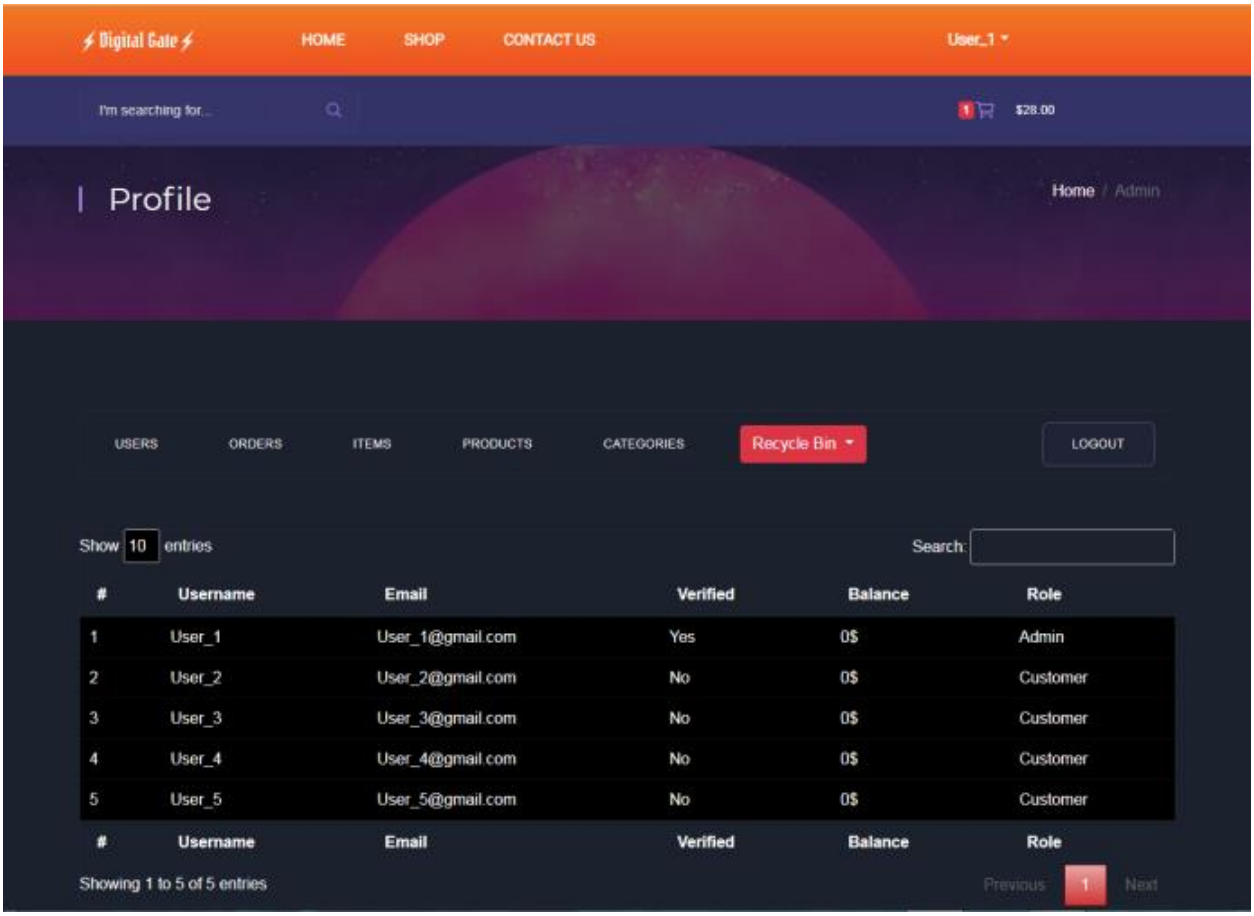
6- Checkout page:

When entering the checkout page, you can buy from there by logging into (paypal) account. After purchase, there will be a discount value for the user when purchasing the product.



