USMAN INSTITUTE OF TECHNOLOGY

Affiliated with NED University of Engineering & Technology, Karachi

Department of Computer Science

B.S. Computer Science

FINAL YEAR PROJECT REPORT

Batch-2017

VIRTUAL REALITY MART

By

Ahmer Irfan	17B-072- CS
Ennas Muhammad	17B-087- CS
Muhammad Ibrahim	17B-068- CS
Sameed Arshad	17B-020- CS

Supervised by

Dr. Lubaid Ahmed
Sir Shahrooz Shamim

ST-13, Block 7, Gulshan-e-Iqbal, Abul Hasan Isphahani Road, Opposite Safari Park, P.O. Box 75300,
Karachi, Pakistan. Phone: 3497827
4-5; 34994305; 34982476; http://www.uit.edu

Declaration

I hereby declare that this project report is based on my original work except for citations and quotations which have been duly acknowledged. I also declare that it has not been previously and concurrently submitted for any other degree or award at USMAN INSTITUTE OF TECHNOLOGY or other institutions.

Ahmer Irfan (17B-072-CS):

Signature: ________________________________

Ennas Muhammad (17B-087-CS):

Signature: ________________________________

Muhammad Ibrahim (17B-068-CS):

Signature: ________________________________

Sameed Arshad (17B-020-CS):

Signature: ________________________________

Date: ________________________

Acknowledgments

We thank and shows our deepest gratitude to all those who have helps us and guided us in the way of developing this idea to a complete product. Some of the individuals to whom we are deeply grateful include our supervisor Dr. Lubaid Ahmed and our Co-supervisor Sir Shahrooz Shamim, for their enthusiasm, guidance, and their unconditional support towards us.

We want to thank our Director Dr. Zahir Ali Syed and Head of Department Mr. Parkash Lohana and Usman Institute of Technology, for giving us an opportunity to learn new things and making us go through this process. Last but not the least to our families especially our parents who had their unconditional love and support throughout our lives and made us what we are today.

Abstract

Virtual Reality mart is a project to increase user experience of online shopping by introducing a new and unique feature in it. Virtual reality mart is a relieving alternative to an extent. It is found to be the most suitable solution to the given situation. A 3-Dimensional mobile based application with the help of this application an individual can achieve shopping experience comparable to real life shopping in the Supermarket.

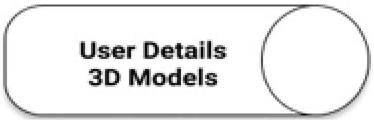
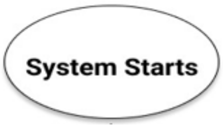
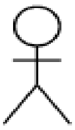
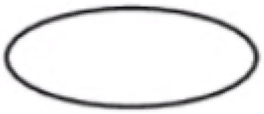
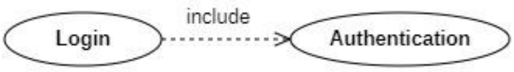
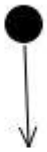
When the user starts the application, they can visualize a 3-dimensional supermarket model with shelves and range of products to choose from. We have used the pictures of real products to compose the product model. Our Virtual Supermarket comprises of thousand variety of products. By using the pictures of the real product, it is possible to provide very clear perception of supermarket of the real and physical world. User will also be able to pay by using relevant means necessary. Multiple payment methods are given in this project to facilitate the users.

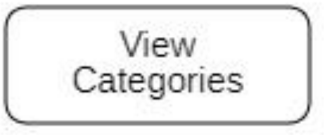
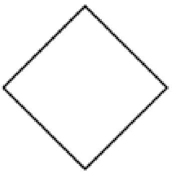
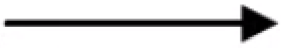
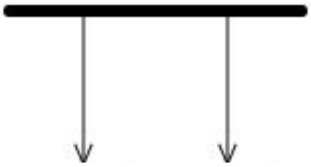
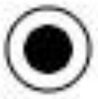
Table of Contents

List of Tables

List of Figures

List of Symbols and Units

	This symbol is used for storing database or data files.
	This symbol represents the start of the system having inputs.
	This symbol represents an actor that performs action in the system.
	This symbol represents a use case that tells about the interaction of an actor with use case or interaction with other use case.
	This represent that a base use case may include another, i.e., included use case behavior is added to the base case user.
	It is the start symbol which represent start of the activity in activity diagram.

	This node is an action that represents every action taken place in the system.
	This symbol represents the decision making.
	This symbol is used for showing the movement of activity from one action to another.
	This symbol is known as fork node. It is used to split the activity into multiple parts.
	Final node indicates the end point of the activities.

CHAPTER 01

INTRODUCTION

1 Introduction

Pandemic has affected the economy of every country and the sufferance is still at its peak. The world is struggling with the sustainability of its economy as well as other resources. All big names in the world of market are subject to change of this very situation and has experienced a major decline in their sales graph. People, who are already managing hard to survive with all the resources they have, are not ready to look for any sort of service or brand which they were crazy about before pandemic. Obviously, it is a huge change which has altered the priorities of everyone. Businesses must find an alternative to survive and launching a new one is again a task which people have never imagined of. When staying home becomes a global slogan and people are advised to work from home, bringing a service or a product in a market meets a challenge at large. Virtual Supermarket, however, is a relieving alternative to an extent. It is found to be the most suitable solution to the given situation.

The virtual supermarket is 3-Dimensional mobile based application. With the help of this application an individual can achieve shopping experience comparable to real life shopping in the Supermarket. When the user starts the application, they are able to visualize a 3-dimensional supermarket model with shelves and range of products to choose from. We have used the pictures of real products to compose the product model. Our Virtual Supermarket comprises of 1000 variety of products. By using the pictures of the real product, it is possible to provide very clear perception of supermarket of the real and physical world.

To navigate the virtual supermarket cursor keys are used. To put the product in the basket user can click the product to the shopping trolley. User can easily keep track of the products present in the trolley with the help of the cart button provided on the top. When the user has finished shopping, they can click on the “Go to counter” button for the checkout. At the checkout counter they won’t be paying with real cash in the form of money instead, they will be asked for the payment method which includes Cash on delivery, jazz cash, Easy paisa. The original receipt is also generated at the counter. So far, we have submitted the prototype. In this application admin plays the key role. The individual possessing the admin rights can login in the admin panel, and can view, remove, and update products when needed.

The user will first signup in the application and then provide the credentials to login to the application. Then the provided passwords will be validated for authentication and if the user is authenticated then the user is allowed to proceed for shopping otherwise the user will enter the credentials again. After the successful login, the user can start shopping the by viewing the products placed in the shelves. The user will click the product in the basket and will use the cursor keys to navigate through the virtual supermarket. The list of the items present in the basket is updated each time the user adds or removes the product from the shopping trolley. Once the user completes the shopping, it can proceed to the counter for checkout. The receipt is generated automatically. Then instead of paying in the real cash the user will be asked for the paying method they would prefer. The products will be later sent to the user's directed place.

According to our project the variables that play important role in our project's performance are internet stability, high internet bandwidth, good delivery service, great specification mobile. We have divided the project into five main tasks namely project definition, work plane, actor use case diagram, system deployment diagram, user interface, software design, database design, develop system module, database connection, project development and completion, perform system testing, deployment of the project.

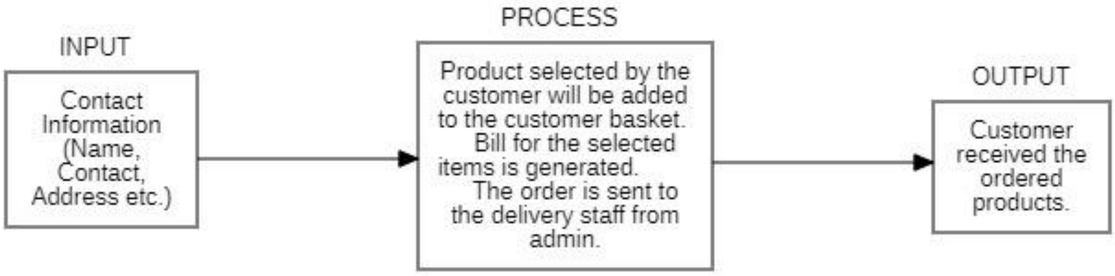

INPUT
Contact Information (Name, Contact, Address etc.)
PROCESS
Product selected by the customer will be added to the customer basket.
Bill for the selected items is generated.
The order is sent to the delivery staff from admin.
OUTPUT
Customer received the ordered products.

Figure 1.1: System Diagram

As shown in Figure 1.1, User inputs the details such as name, contact no, address etc. for registered in this application first time. When user checkout after adding products in cart the processing has been started for generating bill according to payment method of customer details and then order is ready to delivery. In output customer received the order [1].

In chapter 1 Introduction we have provided a detailed background of our project. At the beginning you will be able to get the insight of the problem we are resolving using our project. After that we will give a brief idea about what approach will be followed by us in order to resolve the problem. After providing with the basic structure of the idea discussed the working of the software and for the reference, we have also attached the system diagram.

In the chapter 2 Background and Literature Review we have analyzed the similar projects which are based on the idea of e-commerce but does not provide real life shopping experience. The similar applications we have researched are Daraz, Carrefour and Retailo. For each similar application we have provided the information of their main features, their deployment strategies and their estimated cost. In this chapter we have also discussed the libraries we have used in the project. For each library used we have also provided some detail of the function of the library.

In the chapter 3 Aim and Statement of Problem we have discussed the aim and statement of the problem along with the solution we are providing to solve the discussed problem. When moving forward we have written the modules of our application. After discussing about the project modules, we have written the aim and objectives of our project.

In the chapter 4 Software Analysis and Requirements we have discussed the fact-finding techniques. After the fact-finding techniques, we discussed about the software analysis of our project. After that we have discussed the requirements of our project. In the requirements we have discussed both the functional and non-functional requirements in the software development.

In the chapter 5 Software Design and Modeling we have presented the process flow diagram of our project with explanation of the diagram. After that we attached the UML diagrams for our project. The UML diagrams we have discussed are use case diagram, class diagram, sequence diagram, communication diagram, activity diagram, ER diagram with the explanation of each diagram.

In the chapter 6 Algorithm Analysis and Code Complexity we have discussed the algorithm and the complexity in terms of the best case and the worst case. In the chapter 7 Implementation we have discussed the component diagram, deployment diagram, state diagram with the explanation of each diagram. After the UML diagrams we have attached the screenshots of the GUI of both the customer side and the admin side. After the screenshots of the GUI have attached the code of our application.

In the chapter 8 Testing we have performed the black box testing and the white box testing on different modules of our projects. In chapter 9 Conclusion we have presented the conclusion of our project. In chapter 10 Future Work we have discussed some of the suggestions by implementing which, an individual can increase the features of this virtual supermarket project.

In the chapter 11 Achievements we have discussed the details of the contact person in the Tradeways. We have also discussed that we presented this idea to Tradeways and the commitment they have made with us. In chapter 12 Plagiarism Report we have attached the plagiarism report of our project FYP book using the platform Turnitin as a plagiarism checker. In chapter 13 References we written the references in the IEEE format.

CHAPTER 02

BACKGROUND & LITERATURE REVIEW

2 Background and Literature Review

Currently the famous virtual Supermarkets are developed by Walmart, Tesco, Yihaodian. While talking about Tesco, their Mission was to become number one in Korea without increasing number of stores. After a thorough research they came up with an idea " let the store come to people" and created virtual stores which was first setup up at the subway station although it was virtual, but it gave it uses real life experience of the shelves in a supermarket. The only difference was that to was used with a smartphone in the user’s hand. They use their smartphone to scan the QR code of the product to add it into the basket and can select the payment method they would prefer. The result revealed to the Tesco was that people could shop at Tesco home plus wherever they go without having to visit the actual Store. However, our proposed solution works in a manner as the user will have to sign up and login. It will be very easy for the user to navigate in the supermarket model using the cursor keys. The user is notified each time the item is added or removed from the trolley and the total items in the basket. The users can select the payment method according to their ease. Finally, the ordered items are delivered to the customer fed address. By this they can have an easy solution for shopping and that too from the comfort of their home [2].

2.1 Similar Applications

2.1.1 Daraz

2.1.1.1 Main Features

Daraz is one of the largest e-commerce stores providing millions of customers their desired products by placing order online. These orders received by Daraz are forwarded to the relevant registered Daraz sellers [3].

2.1.1.2 Deployment Strategies

Daraz connects multiple sellers to the customers. Once order is received these sellers are liable to deliver the ordered products to Daraz Collection. Then Daraz deliver these goods to the customers, collect payment and transfers after deducting its commission to the relevant seller [4].

2.1.2 Carrefour

2.1.2.1 Main Features

Carrefour is an international supermarket online store delivering to millions of customers daily. They are based in middle east and has opened their services to customers in Pakistan with hyper star on of the largest supermarkets in Karachi. They aim to target retail and general customers both.

2.1.2.2 Deployment Strategies

The main strategy followed by Carrefour is that they deliver the items to the customers by taking supplies from hyper star. They also allow customers to order online and customers and pick up their order on specified time while maintain their inventory in hyper star warehouse [5].

2.1.2.3 Cost:

The Development of Carrefour application is estimated to be around $10,000 to $50,000 [6].

2.1.3 Retailo

2.1.3.1 Main Features

It is an online business to business with a motive of technological supercharge the retail supply chain in a specific region. It allows the local retail market to upgrade themselves in terms of technology and gain advantage to compare and purchase the best products from their device.

2.1.3.2 Deployment Strategies

The strategy of the **Retailo** is that they connected the retailers on one platform. The firm collects the goods from the suppliers. Theses goods are then stored at **Retailo** warehouse with complete protection and care. Which are then supplied to the retail customer according to the order placed by application.

2.1.3.3 Cost

The Development of **Retailo** application is estimated to be around $20,000 to $60,000 [7].

2.2 Libraries For Project

- System.Collection: Contains interfaces and classes that define various collection of objects such as lists, queues, bit arrays, hash tables and dictionaries [8].
- System.Collection.Generic: It contains interfaces and classes that define generic collections which allow users to create strongly typed collections that provide better type safety and performance than non-generic strongly typed connections [9].
- UnityEditor: UnityEditor is the GUI of the actual program [10].
- UnityEngine.UIElements: UI element is a retained-mode GUI system that opens the door to improved performance and new features such as style sheets, dynamic / contextual event handling, accessibility and much more [11].
- UnityEngine.Networking: A custom attribute that can be added to member functions of NetworkBehaviour scripts, to make them only run-on clients [12].
- UnityEngine.UI: Structure that stores the state of an animation transition on a Selectable [13].

2.3 Nuget Packages for Project

- NewtonSoft.json: For hash conversion to json to retrieve data according to the headers [14].
- Json.Net: To retrieve stream data from API [15].
- Json.User: To authenticate user credentials [16].

CHAPTER 03

AIM & STATEMENT OF PROBLEM

3 Aim And Statement of Problem

In this chapter it will be discussed what were the reasons behind opting such a project and what we aimed to advance and introduce new features to resolve any issues and upgrade the existing market using technology. The solution to the problem will also be provided and link to the industry will help to rectify any market disadvantage in the solution.

3.1 Problem Statement

The problem we are solving is that we can clearly view that most of our population has shifted online. Even those individuals who were reluctant to go online they had to switch their life according to the current conditions of the pandemic. The e-commerce System has a generalized structure since the beginning. The need to improvise and innovate this setup is the current motive for many people.

3.1.1 Inconvenience

In 21st century the trend shifts from local shops to supermarket to online shopping. As the trend of online shopping is increasing with corona virus forcing more and more customers daily to use online platforms, a new and advanced feature will be able to attract more customers and providing a unique solution to the vendors.

3.2 Solution To the Problem

Our project after completion will provide a physical experience to the user by providing them a virtual mart inside their mobiles. It will be more convenient then visiting the mart and purchasing the products and at the same time it will also provide a mart experience inside their mobile making is more attractive and fun to buy.

Moreover, to the vendors this will be a unique opportunity to attract their customers and to increase not only their sells but their brand image as well. By using this application any user will be able to purchase products online and have an amazing time doing it. Customer will be able to walk inside the virtual mart, add products to their cart and roam around the mart and eventually walk to the till to finalize the bill and pay it with relevant payment method.

This application comprises of three main modules that will help in optimizing and increasing its stability which are:

1. Virtual 3D- Modeling.
2. E-Commerce.
3. Admin panel.

3.2.1 Virtual 3D-Modeling

Virtual 3D-Modeling is the baseline for the project's virtual mart. This part will allow user to use the application in the way they shop in physical mart. It will help the user to have the similar feeling as walking and roaming around in any mart and shop at the same time by availing e-commerce solution [17].

3.2.2 E-Commerce

E-commerce is defined as the place where companies sell their product online to their customers and then delivers them to their shipping addresses. This relationship is built through internet. The customer after registration places order which is received to the admin and delivered on the provided address. Our application will be able to perform this task [18].

3.2.3 Admin Panel

Admin panel will allow the admin to control the business analytics of the application and use is to complete the received orders or to update the customer accordingly. The admin will have full access and right to receive and complete or cancel the order as per their business rules and requirements.

3.3 Aim And Objectives

Some of the main aims of this application are as follows

1. Time reduction.
2. Cost effective.
3. Virtual 3D-Mart system.
4. Complete E-commerce Application system.

3.3.1 Time Reduction

The system will reduce time as e-commerce platform eliminates a lot of time-consuming tasks such as traveling, searching, putting effort towards it, etc. The solution provides with not only feeling of the mart but is combined by the e-commerce solution that itself is time saving process.

3.3.2 Cost Effective

User will not have to go to a super mart therefore all the fuel expenses will be reduced with respect to the user. While to the vendor more customers will increase enough revenue and the expenses required to deliver products will be reduced as increase in number of orders will allow to deliver orders of same area at once.

3.3.3 Virtual 3D-Mart System

Virtual mart will attract the user to the application and give them a new and advance exercise. This can be a family entertainment with elderly family members helping them to shop and spend time with children at a same time. 3-dimenstion system will help user to visualize more and have fun while shopping [19].

3.3.4 Complete E-Commerce Application System

E-commerce system is one of the core objects of the system as this will be customer's main objective to purchase the product and get it physically while staying at anyplace as they desire. This system not only saves time and cost of the customer but also saves energy as well.

3.4 Technology Related Issues

Some issues might arise due to different reasons, and these include the basic lack of requirement of the system such as internet. If any user fails to have internet application would not be able to fulfill its job description. It is also possible that the application could be overloaded. This is mainly due to more users accessing the application at the same time then was expected and therefore the application and its servers were not prepared for [20].

CHAPTER 04

SOFTWARE ANALYSIS & REQUIREMENTS

4 Hardware, Software Analysis and Requirements

4.1 Fact Finding Techniques

Fact finding teaches allows any person to understand what is needed and is it viable in the given circumstances. The development of software also requires this fact find to understand what should be developed and is it viable in the given conditions of the developer. It also provides information about any developer who has previously attempted to do any such task and have either failed, had completed or is in the process of doing it. Their experience allows to make better decisions and to gain more information even before starting it. There are multiple techniques of fact finding and each of them are important however it is not necessary that all of these is to be used to have a good understanding but more there are used, greater the advantage [21, 22].

Some of the techniques we have used are as follows:

1. Joint Requirement Planning.
2. Prototyping.
3. Examining research papers.
4. Meetings with the domain experts.
5. Sampling documents and UML diagrams.

4.1.1 Joint Requirement Planning

Joint requirement planning is also important part of fact finding as it, in the end, bring all the developers and stack holders on one page and provide a clear idea of the project's aims and objectives and the expected results. It also brings new ideas on the table and provide better view and a more efficient way to developing the idea [23].

4.1.2 Prototyping

Prototyping helps to visualize the outcome of the project and its flow that how the project will be working and what will be the circumstance and limitations of it. It also provides a great deal of road mapping by understanding the overall outcomes and what the respective project should have in the end [24].

4.1.3 Examining Research Papers

Examining research papers is one of the most effect way to understand about the selected topic. It not only provides a great view of how to implement it but also what are the drawbacks of it and what are the basic requirement for it. It helps to understand the reason why, if others have failed to complete any task and help to rectify the cause problem and to utilize this time to improve project outlines.

4.1.4 Meetings With the Domain Experts

Meeting with the domain experts has its own great importance as the developer gains knowledge from their experience and learn to improve the details previously set by him/her. It also eradicates any misconceptions that have been in the mind of the developer and helps in identifying the limitations and strengths of the selected topic.

4.1.5 Sampling Documents and UML Diagrams

UML diagrams maps the flow of the project and the user roles in it. It helps in developing the idea into actual product. It is kind of a structure or an instruction that helps to develop step by step into one complete product [25, 26].

4.2 Software Analysis

The software requirements for our application are:

1. Visual Studio [27].
2. C# [28].
3. PhpMyAdmin [29].
4. Unity [30].

4.3 Requirements

There are two types of requirements in software development.

1. Functional Requirement [31].
2. Non-Functional Requirement [32].

4.3.1 Functional Requirements

Developed product will be able to:

1. Provide e-commerce functionalities.
2. Virtual mart.
3. Payment methods.
4. Business flow control to the admin.
5. Add/Update details of products by admin.

4.3.2 Non-Functional Requirements

Non-Functional Requirements (Reliability, Authorization, File Integrity, Audit Trail, Continuity of Processing, Service Level, Access Control, Methodology, Correctness, Ease of Use, Maintainable, Portable, Coupling, Performance, Ease of Operation)

- Reliability: The project is less likely to fail.
- Authorization: Access levels will be defined according to the user’s designation.
- File Integrity: No file integrity.
- Audit Trail: Documentation of any failure incurred over a period or in the delivery of the product.
- Continuity of Processing: Appropriate flow of processes.
- Service Level: The number of users using the application.
- Access Control: The access levels for the users are defined.
- Methodology: E-commerce methodology.
- Correctness: Software should follow the procedure accordingly.
- Ease of Use: User friendly UI.
- Maintainable: Updates when needed.
- Portable: Applications are portable.
- Performance: Very responsive.
- Ease of Operation: Very responsive.

CHAPTER 05

SOFTWARE DESIGN & MODELING

5 Software Design and Modeling

5.1 Process Flow Diagram

Process Flow Diagram (PFD) is basically graphically representation of a process in a system. In Figure 5.1, it shows the overall process of the application, this application needs an account than get access to view the super mart. After the application has been start the user has to view the products in different shelves and move the trolley in all direction. Users pick the product by clicking functionality and adding the product into the cart, then go towards counter for checkout and confirm their order [33,34].

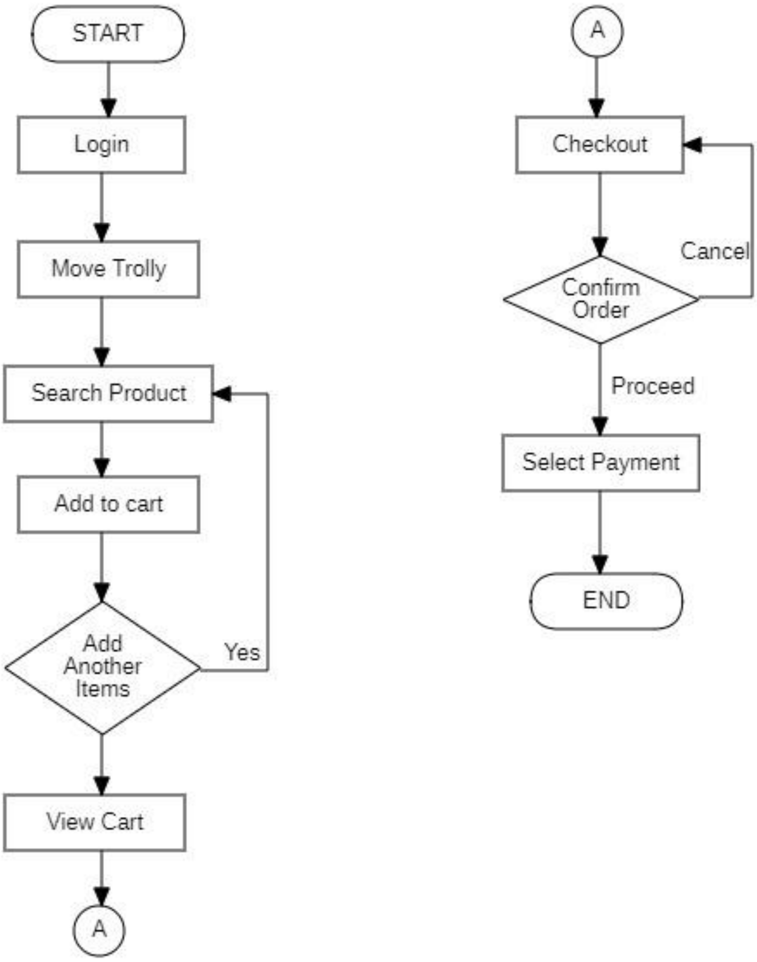


Figure 5.1: Process Flow Diagram

Figure 5.1 shows that the customer will first login in the application for the verification process. If the customer is using the application for the first time, then it is a necessary requirement for the customer to register into the application by creating an account. After the account being successfully created the customer can now login into the application provided correct password is entered.

After successful login attempt from the customer, the supermarket scenery will be displayed to the customer. The customer will be able to see the trolley on the screen. The customer can move the trolley towards and shelf the desire. When upon reaching their desired product shelf they can add the product in to the cart. After putting or placing the product into the trolley, they can also view the products available in trolley to make sure that all the desired products are present in the trolley.

Once it is confirmed that all the required products are added the customer can then proceed towards the checkout counter to confirm their product. After clicking on the confirm product the payment method will be asked. Customer can select the payment method from the options provided according to their convenience.

5.2 UML Diagrams

5.2.1 Use Case Diagram

Use Case Diagram of project defines the actor role, in these 2 actors (user and admin), admin can move, update, view, remove or change the product from the super mart while user can perform only search, view and add to cart in the mart and perform billing process by payment method [35,36].

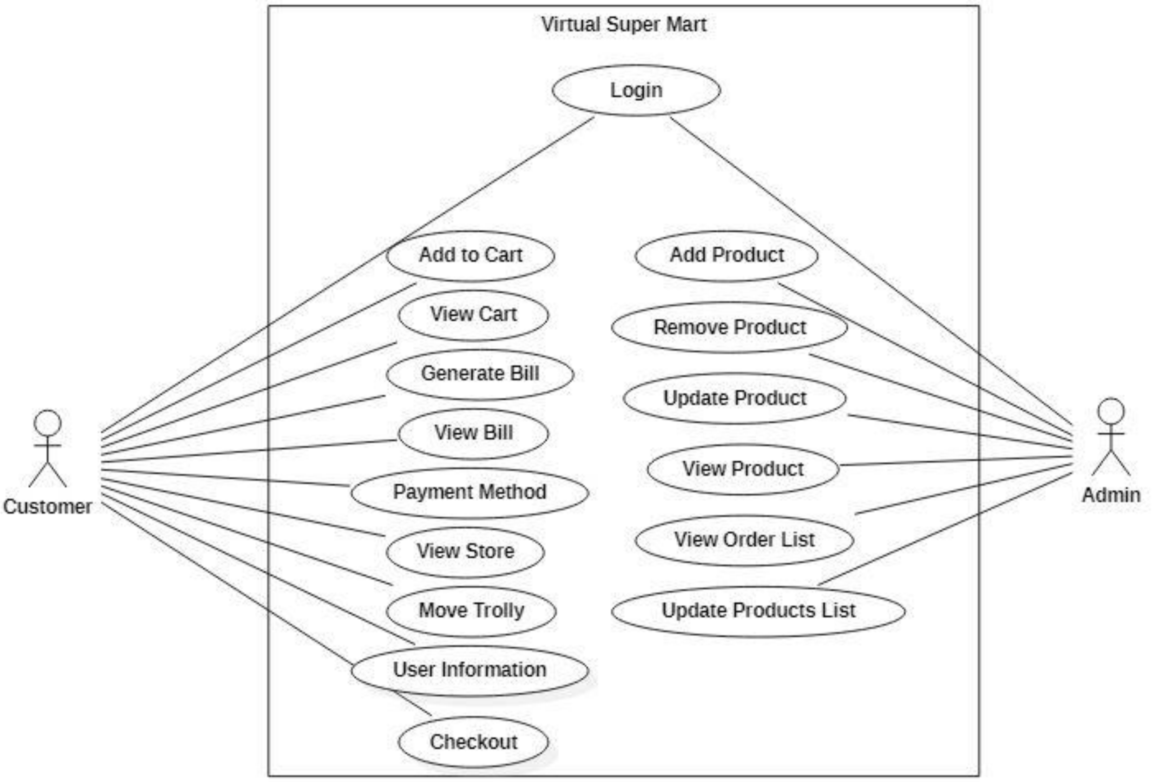


Figure 5.2: Use Case Diagram

Figure 5.2 shows that, those actions that can be performed both customer and the admin respectively. Actions that can be performed by the customer are that the customer can add an item to the cart, cart can be viewed, can generate bill, able to view the bill, can select the payment method according to the customer's convenience, can also view or visit the entire store, can move the trolley towards the desired shelf or location and the customer can also login for authentication purpose. The actions that can be performed by the admin are that the admin can login for authentication, can add any product in the inventory, or remove any product from the inventory, admin can view all the products present in the inventory, admin can view the order list and admin can also update the product list.

5.2.2 Class Diagram

Class Diagram provides a detailed overview of the system through the relationships between the objects and classes inside the system. It gives the structure of a system in terms of classes, attributes, and methods. In Figure. 5.3, classes are represented with boxes which contain three parts: upper part contain the name of class i.e., user, admin, customer, trolley, order product, product, invoice, delivery and category, central part contains attributes and lower part contains methods [37,38].

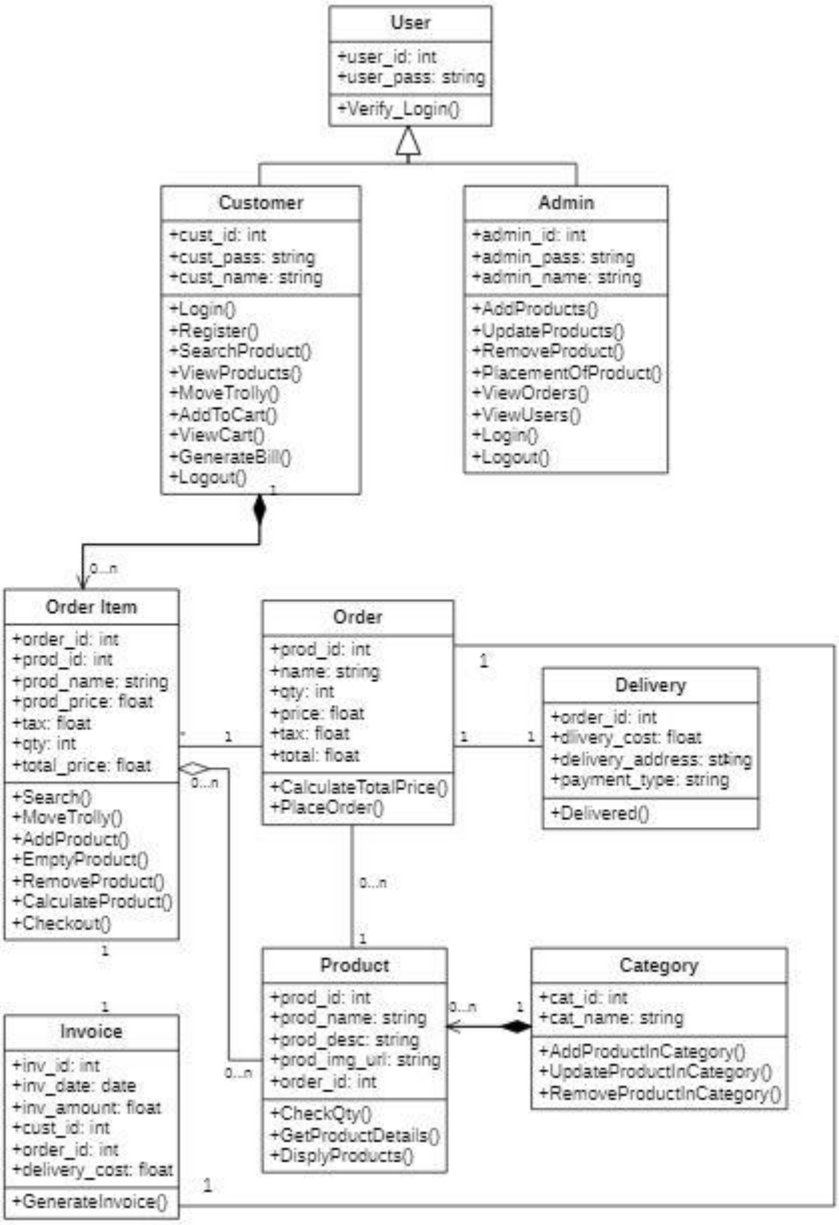


Figure 5.3: Class Diagram

Figure 5.3 shows the class diagram of our project. In this diagram there are total nine classes. Each class has some attributes and function. In the class of user, the attributes are user ID, user password and function of verify login. Both the customer class and the admin class are inherited from the user class. The attributes of the customer class are customer ID, customer password, customer name and the functions of this class are login, register, search product, view product, move trolley, add to cart, view cart.

The attributes of the admin class are admin ID, admin password, admin name and the functions of this class are to add product, update product, remove product, placement of the products, view orders, view users. The attributes of the trolley class are order ID, product ID, product name, product price, tax, quantity, total price and the functions of this class are search, move trolley, add product, empty product, remove product, calculate product, checkout.

The attributes of the invoice class are invoice ID, invoice date, invoice amount, customer ID, order ID, delivery cost and the function of this class is to generate invoice. The attributes of the order product class are product ID, name, quantity, price, tax, total and the functions of this class are calculating total price and placing order. The attributes of the product class are product ID, product name, product description, product image URL, order ID and the functions are check quantity, get product details, display products. The attributes of the class delivery are order ID, delivery cost, delivery address, payment type and the function of this class is delivery.

The attributes of the class category are category ID, category name and the function of this class is to add product in category, update product in category. Both the customer class and the admin class are inherited from the parent class user. The relationship between the customer class and the trolley class is composition because the class trolley lives and dies with the customer class.

The relationship between the trolley class and the order product class is association as both classes are associated with each other which is one to zero or one. The relationship between the order product class and the delivery class is association which is one-to-one relationship. The relationship between the trolley class and the invoice class is association which is a one-to-one relationship.

The relationship between the trolley class and the product class is aggregation because product class is a part of trolley, and both the classes have different timelines. The relationship between the order product class and the product class is association which is a one or many to one relationship. The relationship between the product class and the category class is composition as the product class lives and dies with the category class.

5.3 Interaction Diagram

5.3.1 Sequence Diagram

Sequence Diagram describes interactions among classes in terms of an exchange of messages over time. Sequence Diagram contains object, actions, lifelines, messages, and option loops. In Figure 5.4, firstly customer signup or sign in then application starts, and super mart shows in the screen. Now customer will do shopping in virtual environment [39].

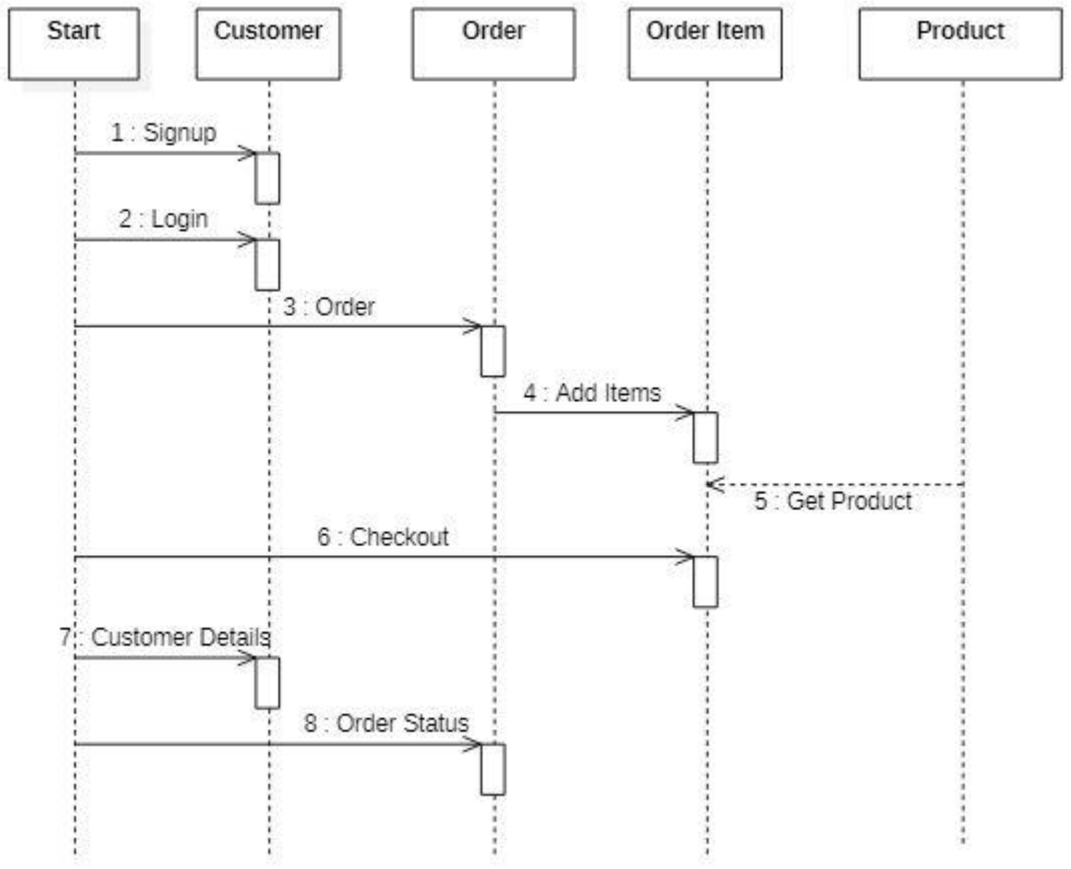


Figure 5.4: Sequence Diagram

Figure 5.4 shows that the sequence diagram of our project. If the customer is not registered, then it will be first directed towards signup. Upon successful account creation or account already registered both can login into the application. If there are several login attempts fail the user will go through the verification process. The customer will undergo to the verification process if required. The customer afterwards can select the product upon his or her desire. The customer can remove or add from the basket if the product has been added mistakenly or was not the desired item. Then customer can request to receive the cart details. Customer will then review the cart details. Now if all the desired items are present in the trolley, then customer can move towards the checkout counter. Then the delivery staff will prepare the order according to the items listed against the order ID and then will be handed over to the delivery man.

5.3.2 Communication Diagram

Communication Diagram used to model the dynamic behavior of the use case. Figure 5.5, it shows the communication diagram of the application [40].

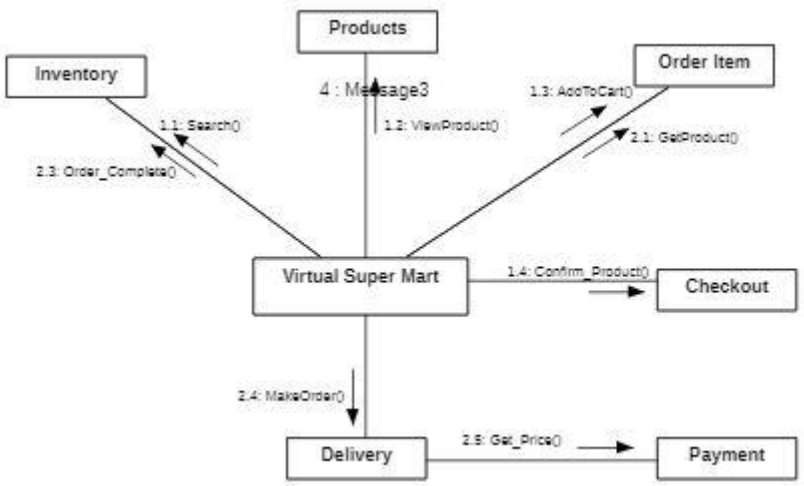


Figure 5.5: Communication Diagram

Figure 5.5 shows that the communication diagram of our project. In this diagram user is our customer. The customer object will send the message of go to mart to the virtual super mart object then from virtual super mart object message of search will be sent to the inventory. Another message of view products will be sent to the products through virtual super mart. Message of add to cart will be sent to the trolley from virtual super mart. The message of the confirm product will be sent to the checkout from virtual super mart. Now the customer will proceed for the checkout by sending the checkout message to the virtual super mart. The message of get the product will be sent to the trolley. The message of the order complete will be sent to the inventory. The message of the make order will be sent to the delivery. The message of the get price will be sent to the payment from the delivery.

5.4 Control Flow Diagram

5.4.1 Activity Diagram

Activity Diagram are graphical representation of workflows of stepwise activities and actions with support for choice, iteration, and concurrency [41].

5.4.1.1 Admin Activity Diagram

In this admin side the process of project is that firstly admin registered or login then it will show the admin panel dashboard, then it will view, add, remove, update the products and orders.