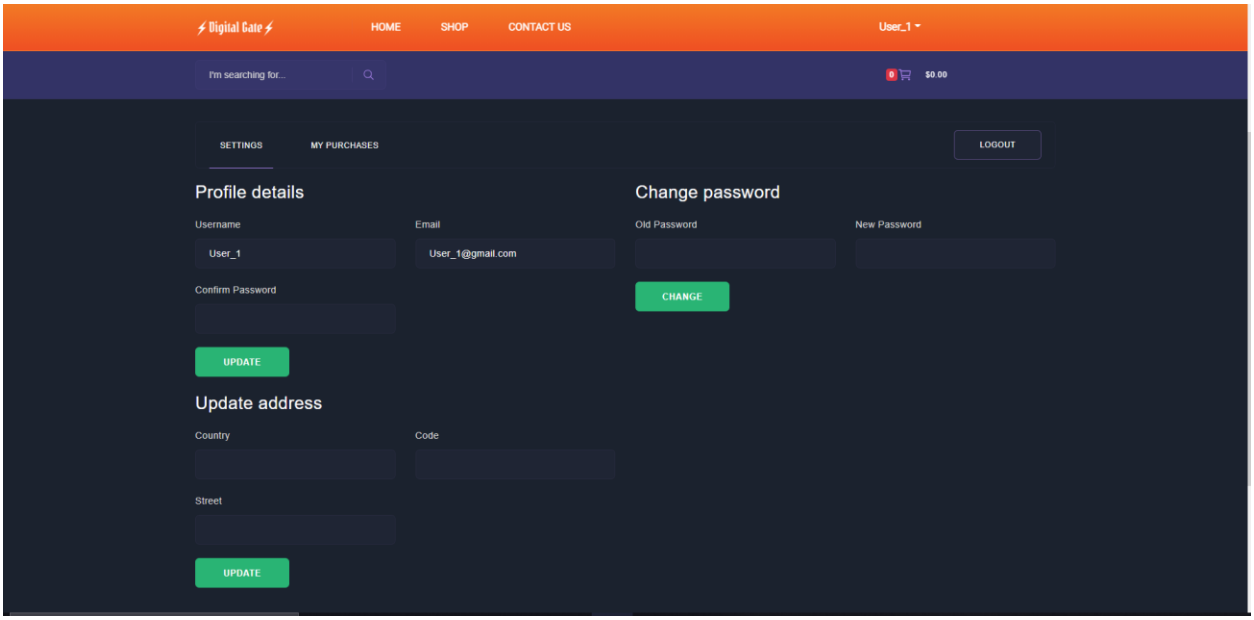
7- <u>Admin Profile page:</u>

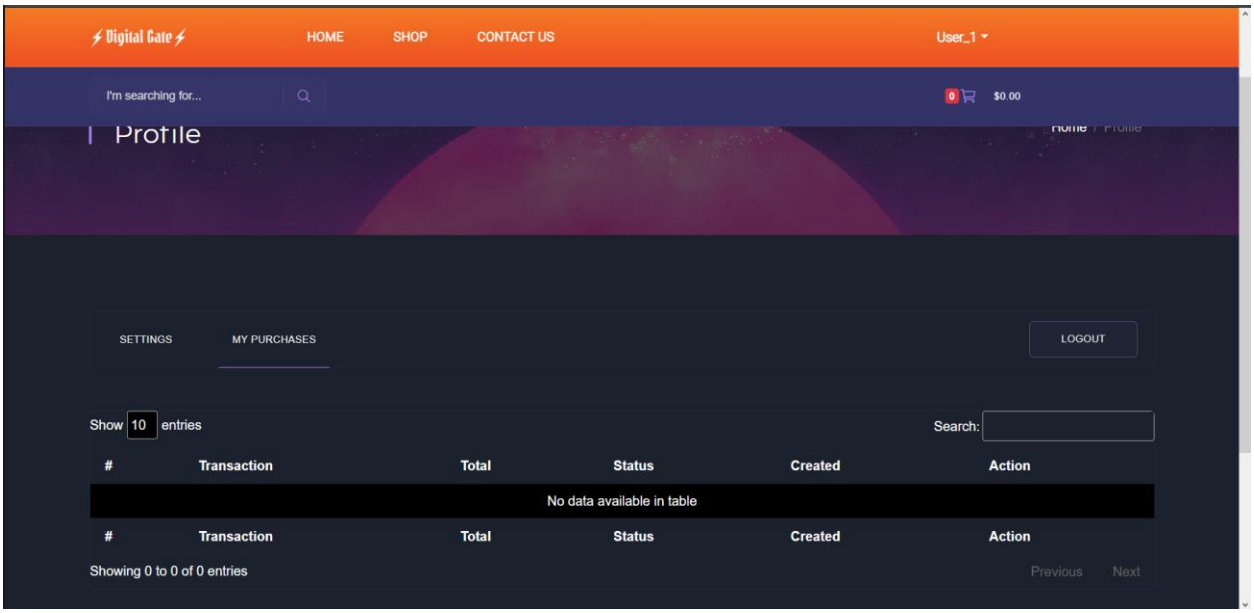
The administrator is the one who adds the products and lists and who deletes them. On the profile page of the administrator there is a group of users who have registered an account on the site, products, items and lists. Also, it can return what you deleted by pressing (recycle bin).