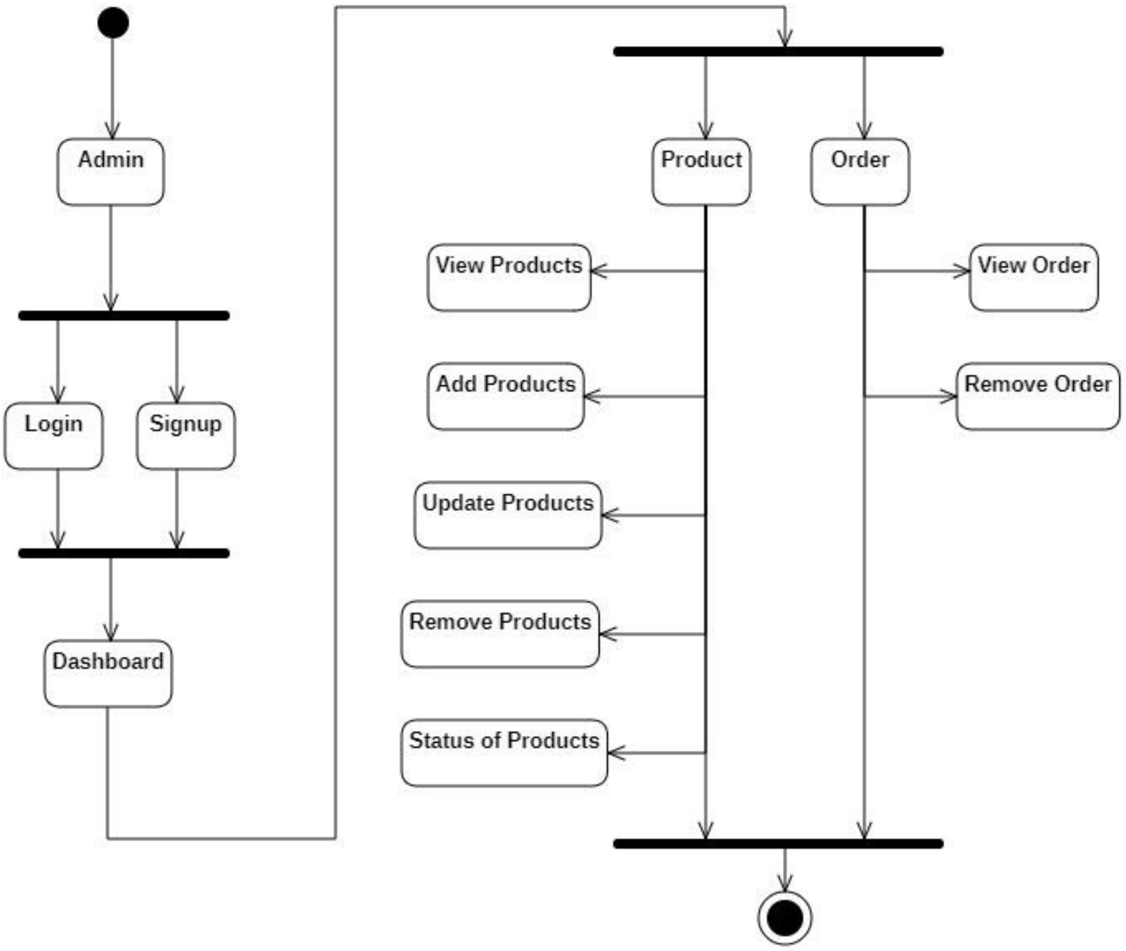


Figure 5.6: Activity Diagram based on Admin.

Figure 5.6 shows that the activity diagram based on admin. According to this diagram the admin can login or signup. If the admin is already registered, then it will directly login in to the application. If the admin is not registered, then the admin will sign up for an account. After successful login attempt the admin is now able to view the dashboard on the screen. On the dashboard screen the admin can work on both the product and the order according to his or her desire. For the product admin can view all the products, can add new products, update products, remove products and can also check, verify or update the status of the product. Admin can also work on orders like admin can view the order and can also remove the order.

5.4.1.2 User Activity Diagram

In this user side the process of project is that firstly user registered or login then it will show the whole super mart in virtual, it will select item by moving trolley and add it into cart, then go to counter and generate bill after select payment method and confirm the order.

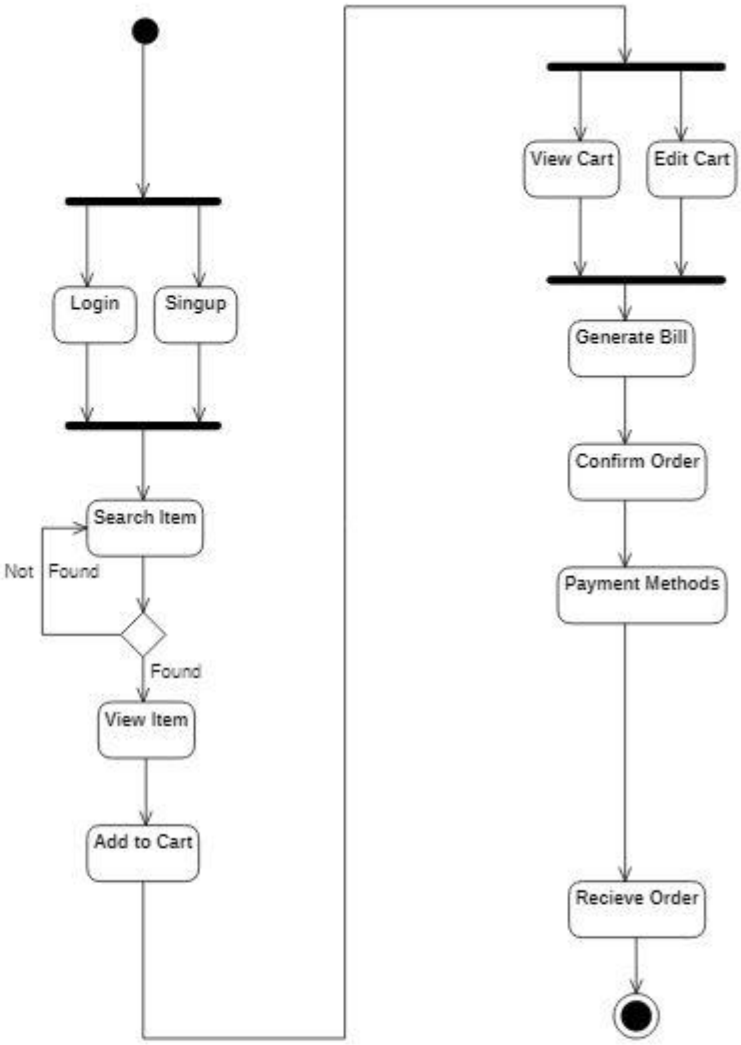
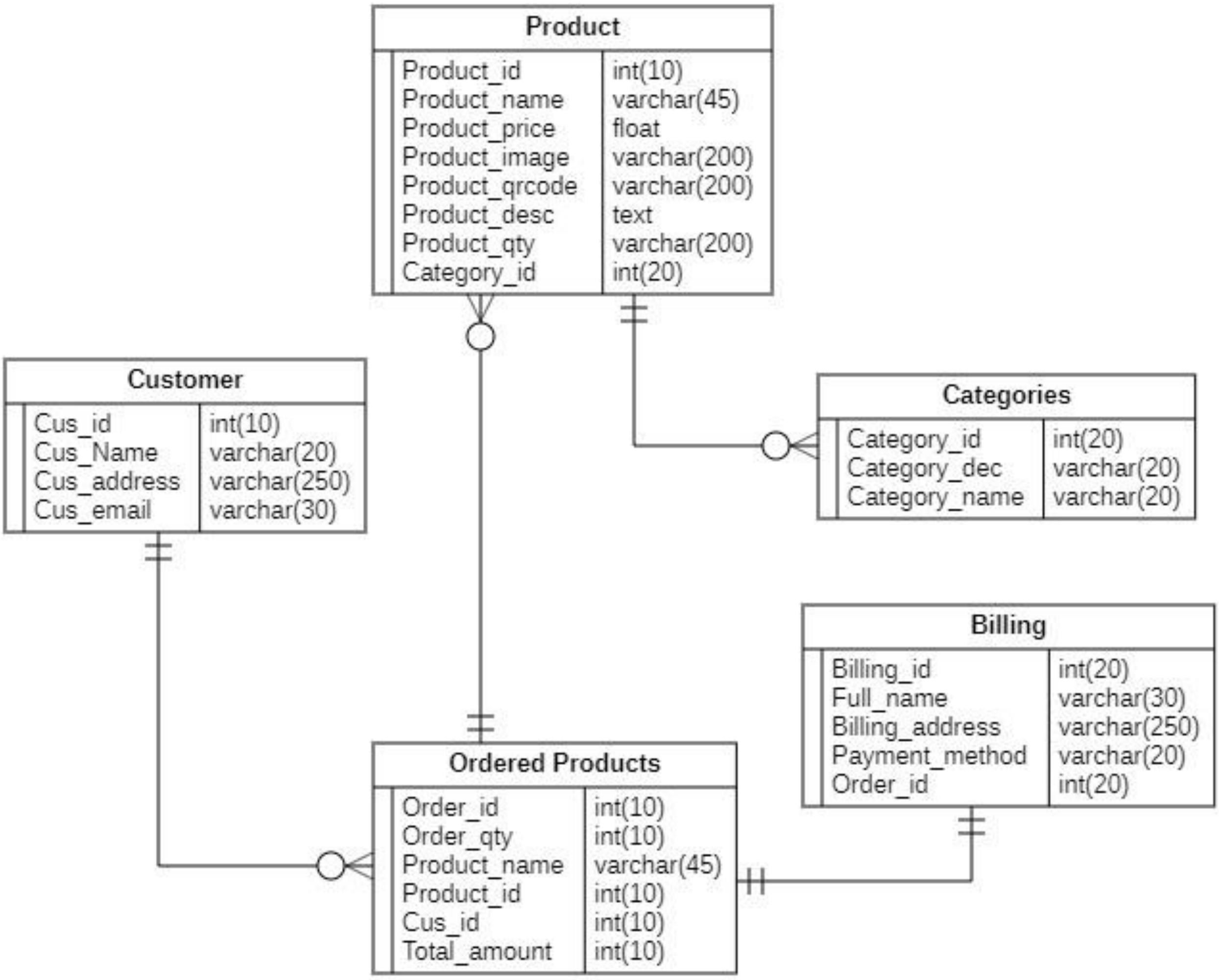


Figure 5.7: Activity Diagram based on Customer.

Figure 5.7 shows that the activity diagram of customer. The customer will first create an account if the customer does not have a registered account. Upon successful creation of account, the customer can now login into the application. After successful login attempt customer can search for an item. If the item is found, then the customer will proceed forwards and add the product in to the trolley. If the item is not found, then customer will look for the desired product again. The customer can view the cart to check whether they have all the desired products in their trolley or not. Then the customer will click on the generate bill. The review of the bill customer can now confirm his or her order.

5.5 ER Diagram

Entity-Relationship diagram is one of the most used structural analysis and an easy to understand and powerful to model real-world problems and translate it into database schema. ERD contains business entities, attributes, and the relationships among them. Where an entity is an object type, relationship is the association of entities and attribute is a property that describes the relationship or the entity [42].



Product	
Product_id	int(10)
Product_name	varchar(45)
Product_price	float
Product_image	varchar(200)
Product_qrcode	varchar(200)
Product_desc	text
Product_qty	varchar(200)
Category_id	int(20)

Customer	
Cus_id	int(10)
Cus_Name	varchar(20)
Cus_address	varchar(250)
Cus_email	varchar(30)

Categories	
Category_id	int(20)
Category_dec	varchar(20)
Category_name	varchar(20)

Billing	
Billing_id	int(20)
Full_name	varchar(30)
Billing_address	varchar(250)
Payment_method	varchar(20)
Order_id	int(20)

Ordered Products	
Order_id	int(10)
Order_qty	int(10)
Product_name	varchar(45)
Product_id	int(10)
Cus_id	int(10)
Total_amount	int(10)

Figure 5.8: ER Diagram

Figure 5.9 shows that the ER diagram of our project. In this diagram there are five entities namely product, customer, categories, ordered product, billing. The attributes of the product entity are product ID, product name, product price, product image, product QR code, product description, product quantity, category ID. The attributes of the customer entity are customer ID, customer name, customer address, customer email.

The attributes of the ordered products are order ID, order quantity, product name, product ID, customer ID, total amount. The attributes of the category's entity are category ID, category description, category name. The attributes of the billing entity are billing ID, full name, billing address, payment method, order ID. The relationship between the customer entity and the order product entity is one and only one and zero or many relationships respectively. The relationship between the product entity and the ordered product entity is zero or many and one and only one respectively. The relationship between the product entity and the categories entity is one and only one and zero or many respectively. The relationship between the ordered product entity and the billing entity is one and only one for both the entities.

CHAPTER 06

ALGORITHM ANALYSIS & CODE COMPLEXITY

6 Algorithm Analysis and Code Complexity

6.1 Algorithm

Set of rules to obtain the expected output from the given input. It is the general structure of a class or any problem to achieve the desired output. It consists of specific instructions and calculates the results based on these instructions regardless the changes in input, the output is always generated accordingly and help achieve the correct results [43].

6.1.1 Pseudo Code (Customer)

Pseudocode is a step-by-step written outline of your code that you can gradually transcribe into the programming language. Is consist of English phrases to explain the required tasks in any given project of program. Ideally Pseudocode does not include the keywords on any programming languages.[44].

```
1. If user is not registered.
        GOTO Registration Process.
2. If the user is already registered.
        GOTO Login Page.
3. Process, send them to MySQL database to verify.
4. Show Mart
5. If the user wants to move trolley.
        Click “Up, Down, Right & Left” button
6. If the user clicks on the product.
        Appears short description about that product.
7. If the user wants to add the product to cart.
        Add Product by Clicking.
8. Else if the user does not want the product.
        GOTO next product by moving trolley.
9. If the user wants to view cart.
         Click “View Cart” button.
```

```
10. If the user wants to check out.
        GOTO counter
11. Else keep purchasing.
12. If user clicks on checkout.
        Appears Bill slip
13. Show payment method
14. Fill out the field of delivery schedule.
15. Process the fields, if all the fields are correctly entered.
        GOTO Confirm Order
16. Else Show error message.
17. Exit.
```

6.2 Search Algorithm

Table 6.1: Search Algorithm

S. No.	Algorithm	Cost	Time
1	Initialize a, n, x:0 Set lowerBound = 1 Set upperBound = n	 C_1 C_2	 1 1
2	while(x)! -> not found a) if upperBound < lowerBound i) Exit: x does not exist	C_3 C_4 C_5	n+1 n-k n-k
3	Set midpoint = lowerBound + (upperBound - lowerBound)/2	C_6	n
4	if A[midPoint] < x a) set lowerBound = midpoint + 1	C_7 C_8	n n-k
5	if A[midPoint] > x a) set lowerBound = midpoint - 1	C_9 C_{10}	n n-k
6	if A[midPoint] < x a) Exit: x found at location midpoint End while End procedure	C_{11} C_{12}	n n-k

k = no. of conditions when IF condition is true.

Table 6.2: Complexity Comparison

Model	Number of comparisons (for n=1000)	Comparison as a function
Best Case (Least comparison)	1 (middle item is the target)	O (1)
Worst Case (Most comparison)	16 (target does not present in array)	O (log n)

CHAPTER 07

IMPLEMENTATION

7 Implementation

7.1 Component Diagram

The component diagram aims to identify and create the structural relationships between the components of a given system. The advantage of this is to easily reuse or substitute different component implementations as these contains the behavior and specified interfaces. It creates the architecture to begin modeling of the solution [45].

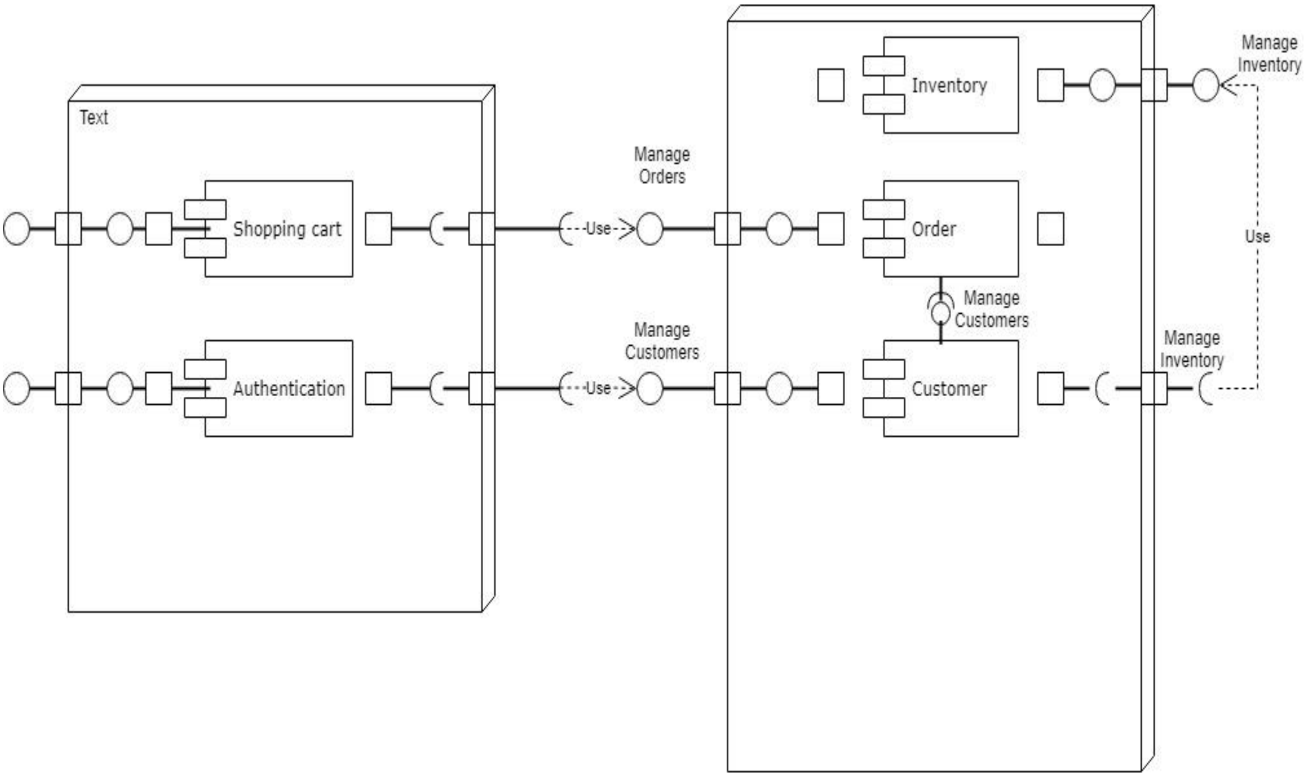


Figure 7.1: Component Diagram

Figure 7.1 shows that the component diagram of our project. The component shopping cart requires data using the required interface from the order component in order to manage the orders and the order component can provide the data by using the provided interface type. The component shopping cart component is dependent on the order component. The order component requires data to manage the customers by using the required interface and the customer component will provide the data to the order component using the provided interface type.

The authentication component requires data by using the required interface to manage the customers and the customer component will provide the data to the authentication component using the provided interface type. Authentication component is dependent on the customer component. The customer component will require data from the inventory component using the required interface to manage the inventory and the inventory component will provide the data to the customer component by using the provided interface type.

7.2 Deployment Diagram

Deployment diagrams aim to provide the physical arrangement of the elements of the system and links to the database server and their connections to achieve the desired deployment of the system. It shows one of the possible arrangements that can be implemented in each program of project [46].

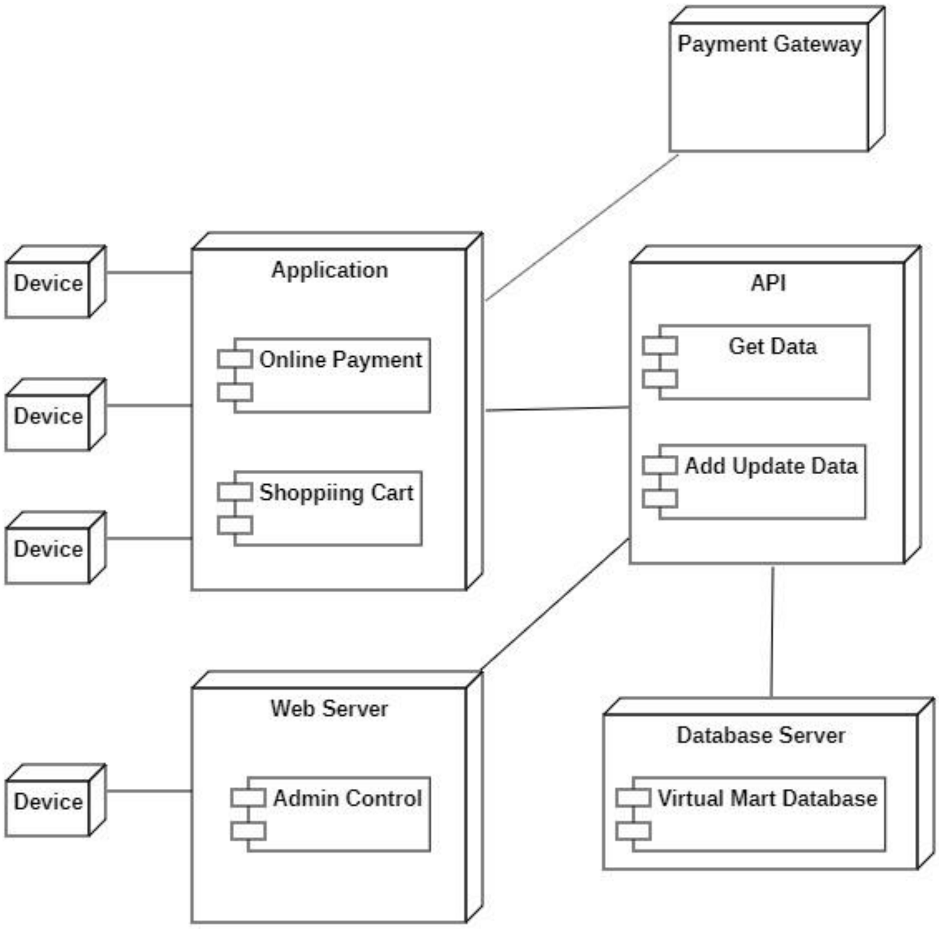


Figure 7.2: Deployment Diagram

Figure 7.2 shows that the deployment diagram of our project. In the application node there are two artifacts online payment and shopping cart. The API gateway node has two artifacts get data and add update data. The web server has one artifact of admin control. The Database server has one artifact of virtual mart database. The application node has association link with the payment gateway in order to manage the payment process. The application data has an associated link with the API in order to manage the shopping cart artifact so that products are added into the trolley. Web server has an associated link with the API so that admin can manage both the products and the orders. The API node is acting as an intermediary layer to facilitate the admin in interacting with the database of the virtual mart. API has an association connection with the database server so that an query from the admin can easily be entertained.

7.3 State Diagram

It also called state machine diagram, which shows discrete behavior of a part of designed system through finite state transitions containing all the states that an object can have, event in which object state is changed, the conditions that must be fulfilled before the any transaction and activities during the life of object [47,48].

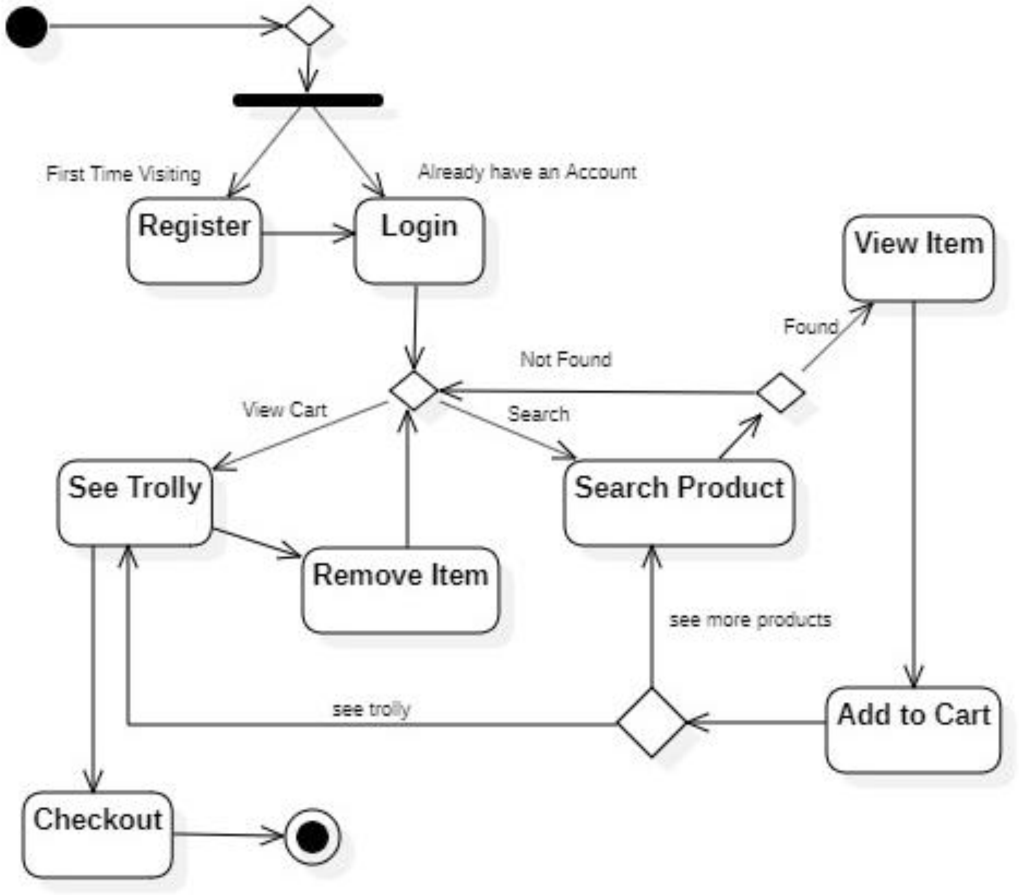


Figure 7.3: State Diagram

Figure 7.3 shows that at first customer will register for an account if he or she is visiting for the first time. If this is not the first time, then the customer can login directly. After sign in customer can view cart, or search for product after search customer can add the product in to the trolley and if the product is not desired it can be removed by the customer easily. After adding all the desired products customer can proceed towards the checkout counter for the payment.

7.4 GUI

7.4.1 Customer Side

This is the login screen of our application Virtual Super Mart. On this page customer can create an account if they are experiencing the application for the first time. If account of the customer is already registered the customer will be able to login directly.

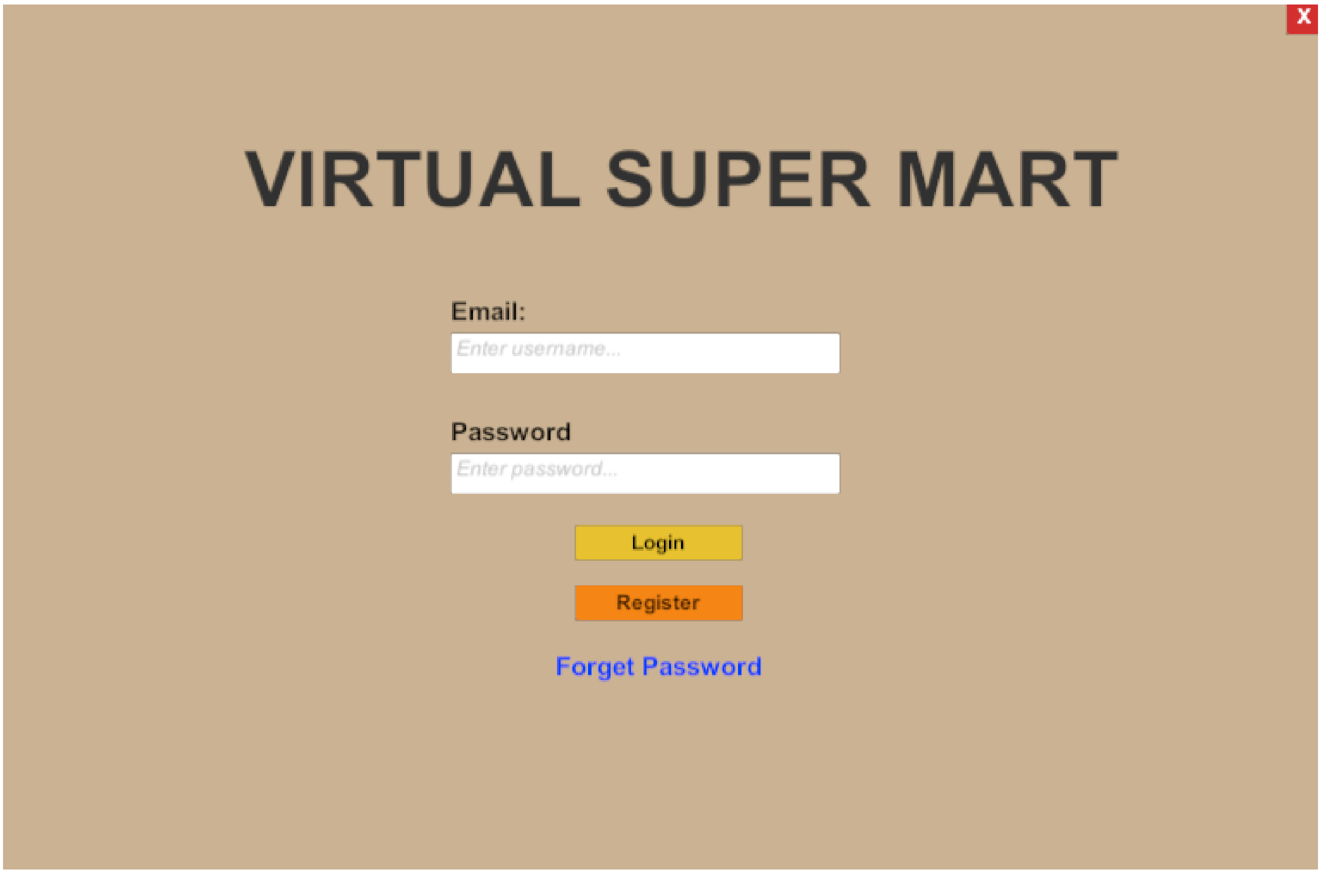


Figure 7.4: Login Interface

After the successful login attempt by the customer. The shopping center scenery will be visible on the customer's screen. On this screen customer can start doing shopping by clicking on the image of the product to add in the shopping trolley cart. Customer can view the cart from button view cart on the top right of the screen to view the product list.

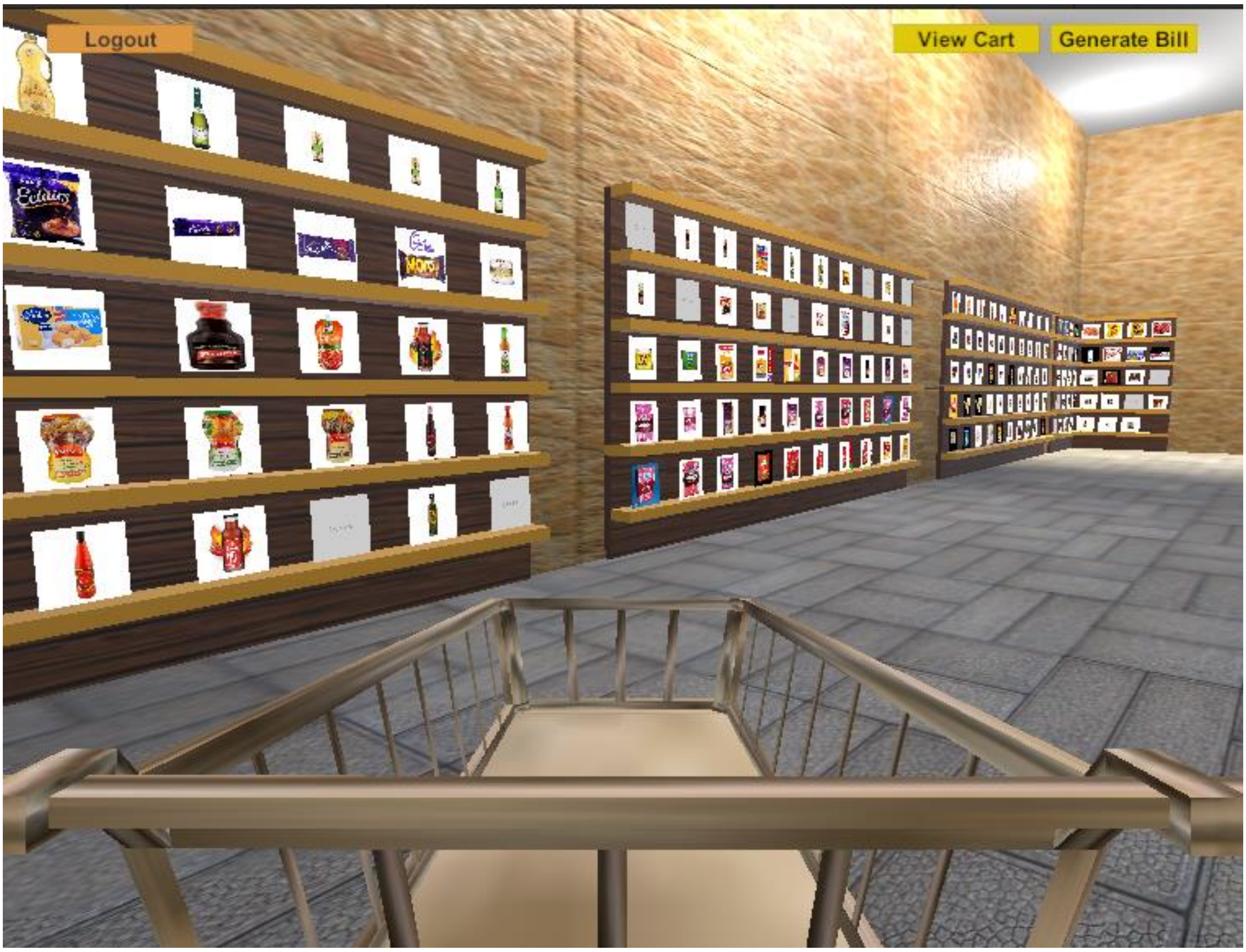
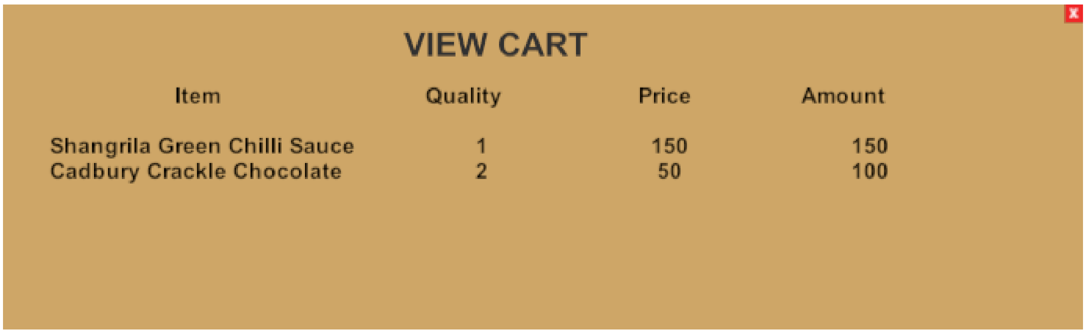


Figure 7.5: Virtual Mart Interface

This image shows the list of the products added into the cart along with the quantity of the product, weight, and the price of the product in single line. I can be used by the customers to tally their cart and acknowledge that they have successfully added their desired products into the cart.



VIEW CART

Item	Quality	Price	Amount
Shangrila Green Chilli Sauce	1	150	150
Cadbury Crackle Chocolate	2	50	100

Figure 7.6: View Cart Interface

The customer will then proceed to the counter for checkout. So that the order can be confirmed by the customer. The order confirmation is only possible at the counter as in real world this is the process to generate the bill and leave the mart. This order will then be received at the admin panel.

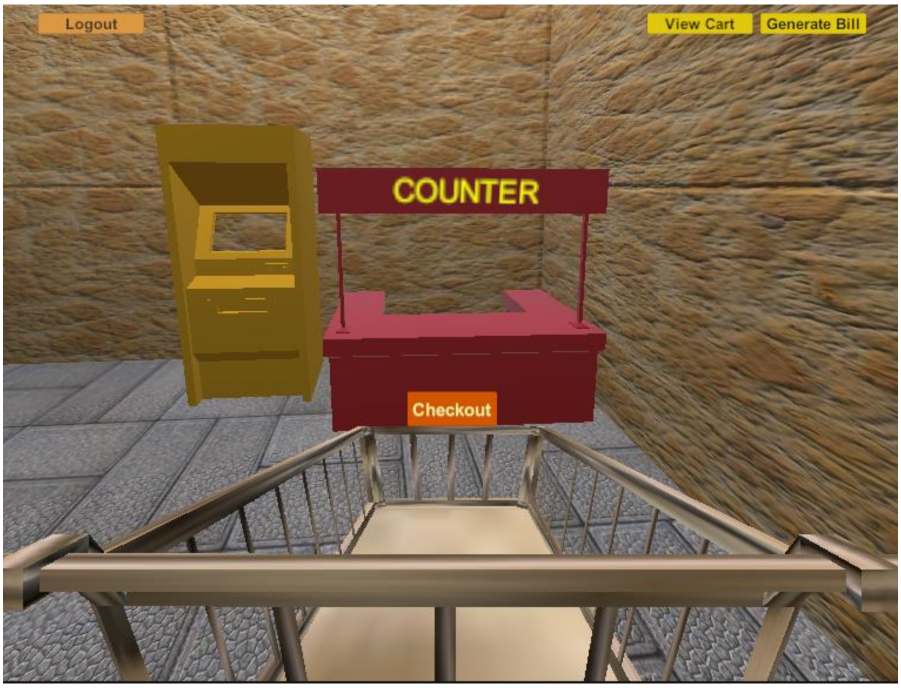
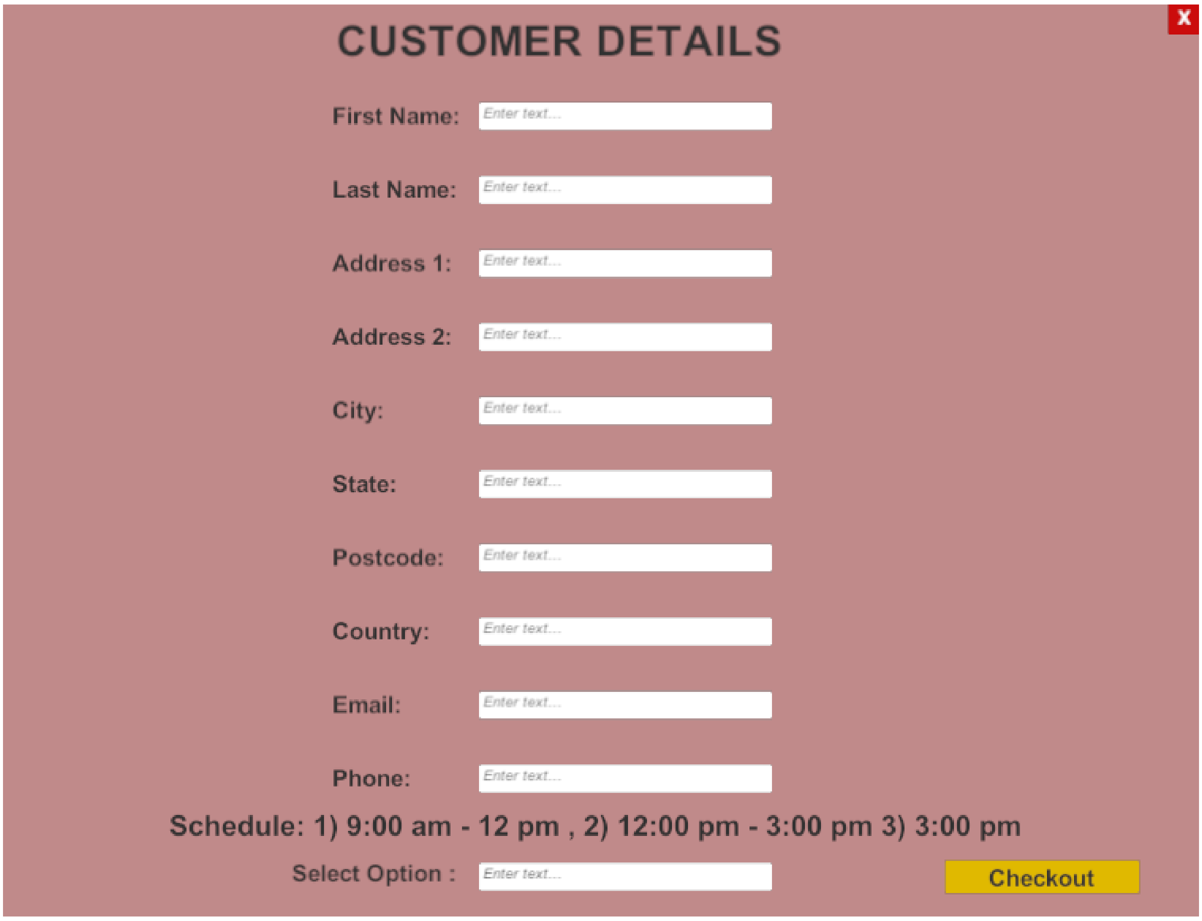


Figure 7.7: Counter Interface

After verifying the products from the view cart list. The customer will then click on the generate bill button next to the view cart button. After clicking the generate button the screen with information form will be appeared on the screen. The customer will then fill out the required information and will at the end also specify the schedule.