8- <u>Customer Profile page:</u>

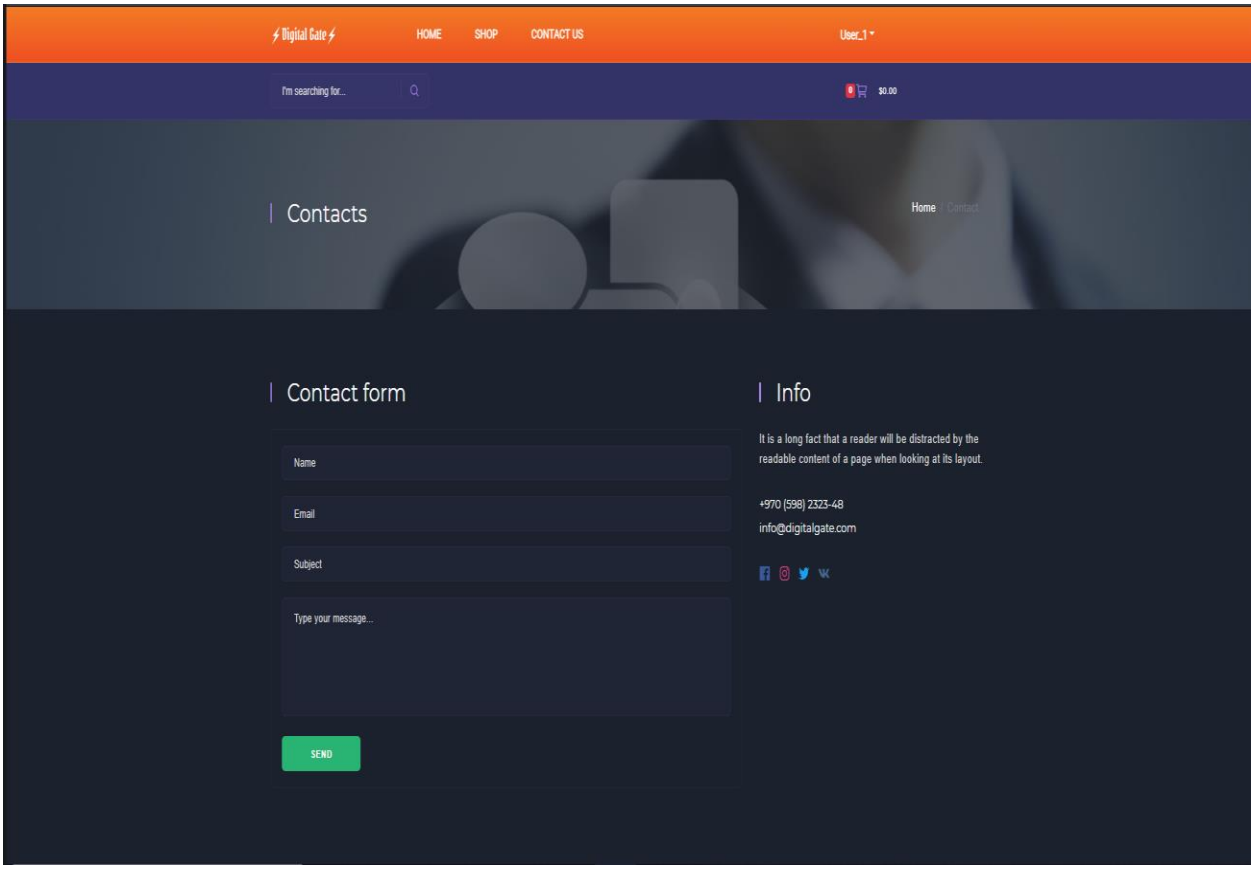
The customer is the one who purchases the products and updates his account information. On the profile page of the customer there are two tabs, one for updating account and the second one is for viewing his orders.





9- <u>Contact page:</u>

Users can send us email from this page.



CHAPTER SIX

System Testing

This section displays two steps for testing specification, the first step is Performing a High-Level Review of the Specification and the second step is Low-Level Specification Test, we will explain them below.

6.1 Testing Specification.

In this section, we will describe the different testing in detail. First, we will pretend to be a customer and try out the system, see if the response will meet desired expectation. Secondly, we will match our system code whether it follows the standard international guidelines. Thirdy, we will check up similar hosted systems on the web.

6.1.1 Pretend a customer

It is important to know who will be using the system and the customers for that system. To communicate with them and understand their needs as well as understanding. Their expectations because it helps clients a lot in setting requirements and fulfilling those requirements. Demanded by users and customers, it also helps test how far it goes. Requirements are met by clients and users.