CUSTOMER DETAILS

First Name: Enter text...

Last Name: Enter text...

Address 1: Enter text...

Address 2: Enter text...

City: Enter text...

State: Enter text...

Postcode: Enter text...

Country: Enter text...

Email: Enter text...

Phone: Enter text...

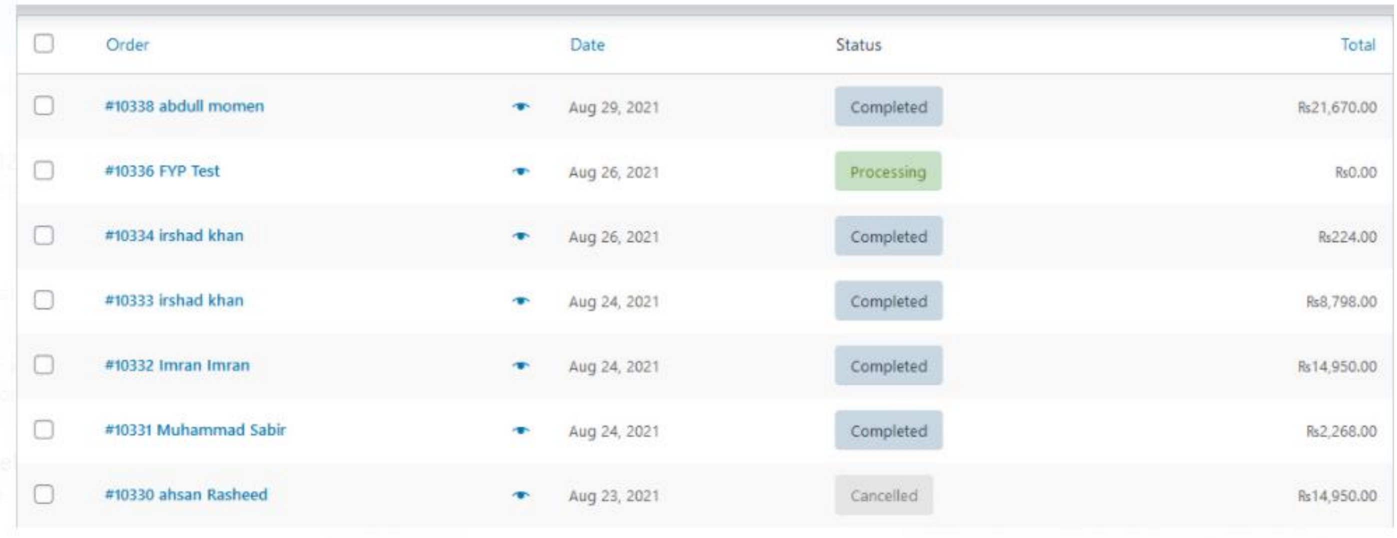
Schedule: 1) 9:00 am - 12 pm , 2) 12:00 pm - 3:00 pm 3) 3:00 pm

Select Option : Enter text...

Figure 7.8: Customer Details Interface

7.4.2 Admin Side

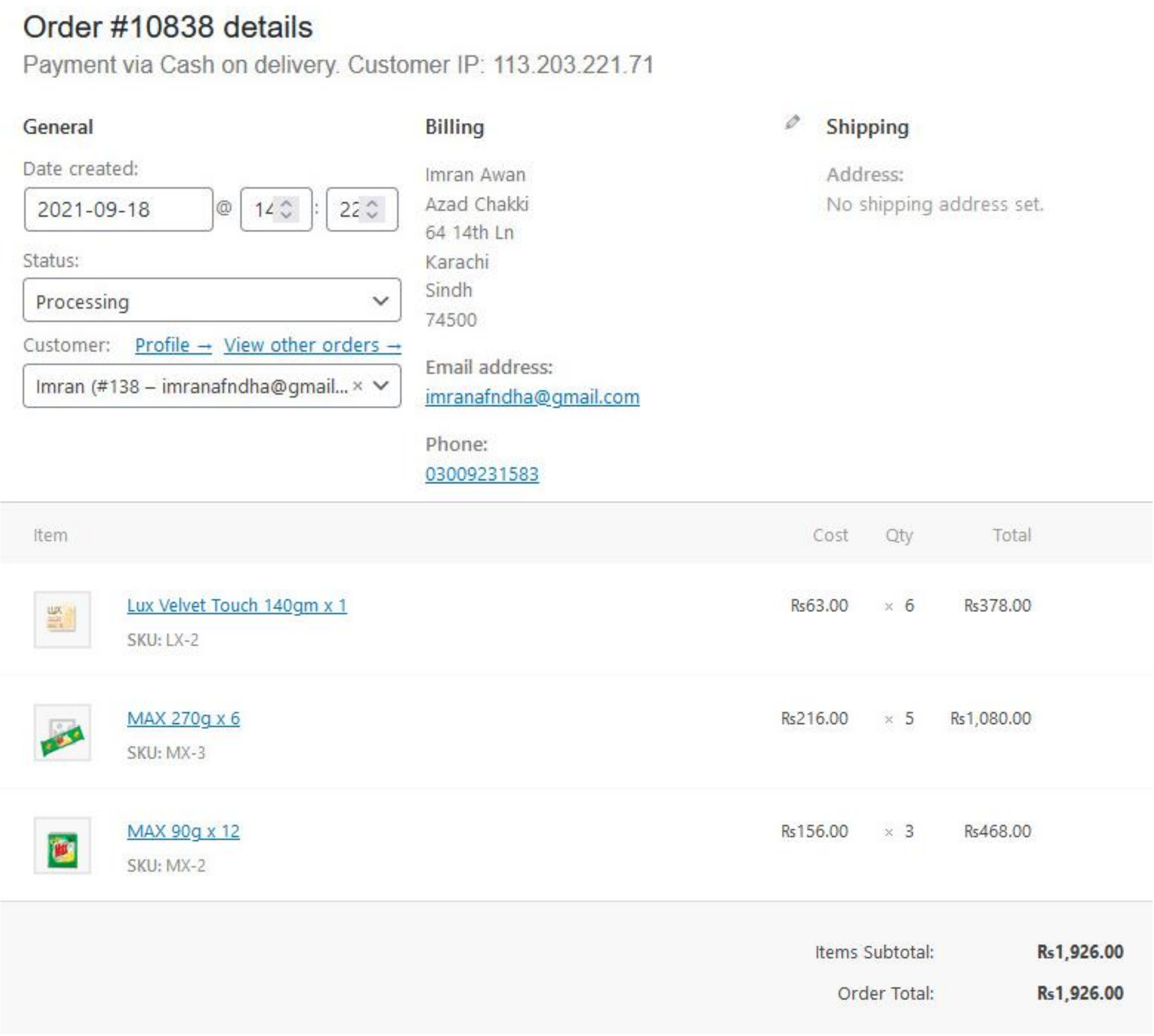
The admin has a list of orders received from the customers. These orders have the date or order placed and the status of the order. This status is updateable by the admin as per the situation admin can cancel or complete the order status. This list brings all orders in one window to facilitate the Admin.



□	Order		Date	Status	Total
□	#10338 abdull momen	👁	Aug 29, 2021	Completed	₨21,670.00
□	#10336 FYP Test	👁	Aug 26, 2021	Processing	₨0.00
□	#10334 irshad khan	👁	Aug 26, 2021	Completed	₨224.00
□	#10333 irshad khan	👁	Aug 24, 2021	Completed	₨8,798.00
□	#10332 Imran Imran	👁	Aug 24, 2021	Completed	₨14,950.00
□	#10331 Muhammad Sabir	👁	Aug 24, 2021	Completed	₨2,268.00
□	#10330 ahsan Rasheed	👁	Aug 23, 2021	Cancelled	₨14,950.00

Figure 7.9: Order List Interface

The details of the order are available to the admin to view and complete the order placed. This is one of the necessary panels for admin in order to know what the demand of the customer is and to fulfill it accordingly. Both the customer details and order details are available in one window to the admin.



Order #10838 details

Payment via Cash on delivery. Customer IP: 113.203.221.71

General

Date created: 2021-09-18 @ 14 : 22

Status: Processing

Customer: Profile → View other orders →

Imran (#138 – imranafndha@gmail... ×

Billing

Imran Awan
Azad Chakki
64 14th Ln
Karachi
Sindh
74500

Email address:
imranafndha@gmail.com

Phone:
03009231583

Shipping

Address:
No shipping address set.

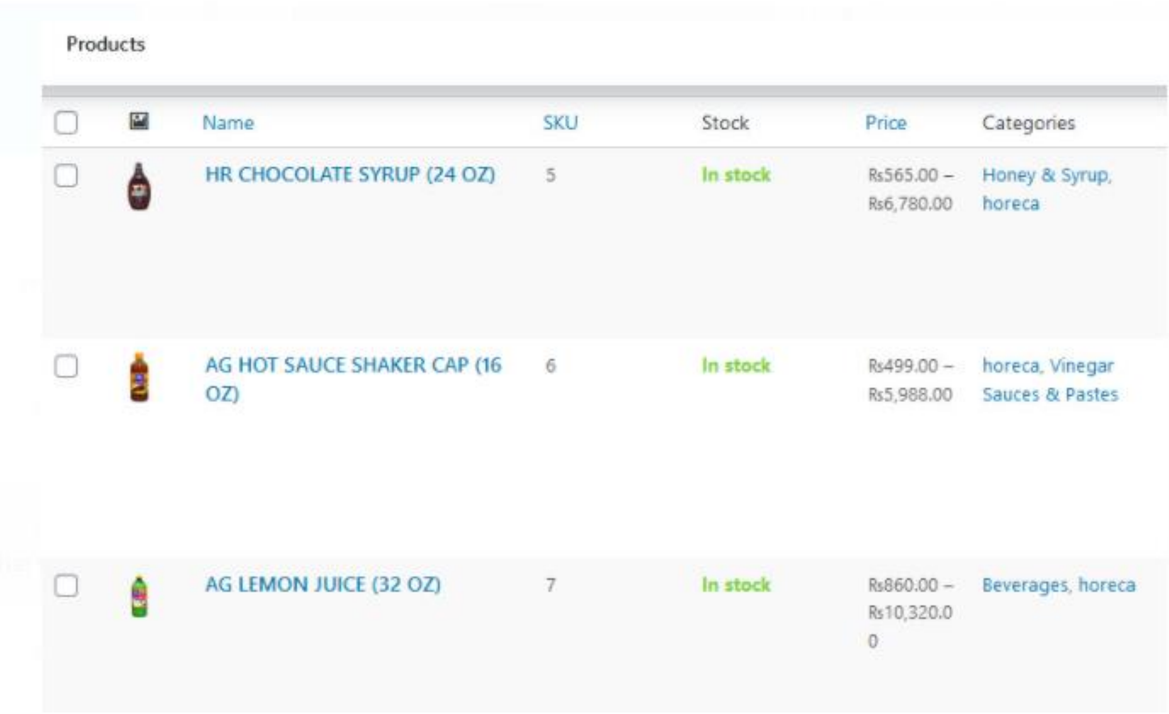
Item	Cost	Qty	Total
Lux Velvet Touch 140gm x 1 SKU: LX-2	Rs63.00	× 6	Rs378.00
MAX 270g x 6 SKU: MX-3	Rs216.00	× 5	Rs1,080.00
MAX 90g x 12 SKU: MX-2	Rs156.00	× 3	Rs468.00

Items Subtotal: Rs1,926.00

Order Total: Rs1,926.00

Figure 7.10: Order Detail Interface

The product list is available to the Admin to view and update the product details including stock quantity, price, and status such as Out of stock or In Stock. This list helps the Admin to easily sort their priorities and change any detail to ensure best possible service to the customer.

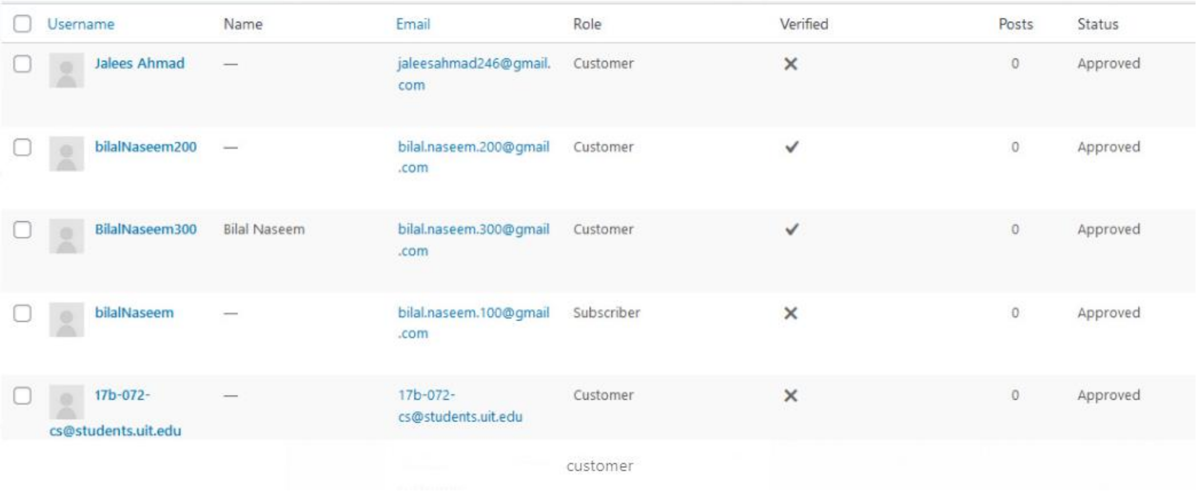


Products

□		Name	SKU	Stock	Price	Categories
□		HR CHOCOLATE SYRUP (24 OZ)	5	In stock	Rs565.00 – Rs6,780.00	Honey & Syrup, horeca
□		AG HOT SAUCE SHAKER CAP (16 OZ)	6	In stock	Rs499.00 – Rs5,988.00	horeca, Vinegar Sauces & Pastes
□		AG LEMON JUICE (32 OZ)	7	In stock	Rs860.00 – Rs10,320.00	Beverages, horeca

Figure 7.11: Product List Interface

This figure depicts the list of the customers at our virtual supermarket. For each customer entry the detail of customer email, their role, their status of verification, the status of the membership whether it is approved or not.



□	Username	Name	Email	Role	Verified	Posts	Status
□	Jalees Ahmad	—	jaleesahmad246@gmail.com	Customer	✕	0	Approved
□	bilalNaseem200	—	bilal.naseem.200@gmail.com	Customer	✓	0	Approved
□	BilalNaseem300	Bilal Naseem	bilal.naseem.300@gmail.com	Customer	✓	0	Approved
□	bilalNaseem	—	bilal.naseem.100@gmail.com	Subscriber	✕	0	Approved
□	17b-072-cs@students.uit.edu	—	17b-072-cs@students.uit.edu	Customer	✕	0	Approved
				customer			

Figure 7.12: Customer List Interface

7.5 Code

7.5.1 Api Class Code

The purpose of this class is to call the data from the server and deploy in virtual mart. At the start of the project this class is called and inside the method start (), which is the first method of any class that is called, the API URL, it's key and secret are defined and saved in a variable of RestApi class object. Later using this same object, the products along with their details are called from server and saved in another variable. Before placing the products, the available place is calculated and then the products are displayed according to available space after being sorted.

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;
using System.Net;
using System.IO;
using System;
using WooCommerceNET;
using WooCommerceNET.WooCommerce.v2;
using UnityEditor;
using UnityEngine.UIElements;
using UnityEngine.Networking;
using UnityEngine.UI;
using System.Linq;

public class API : MonoBehaviour
{
    public static WCObject _wcObject;

    List<Product> listOfProductsFromDatabase;
    [SerializeField] GameObject[] productsOnShelves;
    [SerializeField] Products products;

    private static string website_url = "https://tradeways.pk";
    private static string consumer_key =
"ck_b14c4dc3515782bf3a98286052899e159428430d";
    private static string consumer_secret =
"cs_71fd545f759f542e1292b4df05fd200154bd6f5c";
    void Start()
    {
        productsOnShelves = products.GetListOfProductsOnShelves();
        try
        {
            string url = "https://tradeways.pk/wp-json/wc/v2/";
            string key = "ck_b14c4dc3515782bf3a98286052899e159428430d";
            string secretKey = "cs_71fd545f759f542e1292b4df05fd200154bd6f5c";
            var restAPI = new RestAPI(url, key, secretKey, false);
            _wcObject = new WCObject(restAPI);
            listOfProductsFromDatabase = GetAllProducts();
            PlaceProductsOnShelves();
        }
        catch (Exception e)
        {
            SendMessage(e.Message);
        }
    }
    private void PlaceProductsOnShelves()
    {
        for (int i = 0; i < productsOnShelves.Length; i++)
        {
            var productInfo = listOfProductsFromDatabase[i];
            String imgUrl = productInfo.images[0].src;
            var a = productInfo.sku;
            productsOnShelves[i].GetComponent<ProductItem>().SetProductInfo(productInfo);
            StartCoroutine(ApplyImageToProduct(imgUrl, productsOnShelves[i]));
        }
    }
    IEnumerator ApplyImageToProduct(string MediaUrl, GameObject product)
    {
        UnityWebRequest request = UnityWebRequestTexture.GetTexture(MediaUrl);
        yield return request.SendWebRequest();
        if (request.isNetworkError || request.isHttpError)
        {
            Debug.Log(request.error);
        }
        else
        {
            product.GetComponent<Renderer>().material.mainTexture =
((DownloadHandlerTexture)request.downloadHandler).texture;
        }
    }
    public List<Product> GetAllProducts()
    {
        List<Product> all = new List<Product>();
        for (int i = 1; i < 4; i++)
        {
            Dictionary<string, string> dic = new Dictionary<string, string>();
            dic.Add("per_page", "100");
            dic.Add("page", i.ToString());
            var task = _wcObject.Product.GetAll(dic);
            task.Wait();
            var res = task.Result;
            var ret = res.OrderBy(x => x.sku).ToList();
            all.AddRange(ret);
        }
        return all;
    }
}
```

7.5.2 Login Class Code

The login class contains all the necessary coding for any customer to get register, apply for new password or login into the project. The details of the user are sent to the server and the response from server are passed through Json object to categorize the response and take relevant actions. In case of resetting the password, server is requested to send a auto generated reset link to the user's email. All the users are registered to the server directly.