6.1.2 Research Existing Standards and Guidelines

It is important to know and understand the standards and guidelines related to your project field, and understanding this helps you improve the quality of the system so that the system is in line with the standards and guidelines. Following the standards and guidelines helps a lot in understanding the system and gives it more flexibility if the developers decide to upgrade the system or add new updates.

6.1.3 Review and test similar programs

It is important to look for a similar program as it may be a strong competitor. It should help us think about test situations and test approaches. It also indicates potential problems that may not have been considered. it's also helps us improve many features according to this research.

6.2 Low-Level specification

• Complete: Does the specifications include all the necessary functions that meet the needs of customers?

• Accurate: Is the proposed solution correct? Does it properly define the goal?

• Precise, Unambiguous, and Clear: Is the description exact and not vague? Is there a single interpretation? Is it easy to read and understandable?

• Consistent: Is the description of the feature written so that it doesn't conflict with itself or other items in the specification?

• Relevant: Is the statement necessary to specify the feature? Is it extra information that should be left out? Is the feature traceable to an original customer need?

• Feasible: Can the feature be implemented with the available personnel, tools, and resources within the specified budget and schedule?

• Code-free: Does the specification stick with defining the product and not the underlying software design, architecture, and code?

• Testable: Can the feature be tested? Is enough information provided that a tester could create tests to verify its operation?

ID	Req_id	Req_name	Complete	Accurate	Presice	Clear	Consistent	Relevent	Feasible	Code-free	Testable
1	FR_1	Register	test complete	Yes, because it is working on validation	Yes, understod	its clear	Yes	Yes	Yes, feasible	Yes	its testable
2	FR_2	Login	test complete	Yes, because it is working on validation	Yes, understod	its clear	Yes	Yes	Yes, feasible	Yes	its testable
3	FR_3	View profile	test complete	Yes, it displays all the information	Yes, understod	its clear	Yes	Yes	Yes, feasible	Yes	its testable
4	FR_4	Add product	test complete	Yes, because it is working on validation	Yes, understod	its clear	Yes	Yes	Yes, feasible	Yes	its testable
5	FR_5	Add item	test complete	Yes, because it is working on validation	Yes, understod	its clear	Yes	Yes	Yes, feasible	Yes	its testable
6	FR_6	Add category	test complete	Yes, because it is working on validation	Yes, understod	its clear	Yes	Yes	Yes, feasible	Yes	its testable
7	FR_7	Delete product	test complete	Yes, because it is working on validation	Yes, understod	its clear	Yes	Yes	Yes, feasible	Yes	its testable
8	FR_8	Delete item	test complete	Yes, because it is working on validation	Yes, understod	its clear	Yes	Yes	Yes, feasible	Yes	its testable
9	FR_9	Delete category	test complete	Yes, because it is working on validation	Yes, understod	its clear	Yes	Yes	Yes, feasible	Yes	its testable
10	FR_10	Update product	test complete	Yes, because it is working on validation	Yes, understod	its clear	Yes	Yes	Yes, feasible	Yes	its testable
11	FR_11	Rate product	test complete	Yes, because it is working on validation	Yes, understod	its clear	Yes	Yes	Yes, feasible	Yes	its testable
12	FR_12	View product	test complete	Yes, it displays all the information	Yes, understod	its clear	Yes	Yes	Yes, feasible	Yes	its testable
13	FR_13	Buy product	test complete	Yes, its working a checkout	Yes, understod	its clear	Yes	Yes	Yes, feasible	Yes	its testable
14	FR_14	Send email activation	test complete	Yes, check activation items	Yes, understod	its clear	Yes	Yes	Yes, feasible	Yes	its testable
15	FR_15	View products	test complete	Yes, it displays all the products	Yes, understod	its clear	Yes	Yes	Yes, feasible	Yes	its testable
16	FR_16	View items	test complete	Yes, it displays all the items	Yes, understod	its clear	Yes	Yes	Yes, feasible	Yes	its testable
17	FR_17	View users	test complete	Yes, it displays all the users	Yes, understod	its clear	Yes	Yes	Yes, feasible	Yes	its testable
18	FR_18	View orders	test complete	Yes, it displays all the orders	Yes, understod	its clear	Yes	Yes	Yes, feasible	Yes	its testable
19	FR_19	View categories	test complete	Yes, it displays all the categories	Yes, understod	its clear	Yes	Yes	Yes, feasible	Yes	its testable
20	FR_20	Restore products	test complete	Yes, it works to retrieve all the information	Yes, understod	its clear	Yes	Yes	Yes, feasible	Yes	its testable
21	FR_21	Restore items	test complete	Yes, it works to retrieve all the information	Yes, understod	its clear	Yes	Yes	Yes, feasible	Yes	its testable
22	NFR_22	Usability	test complete	Yes, its accurate, correct define goal	Yes, understod	its clear	Yes	Yes	Yes, feasible	Yes	its testable
23	NFR_23	Reliability	test complete	Yes, its accurate, correct define goal	Yes, understod	its clear	Yes	Yes	Yes, feasible	Yes	its testable
24	NFR_24	Portability	test complete	Yes, its accurate, correct define goal	Yes, understod	its clear	Yes	Yes	Yes, feasible	Yes	its testable
25	NFR_25	Availability	test complete	Yes, its accurate, correct define goal	Yes, understod	its clear	Yes	Yes	Yes, feasible	Yes	its testable
26	NFR_26	Maintianability	test complete	Yes, its accurate, correct define goal	Yes, understod	its clear	Yes	Yes	Yes, feasible	Yes	its testable