```
using System;
using System.Collections;
using System.Collections.Generic;
using System.IO;
using UnityEngine;
using UnityEngine.UI;
using UnityEngine.SceneManagement;
using WooCommerceNET;
//using WooCommerceNET.WooCommerce.v3;
using WooCommerceNET.WooCommerce.v2;
using System.Net;

public class LogInForm : MonoBehaviour
{
    public InputField usernameInput;
    public InputField passwardInput;
    public Button gotoLoginButton;
    public Button registerButton;
    public Button closeButton;
    public Button ForgetPassButton;
    public Text textstring1;
    public Text textstring2;
    Orders o = new Orders();
    ArrayList credentials;
    void Start()
    {
        registerButton.onClick.AddListener(Register);
        gotoLoginButton.onClick.AddListener(goToLoginScene);
        closeButton.onClick.AddListener(Exit);
        ForgetPassButton.onClick.AddListener(Forget_Password);
    }
    public void goToLoginScene()
    {
        #region Login
        string Username = usernameInput.text;
        string password = passwardInput.text;
        string url1 = "https://tradeways.pk/api/user/generate_auth_cookie/?username=";
        string url2 = "&password=";
        string website_url = url1 + Username + url2 + password;
        HttpWebRequest request = (HttpWebRequest)HttpWebRequest.Create(website_url);
        HttpWebResponse response = (HttpWebResponse)request.GetResponse();
        Stream dataStream = response.GetResponseStream();
        StreamReader reader = new StreamReader(dataStream);
        string responseFromServer = reader.ReadToEnd();
        StreamReader stream = new StreamReader(response.GetResponseStream());
        string json = reader.ReadToEnd();
        var httpResponse = (HttpWebResponse)request.GetResponse();
        data json_text = JsonUtility.FromJson<data>(responseFromServer);
        if (json_text.status == "ok")
        {
            o.uid = json_text.user.id;
            Debug.Log("Login Successfully");
            gameObject.SetActive(false);
            Debug.Log(json_text);
        }
        else
        {
            textstring2.text = " ";
            textstring1.text = "Incorrect Username & password";
            Debug.Log("Incorrect UserNAME & password");
        }
```

```
        #endregion
    }
    public void Forget_Password()
    {
        string url_forgetPass = "https://tradeways.pk/api/user/retrieve_password/?user_login=";
        string userName_forgetpass = usernameInput.text;
        string url_forget = url_forgetPass + userName_forgetpass;
        HttpWebRequest request = (HttpWebRequest)HttpWebRequest.Create(url_forget);
        HttpWebResponse response = (HttpWebResponse)request.GetResponse();
        Stream dataStream = response.GetResponseStream();
        StreamReader reader = new StreamReader(dataStream);
        string responseFromServer = reader.ReadToEnd();
        StreamReader stream = new StreamReader(response.GetResponseStream());
        string json = reader.ReadToEnd();
        var httpResponse = (HttpWebResponse)request.GetResponse();
        textstring1.text = " ";
        textstring2.text = "        Email Successfully Sent!";
    }
    public void Register()
    {
        string user_name = usernameInput.text;
        string pass_word = passwardInput.text;
        string url_nonce =
  "https://tradeways.pk/api/get_nonce/?controller=user&method=register";
        HttpWebRequest request = (HttpWebRequest)HttpWebRequest.Create(url_nonce);
        HttpWebResponse response = (HttpWebResponse)request.GetResponse();
        Stream dataStream = response.GetResponseStream();
        StreamReader reader = new StreamReader(dataStream);
        string responseFromServer = reader.ReadToEnd();
        StreamReader stream = new StreamReader(response.GetResponseStream());
        string json = reader.ReadToEnd();
        var httpResponse = (HttpWebResponse)request.GetResponse();
        string nonce = "";
        RegisterData json_text = JsonUtility.FromJson<RegisterData>(responseFromServer);
        if (json_text.status == "ok")
        {
            //RegisterData json_text =
JsonUtility.FromJson<RegisterData>(responseFromServer);
            nonce = json_text.nonce;
            string name_reg = user_name;
            //&notify=both&user_pass=YOUR-PASSWORD
            string url_nonce2 = "https://tradeways.pk/api/user/register/?username=" +
                name_reg +
                "&email=" +
                user_name +
                "&nonce=" +
                nonce +
                "&display_name=" +
                name_reg +
                "&notify=both&user_pass=" +
                pass_word;
            HttpWebRequest request2 = (HttpWebRequest)HttpWebRequest.Create(url_nonce2);
            HttpWebResponse response2 = (HttpWebResponse)request2.GetResponse();
            Stream dataStream2 = response2.GetResponseStream();
            StreamReader reader2 = new StreamReader(dataStream2);
            string responseFromServer2 = reader2.ReadToEnd();
            StreamReader stream2 = new StreamReader(response2.GetResponseStream());
            string json2 = reader2.ReadToEnd();
            var httpResponse2 = (HttpWebResponse)request2.GetResponse();
            User_register_error json_text_error =
JsonUtility.FromJson<User_register_error>(responseFromServer);
            User_register json_confirm_reg =
JsonUtility.FromJson<User_register>(responseFromServer);
            if (json_confirm_reg.status == "ok")
            {
                textstring1.text = " ";
                textstring2.text = " Account Registered Successfully !";
            }
            else
            {
                textstring2.text = $"'{json_text_error.error}'";
            }
        }
        else
        {
            textstring2.text = " ";
            textstring1.text = "Unable to register due to connection with server please try again";
        }
    }
    public void Exit()
    {
        Application.Quit();
    }
}
[Serializable]
public class data
{
    public string status;
    public string cookie;
    public string cookie_admin;
    public string cookie_name;
    public User_Custom user;
}
[Serializable]
public class User_Custom
{
    public int id;
    public string username;
    public string nicename;
    public string email;
    public string url;
    public string registered;
    public string displayname;
    public string firstname;
    public string lastname;
    public string nickname;
    public string description;
    public string capabilities;
    public string avatar;
}
[Serializable]
public class RegisterData
{
    public string status;
    public string controller;
    public string method;
    public string nonce;
}
[Serializable]
public class User_register_error
{
    public string status;
    public string error;
}
[Serializable]
public class User_register
{
    public string status;
    public string cookie;
    public string cookie_admin;
    public string cookie_name;
    public string user_id;
    public string username;
}
```

7.5.3 Order Class Code

Order class is called from POS terminal is it is intended to finalize the order along with complete details of the user and cart. These details are taken from user and stored in a list of class OrderBilling. The details are transferred when the GetBillingData method is called, it returns the object. That object contains complete detail which is then passed to server to save order while customer id is called from login class which was saved on successful login.

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;
using UnityEngine.UI;
using WooCommerceNET;
//using WooCommerceNET.WooCommerce.v3;
using System.IO;
using WooCommerceNET.WooCommerce.v2;
using System.Net;
using System;

public class Orders : MonoBehaviour
{
    public InputField firstname;
    public InputField lastname;
    public InputField address_1;
    public InputField address_2;
    public InputField city;
    public InputField state;
    public InputField postcode;
    public InputField country;
    public InputField email;
    public InputField phone;
    public InputField option;
    public int uid;
    public Button closeButton;
```

```
 public Button checkoutButton;
 [SerializeField] GameObject recieptPanel;
 // Start is called before the first frame update
 Order Ap_Order = new Order();
 Cart crt = new Cart();
 Cart cart;
 Order placeorder = new Order();
 OrderShipping ordShipping = new OrderShipping();
 //[SerializeField] List<OrderBilling> ordbill_list = new List<OrderBilling>();
 void Start()
 {
     cart = FindObjectOfType<Cart>();

     List<OrderLineItem> crt_item = cart.GetProductsInCart();
     closeButton.onClick.AddListener(ClosePanel);
     checkoutButton.onClick.AddListener(RecieptPanel);
 }
public OrderBilling  GetBillingData()
 {
     OrderBilling ordBilling = new OrderBilling();
     //first_name { get; set; }
     ordBilling.first_name = firstname.text;
     //[DataMember(EmitDefaultValue = false)]
     //public string last_name { get; set; }
     ordBilling.last_name = lastname.text;
     //[DataMember(EmitDefaultValue = false)]
     //public string company { get; set; }
     ordBilling.company = "Virtual Mart";
     //[DataMember(EmitDefaultValue = false)]
     //public string address_1 { get; set; }
     ordBilling.address_1 = address_1.text;
     //[DataMember(EmitDefaultValue = false)]
     //public string address_2 { get; set; }
     ordBilling.address_2 = address_2.text;
```

```
//[DataMember(EmitDefaultValue = false)]
//public string city { get; set; }
ordBilling.city = city.text;
//[DataMember(EmitDefaultValue = false)]
//public string state { get; set; }
ordBilling.state = state.text;
//[DataMember(EmitDefaultValue = false)]
//public string postcode { get; set; }
ordBilling.postcode = postcode.text;
//[DataMember(EmitDefaultValue = false)]
//public string country { get; set; }
ordBilling.country = country.text;
//[DataMember(EmitDefaultValue = false)]
//public string email { get; set; }
ordBilling.email = email.text;
//[DataMember(EmitDefaultValue = false)]
//public string phone { get; set; }
ordBilling.phone = phone.text;
//ordBilling.option = option;
//-----------------------------------shipping-------------------
//ublic string first_name { get; set; }
ordShipping.first_name = ordBilling.first_name;
//[DataMember(EmitDefaultValue = false)]
//public string last_name { get; set; }
ordShipping.last_name = ordBilling.last_name;
//[DataMember(EmitDefaultValue = false)]
//public string company { get; set; }
ordShipping.company = ordBilling.company;
//[DataMember(EmitDefaultValue = false)]
//public string address_1 { get; set; }
ordShipping.address_1 = ordBilling.address_1;
//[DataMember(EmitDefaultValue = false)]
//public string address_2 { get; set; }
ordShipping.address_2 = ordBilling.address_2;
```

```
    //[DataMember(EmitDefaultValue = false)]
    //public string city { get; set; }
    ordShipping.city = ordBilling.city;
    //[DataMember(EmitDefaultValue = false)]
    //public string state { get; set; }
    ordShipping.state = ordBilling.state;
    //[DataMember(EmitDefaultValue = false)]
    //public string postcode { get; set; }
    ordShipping.postcode = ordBilling.postcode;
    //[DataMember(EmitDefaultValue = false)]
    //public string country { get; set; }
    ordShipping.country = ordBilling.country;
    //-----------------------------------shipping-----------------
    return ordBilling;
}
public void Place_Order(Cart cart)
{
    Time time = new Time();
    Ap_Order.line_items = cart.GetCartItems();
    Ap_Order.customer_id = uid;
    Ap_Order.currency = "PKR";
    Ap_Order.status = "pending";
    Ap_Order.created_via = "rest-api";

    if(option.text == "1")
    {
        Ap_Order.customer_note = "9:00 am - 12:00 pm";
    }
    else if (option.text == "2")
    {
        Ap_Order.customer_note = "12:00 pm - 3:00 pm";
    }
    else if (option.text == "3")
    {
            Ap_Order.customer_note = "3:00 am - 6:00 pm";
        }
        else
        {
            Ap_Order.customer_note = "Timmings is not issue";
        }
        Ap_Order.customer_note = option.text;
        Ap_Order.billing = GetBillingData();
        Ap_Order.shipping = ordShipping;
        Ap_Order.payment_method = "COD";
        Ap_Order.payment_method_title = "Cash on Delivery";

        Ap_Order.total = crt.totalbill();
        try
        {
            API._wcObject.Order.Add(Ap_Order);
        }
        catch
        { }
    }
    public void ClosePanel()
    {
        gameObject.SetActive(false);

    }
    public void RecieptPanel()
    {
        gameObject.SetActive(false);
        recieptPanel.SetActive(true);
    }
}
```

7.5.4 ProductItem Class Code

From this class the AddToCart method is called by the trigger event when user select item to add in cart. This event gets the details of product previously stored in productinfo at time of display. These details are passed along to add details in the list of carts.

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;
using WooCommerceNET;
using WooCommerceNET.WooCommerce.v2;

public class ProductItem : MonoBehaviour
{
    public Product productinfo;
    [SerializeField] Cart cart;

    private void Start()
    {
        cart = FindObjectOfType<Cart>();
    }
    public void SetProductInfo(Product productinfo)
    {
        this.productinfo = productinfo;
    }
    public void OnMouseDown()
    {
        Debug.Log("Selected: " + productinfo.name);
        cart.AddToCart(productinfo);
    }
}
```

7.5.5 RecieptPanel Class Code

This is to close the panel of receipt by setting it as false and closing the window for user-on-user request.

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;

public class ReceiptPanel : MonoBehaviour
{
    public void ClosePanel()
    {
        gameObject.SetActive(false);
    }
}
```

7.5.6 CartContent Class Code

The main objective of this part is to display the items inside the cart by calling the method in cart class and get all the data of items that are selected by the user along with the quantity, price and amount and is displayed onscreen to user. It is mainly done by printing the list in loop where loop runs for n number of times where n is the number of products in the cart.

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;
using UnityEngine.UI;
using WooCommerceNET.WooCommerce.v2;

public class CartContentsPanel : MonoBehaviour
{
    List<OrderLineItem> productsInCart;
    [SerializeField] Cart cart;
    [SerializeField] Text cartContents;
    OrderLineItem ordLine = new OrderLineItem();
    private void Start()
    {
        cart = FindObjectOfType<Cart>();
    }
    public void OpenPanel()
    {
        string gap = "\t\t";
        if (cart != null)
        {
            productsInCart = cart.GetProductsInCart();
            string cartContentsText = "Your Cart:\n";
            cartContentsText = "Item" + gap + "Quantity" + gap + "Price" + "Amount" + "\n";
            for (int loop=0;loop<productsInCart.Count;loop++)
            {
                ordLine = productsInCart[loop];
                cartContentsText += ordLine.name + gap + ordLine.quantity + gap + ordLine.price
+ gap + ordLine.price * ordLine.quantity + "\n";
            }
            cartContents.text = cartContentsText;
        }

        gameObject.SetActive(true);
    }

    public void ClosePanel()
    {
        gameObject.SetActive(false);
    }
}
```

7.5.7 PosTerminal Class Code

Checkout button is visible to user when customer enters in the region of counter. This is handled by trigger events of enter and exit. On checkout the object of cart is called, and all the data is passed to the order's class to place order after which the generated receipt is displayed to the customer on screen

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;
using UnityEngine.UI;
using WooCommerceNET.WooCommerce.v2;

public class PosTerminal : MonoBehaviour
{
    [SerializeField] Button checkout;
    [SerializeField] Text receipt;
    [SerializeField] GameObject CheckoutPanel;

    /* [SerializeField]*/
    Cart cart;
    [SerializeField] List<OrderLineItem> productsInCart;
    Orders ord = new Orders();
    OrderLineItem ordLine = new OrderLineItem();
    // Start is called before the first frame update
    void Start()
    {
        cart = FindObjectOfType<Cart>();
    }
    public void CheckOut()
    {
        Debug.Log("Checkout pressed");
        CheckoutPanel.SetActive(true);
        productsInCart = cart.GetProductsInCart();
        string gap = "\t\t";
        string receiptContent = "Your receipt:\n\n";
        for (int loop = 0; loop < productsInCart.Count; loop++)
        {
            ordLine = productsInCart[loop];
            receiptContent += ordLine.name + gap + ordLine.quantity + gap + ordLine.price +
gap + ordLine.price * ordLine.quantity + "\n";
        }
        for (int loop = 0; loop < productsInCart.Count; loop++)
        {
            productsInCart.ToString();
            OrderLineItem od = new OrderLineItem();
            od = productsInCart[loop];
            receiptContent += "\n--------------------------------------------------------\n";
            receiptContent += od.name + "                Quantity:" + od.quantity + "\n";
        }
        receipt.text = receiptContent;
        try { ord.Place_Order(cart); }
        catch { }
    }
    private void OnTriggerEnter(Collider other)
    {
        Debug.Log("Proceed to checkout");
        checkout.gameObject.SetActive(true);
    }

    private void OnTriggerExit(Collider other)
    {
        Debug.Log("Proceed to checkout");
        checkout.gameObject.SetActive(false);
    }
}
```

CHAPTER 08

TESTING

8 Testing

8.1 Black Box Testing

Black Box Testing is a software testing method in which the internal structure / design / implementation of the item being tested is NOT known to the tester, it is generally done by software testers based on specifications or requirements of the software without reference to its internal workings [49].

Module: Product List Testing (Test - Case)

Table 8.1: Product List Testing (Test - Case)

S. No.	Steps (Description)	Expected Result	Actual Result	Pass / Fail
1	There is a button on the web page and button should be clicked for product list	Product list page will appear	Product list page will be displayed	Pass
2	Product list should not contain duplicate products	No duplicate products in the list	All products are valid	Pass
3	Product list should contain the SKU, product name, price, weight and link to product details.	All information regarding product will be displayed	Displayed product information	Pass
4	There is a button on the product list page to add new products	Products will be added	Selected products will be added	Pass
5	Check whether clicking on cart basket tasks the user to shopping cart page or not	Shopping cart will be appeared	Shopping cart page will be displayed	Pass

Module: Trolley Cart Testing (Test - Case)

Table 8.2: Trolley Cart Testing (Test - Case)

S. No.	Steps (Description)	Expected Result	Actual Result	Pass / Fail
1	There is a button “view cart” to display all the products which users selected	All valid products will be shown	All selected products will appear	Pass
2	User should be able to update the Trolley Cart	Cart will be updated	Not updated	Pass
3	Check whether clicking on checkout button calculating the total price correctly or not	Cart properly calculated total price of products	Accurately total price will be displayed	Pass

Module: Counter Testing (Test - Case)

Table 8.3: Counter Testing (Test - Case)

S. No.	Steps (Description)	Expected Result	Actual Result	Pass / Fail
1	Visible check out button near the counter	Check out button visible	Check out button visible	Pass
2	Place Order on button click	Order placed	Order placed	Pass

8.2 White Box Testing

White Box Testing is a software testing method in which the internal structure / design / implementation of the item being tested is known to the tester, it is generally done by software developer since it requires access to the source code of the program or the project. It is better to perform it during unit testing phase [50].

Module: Fetching Products Testing (Test - Case)

Table 8.4: Fetching Products Testing (Test - Case)

S. No.	Steps (Description)	Expected Result	Actual Result	Pass / Fail
1	PNG file will be displayed	Display image on shelves	Display image on shelves	Pass
2	JPEG file will be displayed	Display image on shelves	Displaying image on shelves	Pass
3	SKU	Read string values and use it for other functionalities	Reading SKU and being use in other functionalities	Pass

Module: Trolley Cart Testing (Test - Case)

Table 8.5: Trolley Cart Testing (Test - Case)

S. No.	Steps (Description)	Expected Result	Actual Result	Pass / Fail
1	Fetching details through SKU	Fetch details and displayed in cart	Fetch details and displayed in cart	Pass

Module: Counter Testing (Test - Case)

Table 8.6: Counter Testing (Test - Case)

S. No.	Steps (Description)	Expected Result	Actual Result	Pass / Fail
1	Read SKU from cart and place order	Read SKU and order placed	Read SKU and order placed	Pass

8.3 Deployment

Agile is the collection of beliefs, that teams can use for making decisions about how to do the work of developing software. The reason for nominating the agile as strategy is that agile does not make decisions for you. Instead, it gives a foundation for teams to make decisions that result in better software development. Agile manifesto uncovers better ways of developing software by doing it and helping others do it. Through this we have come to value individuals and interactions over processes and tools, working software over comprehensive documentation, responding to change over following a plan.

The architecture we are using in this Virtual Reality Mart application is 3-tier architecture because it is a web-based application. In this architecture at the client side the interface helps to make the connectivity between the client and the database server. This interface is the unity-based application and to support this application the business layer exists in between the client and the database server. This intermediate layer is also known as application server. Every query generated by the client is processed at the application layer of the architecture.

CHAPTER 09

CONCLUSION

9 Conclusion

The way we shop in the old days has now been changed because of the new trend of shopping from the supermarket virtually. This happened because shopping online from ecommerce website like Daraz was not at all joyful for the customers who love the real-life shopping experience. We have an ecommerce setup for our project which provides it users a real-life shopping experience right from the comfort of their home without worrying about the traffic, crowd which are vulnerable especially in the situation of Covid-19. The customers by simply sign into the website can start experiencing the virtual super mart. Shopping exactly like he or she would shop in the super mart physically.

The application was also designed to reduce the overflow of customers in supermarkets as there was high probability of spreading of the Covid-19 virus. On the other hand, it was also necessary for people to buy groceries for the whole moth or for weeks in one go. Other advantage of this application was that it creates white collar jobs for delivery boys, IT specialists and many of those who had lost their jobs during the Lockdown due Covid-19 this does not play part in increasing the employment rate. Overall, we can conclude that this project is a great entrepreneurship idea and has been successful according to our expectations. But we are still in learning phase we can do more research and make it even more better by adding servers and features in the application. This application is user friendly through which we users can easily perform shopping from their home.

CHAPTER 10

FUTURE WORK

10 Future Work

This project “Virtual Super Mart” is based on mobile based application. The purpose of developing this application that users can do shopping by their smartphones with user friendly interface along with various options. To make the application more beneficial for the customer we can endeavor to improve on some areas of the project for betterment of the outcome.

Following are some suggestions and the areas are also mentioned listed below for future enhancement in this project.

1. We can enhance our application in accordance with e-commerce criteria to meet every customer’s desire and needs by facilitating the customers with more and more products and their variations for customers to buy the goods in better or high satisfaction manner.
2. Another way through which the future enhancement can be made by introducing the loyalty card and club membership card. With the use of customer redemption and award facility. This system will act as an incentive for the customers.
3. We can increase enhancement of our project by providing better user experience to our customers. We can achieve this feature by making the scenery more and more attractive and appealing to the customers so that it can be easy to provide or deliver the actual value to the customers.
4. Another area that could be focus of an individual for future enhancement is that we can deploy advertisements around the shelves. The advertisements will be deployed in such a manner that as soon the customers come near or pass a shelf the advertisement of the product present in that shelf will be displayed. Keeping in mind that the advertisements video should be of few seconds.

CHAPTER 11

ACHIEVEMENTS

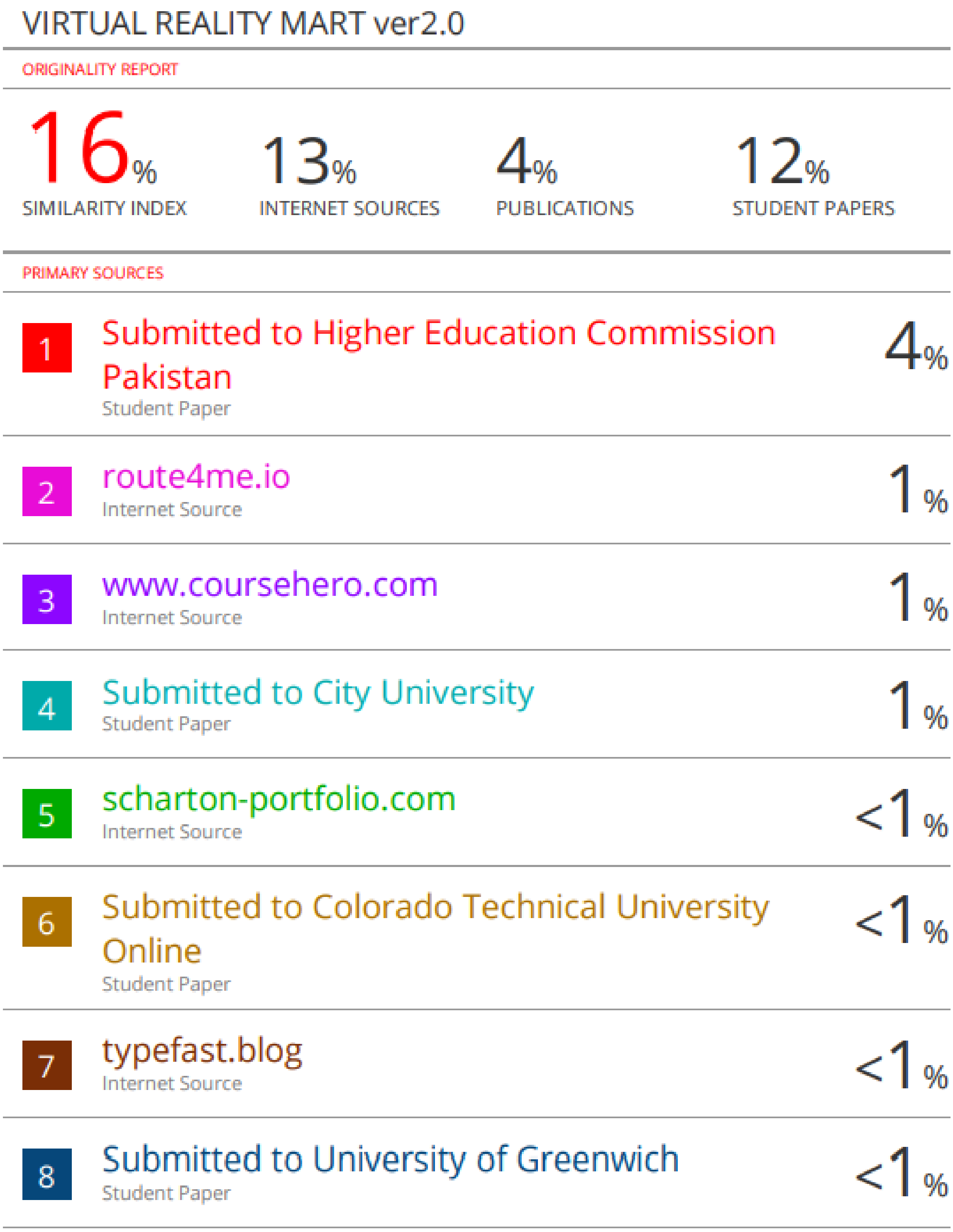
11 Achievements

We pitched our idea of virtual super mart to the Tradeways. In the organization we contact person is Mr. Khurram Shehzad. The designation of Mr. Khurram in Tradeways is Business Manager. We discussed the whole project plan with Mr. Khurram Shehzad. They started to take interest in our project because it is one of its kind in the current market. They agreed to share the information of the products they were providing to their customers. The data Tradeways shared with us was a live data. They have said that they will go to the deployment phase after seeing the final product. We were able to manage the live data of the Tradeways from scratch. We provided Tradeways with admin panel to manage their products and their customer’s orders very efficiently and effectively. We also provided them the end user application for their customers.

CHAPTER 12

PLAGARISM REPORT

12 Plagiarism Report



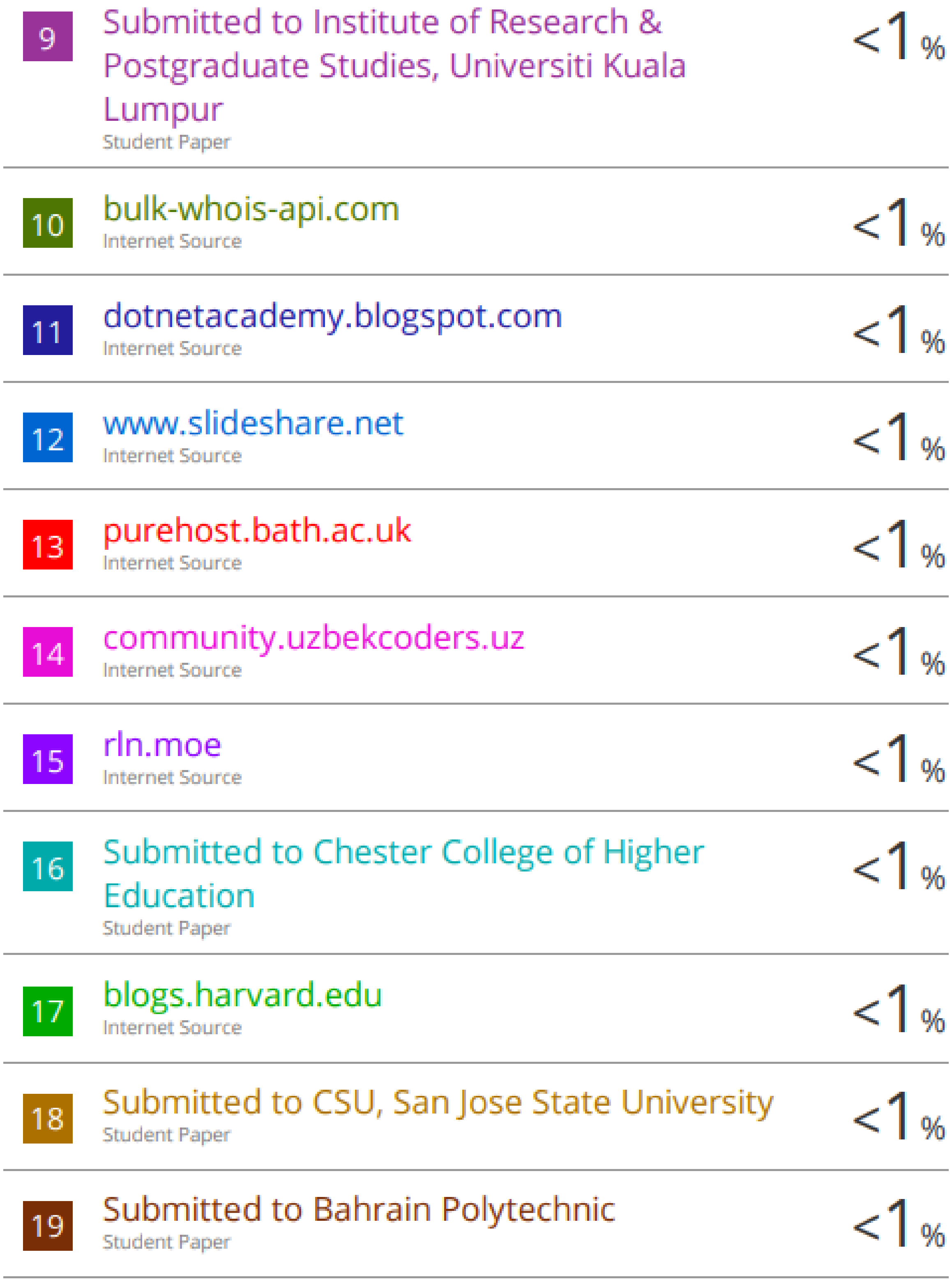
VIRTUAL REALITY MART ver2.0

ORIGINALITY REPORT

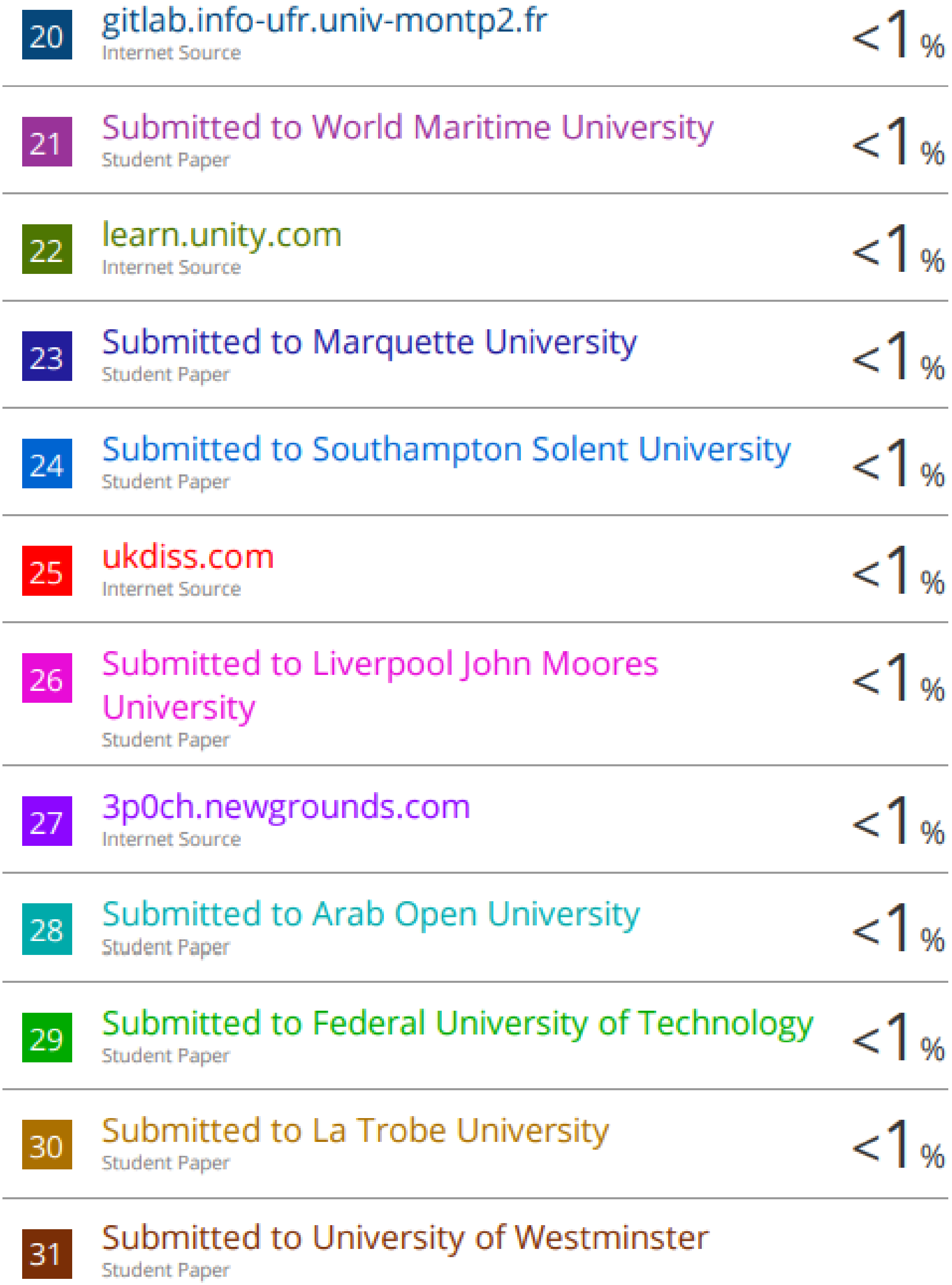
16%	13%	4%	12%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

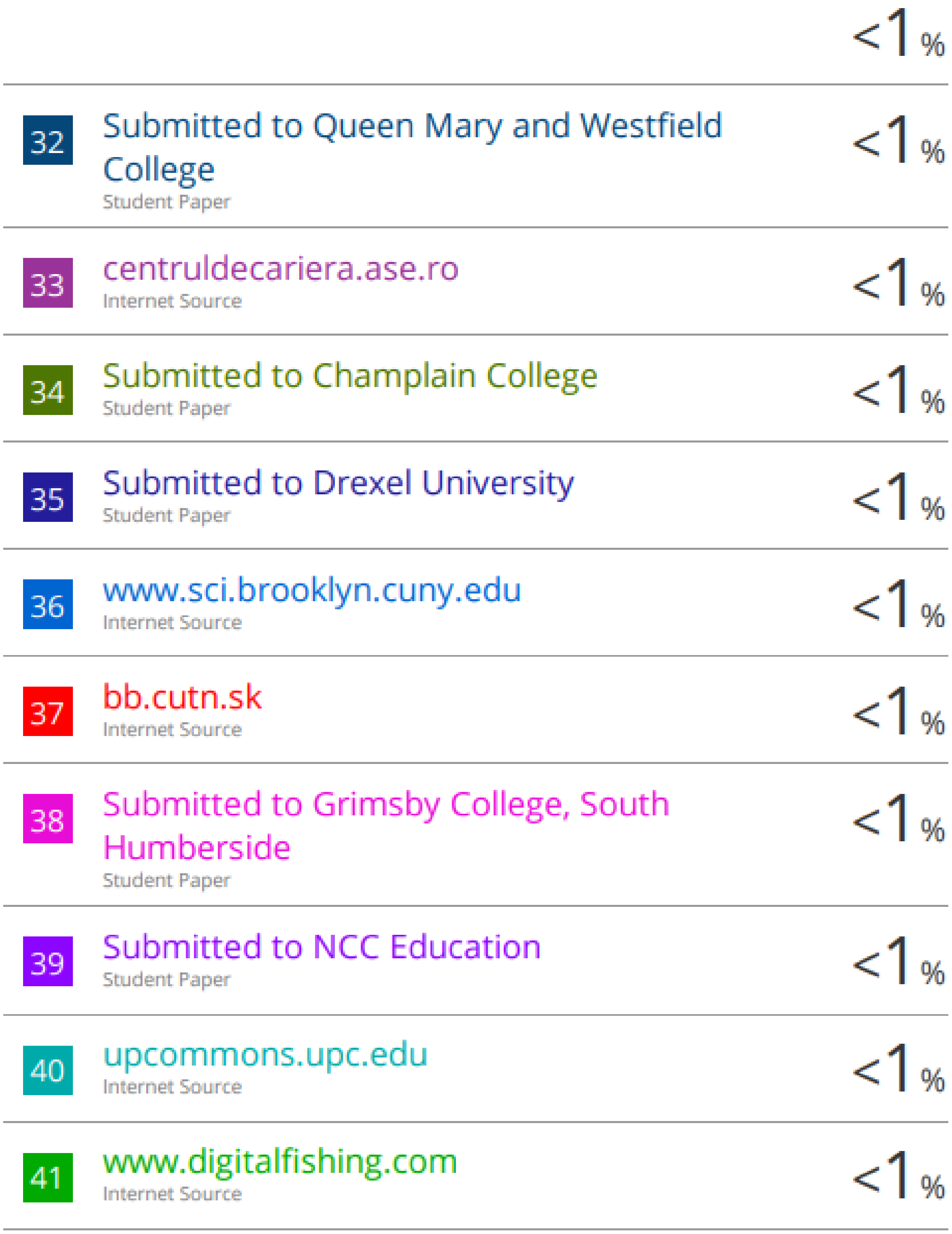
1	Submitted to Higher Education Commission Pakistan Student Paper	4%
2	route4me.io Internet Source	1%
3	www.coursehero.com Internet Source	1%
4	Submitted to City University Student Paper	1%
5	scharton-portfolio.com Internet Source	<1%
6	Submitted to Colorado Technical University Online Student Paper	<1%
7	typefast.blog Internet Source	<1%
8	Submitted to University of Greenwich Student Paper	<1%



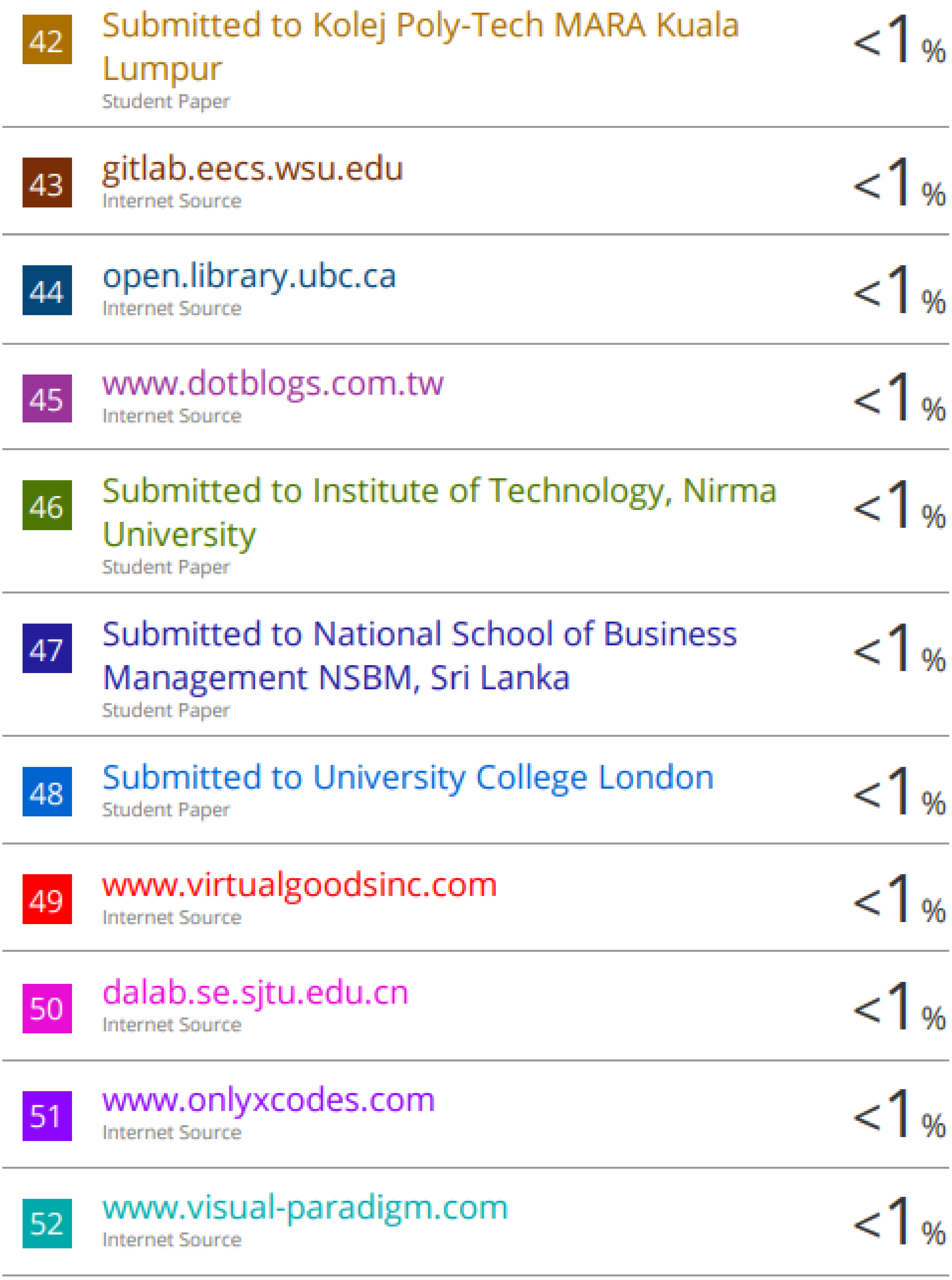
9	Submitted to Institute of Research & Postgraduate Studies, Universiti Kuala Lumpur Student Paper	<1 %
10	bulk-whois-api.com Internet Source	<1 %
11	dotnetacademy.blogspot.com Internet Source	<1 %
12	www.slideshare.net Internet Source	<1 %
13	purehost.bath.ac.uk Internet Source	<1 %
14	community.uzbekcoders.uz Internet Source	<1 %
15	rln.moe Internet Source	<1 %
16	Submitted to Chester College of Higher Education Student Paper	<1 %
17	blogs.harvard.edu Internet Source	<1 %
18	Submitted to CSU, San Jose State University Student Paper	<1 %
19	Submitted to Bahrain Polytechnic Student Paper	<1 %