6.3 Test scenario

Project name	DigitalGate System
Reference document	DigitalGate System Documentation
Date of creation	01/05/2021
Date of review	20/5/2021

Test Scenario ID	Requirments	Test Senario Description	Importance	No, Test Cases
TS_FR_1	Register	validate if the user is able to enter the system successful	High	7
TS_FR_2	Login	validate if the user is able to change password successfully	High	4
TS_FR_3	Login	validate if the user is able to reset password successfully if he forgot it	High	3
TS_FR_4	View profile	validate if user signing in to account successfuly and able to it	Meduim	3
TS_FR_5	Add product	validate if the admin is able to add product	Meduim	4
TS_FR_6	Add item	validate if the admin is able to add item to product	Meduim	4
TS_FR_7	Add category	validate if the admin is able to add new category	Meduim	4
TS_FR_8	Delete product	validate if the admin is able to delete exists product	High	5
TS_FR_9	Delete item	validate if the admin is able to delete exists item	High	5
TS_FR_10	Delete category	validate if the admin is able to delete exists category	High	5
TS_FR_11	Update product	validate if the admin is able to update exists product	High	5
TS_FR_12	Rate product	validate if the user is able to rate any product	Meduim	3
TS_FR_13	View product	validate if user able to view any product	High	2
TS_FR_14	Buy product	validate if the user is able to pay online through the system	High	3
TS_FR_15	Serial activation	validate if user buy product and able to activate	High	4
TS_FR_16	View products	validate if the admin to view all products	Meduim	5
TS_FR_17	View items	validate if the admin to view all items	Meduim	5
TS_FR_18	View users	validate if the admin to view all users	Meduim	5
TS_FR_19	View orders	validate if the admin to view all orders	Meduim	5
TS_FR_20	View categories	validate if the admin to view all categories	Meduim	5
TS_FR_21	Restore products	validate if the admin able to retreive deleted products	High	6
TS_FR_22	Restore items	validate if the admin able to retreive deleted items	High	6
TS_FR_23	Usability	validate if the system is easy to use by users	High	not-define
TS_FR_24	Usability	validate if the system is easy to understand by users	High	not-define
TS_FR_25	Reliability	validate if all pages of the system download in less than 4 seconds	High	not-define
TS_FR_26	Portability	validate if the system compatible with many browsers(Chroom, Edge,...)	High	not-define
TS_FR_27	Availability	validate if the system is available for users all the time	High	not-define
TS_FR_28	Maintianability	validate if the data existed in system can be retrived if it lose	High	not-define

CHAPTER SEVEN

SYSTEM EVOLUTION

7.1 Conclusion

The “digitalgate” is designed to provide a web-based application that gives an amazing shopping experience by making the purchasing and looking for products a lot easier on the end-user as well as simplify store management procedures for the owner of the store.

The application implements a full recommendation system in order to provide an easy and comfortable way to search for products, where the customer can interact with products, and the application will update the customer recommendation list.

7.2 Future work

Many different adaptations and tests have been left for the future due to lack of time. Future work concerns new proposals to try different methods, or adding new features. There are some ideas that we would have liked to try during the development, but we couldn't complete them so we left them to the future. The following are some of these ideas:

• Build a full REST API in order to allow third-party stores and applications to contact our store and retrieve data.

• Working on a merchant application for Smartphone’s, so that merchants can use the system from any mobile device or computer at any time and in order to facilitate merchants in the process of ordering goods.

• Implementing a dynamic recommendation system that changes itself to suit system resources.