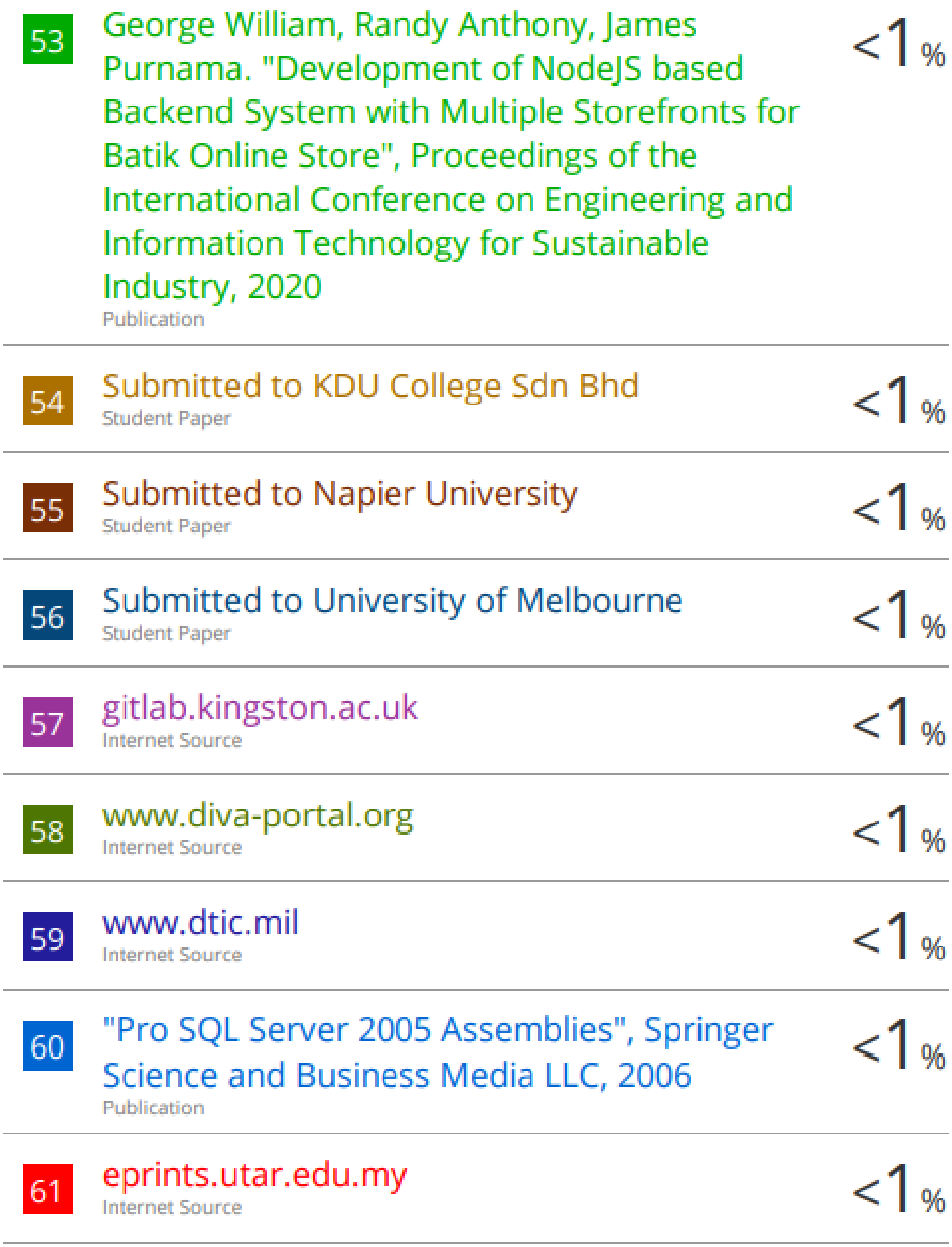
20	gitlab.info-ufr.univ-montp2.fr Internet Source	<1 %
21	Submitted to World Maritime University Student Paper	<1 %
22	learn.unity.com Internet Source	<1 %
23	Submitted to Marquette University Student Paper	<1 %
24	Submitted to Southampton Solent University Student Paper	<1 %
25	ukdiss.com Internet Source	<1 %
26	Submitted to Liverpool John Moores University Student Paper	<1 %
27	3p0ch.newgrounds.com Internet Source	<1 %
28	Submitted to Arab Open University Student Paper	<1 %
29	Submitted to Federal University of Technology Student Paper	<1 %
30	Submitted to La Trobe University Student Paper	<1 %
31	Submitted to University of Westminster Student Paper	



		<1 %
32	Submitted to Queen Mary and Westfield College Student Paper	<1 %
33	centruldecariera.ase.ro Internet Source	<1 %
34	Submitted to Champlain College Student Paper	<1 %
35	Submitted to Drexel University Student Paper	<1 %
36	www.sci.brooklyn.cuny.edu Internet Source	<1 %
37	bb.cutn.sk Internet Source	<1 %
38	Submitted to Grimsby College, South Humberside Student Paper	<1 %
39	Submitted to NCC Education Student Paper	<1 %
40	upcommons.upc.edu Internet Source	<1 %
41	www.digitalfishing.com Internet Source	<1 %



42	Submitted to Kolej Poly-Tech MARA Kuala Lumpur Student Paper	<1%
43	gitlab.eecs.wsu.edu Internet Source	<1%
44	open.library.ubc.ca Internet Source	<1%
45	www.dotblogs.com.tw Internet Source	<1%
46	Submitted to Institute of Technology, Nirma University Student Paper	<1%
47	Submitted to National School of Business Management NSBM, Sri Lanka Student Paper	<1%
48	Submitted to University College London Student Paper	<1%
49	www.virtualgoodsinc.com Internet Source	<1%
50	dalab.se.sjtu.edu.cn Internet Source	<1%
51	www.onlyxcodes.com Internet Source	<1%
52	www.visual-paradigm.com Internet Source	<1%



53	George William, Randy Anthony, James Purnama. "Development of NodeJS based Backend System with Multiple Storefronts for Batik Online Store", Proceedings of the International Conference on Engineering and Information Technology for Sustainable Industry, 2020 Publication	<1 %
54	Submitted to KDU College Sdn Bhd Student Paper	<1 %
55	Submitted to Napier University Student Paper	<1 %
56	Submitted to University of Melbourne Student Paper	<1 %
57	gitlab.kingston.ac.uk Internet Source	<1 %
58	www.diva-portal.org Internet Source	<1 %
59	www.dtic.mil Internet Source	<1 %
60	"Pro SQL Server 2005 Assemblies", Springer Science and Business Media LLC, 2006 Publication	<1 %
61	eprints.utar.edu.my Internet Source	<1 %

62	es.scribd.com Internet Source	<1 %
63	ftp.mdt.mt.gov Internet Source	<1 %
64	ijrar.org Internet Source	<1 %
65	www.hivmr.com Internet Source	<1 %
66	www.geeksforgeeks.org Internet Source	<1 %
67	Daniel M. Solis. "Illustrated C# 2012", Springer Science and Business Media LLC, 2012 Publication	<1 %
68	Jesse Liberty, Paul Betts. "Programming Reactive Extensions and LINQ", Springer Science and Business Media LLC, 2011 Publication	<1 %
69	Pro NET 1 1 Network Programming, 2004. Publication	<1 %

Exclude quotes	On	Exclude matches	< 3 words
Exclude bibliography	On		

CHAPTER 13

REFERENCES & ANNEXURE

13 References and Annexure

13.1 References

[1] N. Hebb, “Flow Chart Diagram”. [Online]. Available: https://www.breezetree.com/articles/what-is-a-flow-chart

[2] A. Tikkanen, “Walmart”, 2021. [Online]. Available: https://www.britannica.com/topic/Walmart

[3] M. S. S. Jajja, M. N. Jat, “Online Marketplace’s Value Chain”,2019. [Online]. Available: https://journals.sagepub.com/doi/full/10.1177/0972820119825976

[4] T. Hassan, “Are Daraz pick-up points a good business strategy?”,2019. [Online]. Available:https://profit.pakistantoday.com.pk/2019/10/08/are-daraz-pick-up-points-a-good-business-strategy/

[5] Anonymous, “Carrefour”. [Online]. Available: https://www.companieshistory.com/carrefour/

[6] Manjusha, “How much does it cost to develop an ecommerce platform like carrefour?”, 2020 [Online]. Available: https://appdevelopmentcompany97739586.wordpress.com/2020/04/27/how-much-does-it-cost-to-develop-an-ecommerce-platform-like-carrefour/

[7] Z. N. Paracha,“ Retailo raises $2.3 million pre-seed for its B2B ecommerce marketplace in Saudi & Pakistan”, 2020. [Online]. Available: https://www.menabytes.com/retailo-pre-seed/

[8] Anonymous, “System. Collections”, [Online]. Available: https://docs.microsoft.com/en-us/dotnet/api/system.collections?view=net-5.0

[9] Anonymous, “System. Collections. Generic”, [Online]. Available: https://docs.microsoft.com/en-us/dotnet/api/system.collections.generic?view=net-5.0

[10] Nightbane30, “UnityEditor”, 2015. [Online], Available: https://answers.unity.com/questions/1024303/using-unityeditor-throws-an-error.html

[11] A. Venkat, “UnityEngine.UIElement”, 2020. [Online], Available: https://www.raywenderlich.com/6452218-uielements-tutorial-for-unity-getting-started

[12] Anonymous, “UnityEngine.Networking”, [Online], Available:

https://docs.unity3d.com/Packages/com.unity.multiplayer-hlapi@1.0/api/UnityEngine.Networking.html

[13] PatrickVDM, “UnityEngine.UI”, 2019. [Online], Available: https://forum.unity.com/threads/unityengine-ui-deprecated-in-unity-2019-2.778883/

[14] Anonymous, “NewtonSoft.json”, [Online], Available: https://docs.unity3d.com/Packages/com.unity.nuget.newtonsoft-json@2.0/manual/index.html

[15] Anonymous, “json.Net”, [Online], Available: https://www.newtonsoft.com/json

[16] Anonymous, “json. User”, [Online], Available: https://developer.yammer.com/docs/usersjson

[17] N. Xi, J. Hamari, “Shopping in virtual reality: A literature review and future agenda”, 2021. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0148296321003209

[18] E. Guide, “What is Ecommerce?”, [Online]. Available: https://ecommerceguide.com/guides/what-is-ecommerce/

[19] M. Alcañiz, “Virtual Reality in Marketing: A Framework, Review, and Research Agenda”, 2019. [Online]. Available: https://www.frontiersin.org/articles/10.3389/fpsyg.2019.01530/full

[20] S. Phuyal, “Challenges, Opportunities and Future Directions of Smart Manufacturing: A State of Art Review”, 2020. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2666188820300162

[21] Thirdsem, “Fact Finding Techniques || Fact-Finding Techniques for Requirements Discovery || Bcis Notes”, 2020. [Online]. Available: https://bcisnotes.com/thirdsemester/system-analysis-and-design/fact-finding-techniques-fact-finding/

[22] Anonymous, “Fact finding techniques”,2015. [Online]. Available: https://www.slideshare.net/imthiyasbtm/fact-finding-techniques-53785925

[23] Anonymous, [Online]. Available: https://www.acronymattic.com/Joint-Requireme nt-Planning-(JRP).html

[24] R. F. Dam & T. Y. Siang, “Design Thinking: Get Started with Prototyping”, 2020. [Online]. Available: https://www.interaction-design.org/literature/article/design-thinking-get-started-with-prototyping

[25] K. Siau, “An Analysis of Unified Modeling Language (UML) Graphical Constructs Based on BWW Ontology”, 2015. [Online]. Available: https://www.researchgate.net/publication/242486128_An_Analysis_of_Unified_Modeling_Language_UML_Graphical_Constructs_Based_on_BWW_Ontology

[26] D. Bell, “An introduction to the Unified Modeling Language”, 2003. [Online]. Available: https://developer.ibm.com/articles/an-introduction-to-uml/

[27] D. Ramel, “Visual Studio”, 2019. [Online]. Available: https://visualstudiomagazine.com/articles/2019/04/02/visual-studio-2019.aspx?oly_enc_id=3025F3612090E7X

[28] Anonymous, “An Introduction to C# Programming Language”, [Online]. Available: https://www.universalclass.com/articles/computers/an-introduction-to-c-programming-language.htm

[29] Nick, “Introduction to phpMyAdmin | What is phpMyAdmin?”, 2021. [Online]. Available: https://heytuts.com/web-dev/introduction-to-phpmyadmin-what-is-phpmyadmin

[30] E. Peckham, “How Unity built the world’s most popular game engine”, 2019. [Online]. Available: https://techcrunch.com/2019/10/17/how-unity-built-the-worlds-most-popular-game-engine/

[31] Anonymous, “Functional and Nonfunctional Requirements: Specification and Types”, 2021. [Online]. Available: https://www.altexsoft.com/blog/business/functional-and-non-functional-requirements-specification-and-types/

[32] Anonymous, “Non-functional Requirements: Examples, Types, How to Approach”,2019. [Online]. Available: https://www.altexsoft.com/blog/non-functional-requirements/

[33] B. Elahi, “Process Flow Diagram”, 2018. [Online]. Available: https://www.sciencedirect.com/topics/engineering/process-flow-diagram

[34] S. S. Elawady, “Flow Chart”, 2018. [Online]. Available: https://journals.plos.org/plosone/article?id=10.1371/journal.pone.0195955

[35] Y. Waykar, “Role of use case diagram in software development”, 2015. [Online]. Available:https://www.researchgate.net/publication/322991847_role_of_use_case_diagram_in_software_development

[36] M. Seth,” Use case Diagram”, 2003. [Online]. Available: https://www.developer.com/design/creating-use-case-diagrams/

[37] K. McNeish, “UML Class Diagrams”, 2019. [Online]. Available: https://www.codemag.com/article/0201061/UML-Class-Diagrams

[38] D. Berardi, “Class Diagram”, 2005 [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0004370205000792

[39] A. A. B. Baqais, M. Alshayeb, “Sequence Diagram”, 2018. [Online]. Available: https://journals.plos.org/plosone/article?id=10.1371/journal.pone.0202629

[40] A. Alsaadi, “Communication Diagram” ,2007. [Online]. Available: https://www.researchgate.net/publication/4274154_The_UML_Communication_Diagram_Revisited

[41] C. Britton, J. Doake, “Activity Diagram”, 2005. [Online]. Available: https://www.sciencedirect.com/topics/computer-science/activity-diagram

[42] E. K. Park & M. Evans, “ER Diagram”, 1995. [Online]. Available: https://www.researchgate.net/publication/243781001_A_Comparative_Analysis_of_Entity-Relationship_Diagrams

[43] D. R. S. M. R. Lowry, “Algorithm theories and design tactics”, 1990. [Online]. Available: https://www.sciencedirect.com/science/article/pii/0167642390900259

[44] R. Toulson, T. Wilmshurst, “Pseudocode”,2017. [Online]. Available: https://www.sciencedirect.com/topics/engineering/pseudocode

[45] D. Bell, “Component Diagram” ,2004. [Online]. Available: https://developer.ibm.com/articles/the-component-diagram/

[46] C. Britton, J. Doake, “Deployment Diagram”, 2005. [Online]. Available: https://www.sciencedirect.com/topics/computer-science/deployment-diagram

[47] C. Britton, J. Doake, “State Diagram”, 2005. [Online]. Available: https://www.sciencedirect.com/topics/computer-science/state-diagram

[48] L. Copeland , “State Diagram”, 2003. [Online]. Available: https://www.stickyminds.com/article/state-transition-diagrams

[49] R. Dev, M. Katara, “Black Box Testing”, 2012. [Online]. Available: https://www.sciencedirect.com/topics/computer-science/black-box-testing

[50] G. J. K. V. Wyk , “White Box Testing”, 2013. [Online]. Available: https://us-cert.cisa.gov/bsi/articles/best-practices/white-box-testing/white-box-testing

13.2 Annexure

13.2.1 Organization’s Introduction

Tradeways is an online marketplace digitalizing the B2B business system by shifting the business dealings to e-commerce system. The organization aims to upgrade the retailers to use the technology for their business dealings. The company has provided multiple ways to its customer to view their products and price range and place order as per their convenience. These ways include

- Mobile application
- Website
- Tele sales.

Tradeways Mobile application is available to all users on Google play store. It has a simple and user-friendly GUI facilitating the retailers that are uneducated. The application has an advantage of Urdu mode in which the application language is changed to Urdu since it is highly likely that is their target market, more people might not understand English.

While a good working mobile application is available, tradeways also offers another easy self-serving way of website as a medium to get connected with them and perform business dealings. The website is itself simple however, the service will only be available to the registered users. This ensures the orders are placed by the audience of their target market.

With giving two easy ways to the customers tradeways focuses on tele sales as well. Tele sale is a type or sale in which the company’s representative communicate with the client on call which is one of the most effective way.

Tradeways has been doing a little work or HoReCa i.e Hotels, Restaurant and Café’s. Although this is not their flagship program, but they have a little market input in this department as well.

After working on two different categories of customers Tradeways business manager, Mr. Khurram Shazhad aims to target B2C market in near future. To create a good impact on this segment tradeways has collaborated with our project to innovate a solution for its GT customers.

13.2.2 Detail Proposal

The detail proposal contains the scope of the project, out of scope, objectives, team member’s details, and other supporting points to clear the understanding of the final deliverable project and it’s costs and durations, etc.

Super Mart Based on Virtual Reality

17B-072-CS	Ahmer Irfan
17B-087-CS	Ennas Muhammad
17B-068-CS	Muhammad Ibrahim
17B-020-CS	Sameed Arshad

Batch 2017-18

Date: November 16’ 2020

DEPARTMENT OF COMPUTER SCIENCE
USMAN INSTITUTE OF TECHNOLOGY

Usman Institute of Technology
Department Of Computer Science

FINAL PROJECT APPROVAL FORM

The Head of Department, Date: 16-November-2020
Computer Science Department, Batch: 2017-18
Usman Institute of Technology,
Karachi.

Subject: **Bachelor of Science in Computer Science Final Year Project**

Respect Sir,

We, the below listed students of final Year BS Computer Science class, desire to undertake work on the following project.

Super Mart Based on Virtual Reality

We request you to kindly grant approval for undertaking the work on the above-cited project. I abide by all terms and conditions mentioned below.

1. I have selected this project on my own.
2. I have no objection working under the supervision of male/female supervisor, or if my project work is evaluated by male/female externals.
3. I am sure I can complete this project till **Final Milestone of 8th Semester**.
4. I am eager to work under the supervision of advisor assigned to this project.
5. I understand that FYP committee can modify the scope of the project as and when required.
6. I know that if do not appear in regular project progress presentations/milestones my project will be disqualified.
7. I know that if I do not appear in mid project presentation, whenever it is scheduled, I will not be eligible for final project viva
8. I fully understand that "**cheating**" ***** may lead to cancelation of my project.
9. I understand that the decision of the FYP evaluation committee, for all issues, would be final, and no objections will be accepted.
10. I have no objection presenting my project to external or internal examiner assigned by the Head of the Department.
11. Project and Product deliverables at the time of submission of final year project every group is responsible to submit complete running system along with printed reports, source code, hardware (if any) etc. to the project coordinator.
12. It would be the responsibility of Project coordinator to keep record of all projects in a system (in running form) as it would help to continue next project in continuation, depends upon the scope and application of project.
13. Proper dressing and way of presentation should be in English during proposal defend session, milestones and final presentations.
14. Marking of milestones and final presentation should be based on individual evaluation of each faculty members and marks would be granted during session.
15. When we go for proposal defends session a list of all previous projects with their brief introduction must be available during session for our reference. (Introduction, Scope of project, tools and technology and batch must be available).
16. I understand that it is my responsibility to update my advisor and FYP committee members with the status of my project and submit reports on time.
    - Copying code from any resources
    - Using off the shelf components without prior permission
    - Outsourcing your project

- Hire a resource for the completing the FYP code or any part of the project.

Yours sincerely,

S. No	Roll No.	Name	Email	Signature
1	17B-072-CS	Ahmer Irfan	17b-072-cs@students.uit.edu	
2	17B-087-CS	Ennas Muhammad	17b-087-cs@students.uit.edu	
3	17B-068-CS	Muhammad Ibrahim	17b-068-cs@students.uit.edu	
4	17B-020-CS	Sameed Arshad	17b-020-cs@students.uit.edu	

TABLE OF CONTENTS

Project overview

The main aim of this project is to provide new perception for the users to shift from Online shopping to virtual shopping. This app will allow the users to shop in a manner comparable to a real supermarket. Supermarkets based on virtual reality would be embedded with virtual floor plans, 3D lighting, HD items would be displayed, immersive walk-throughs, and many more. It would also offer a storytelling platform built on customer behavior insights.

Project objectives

The **Super Mart Based on Virtual Reality** project will meet the following objectives:

- Objective #1: Our app will provide a virtual experience of real time shopping for users.
- Objective #2: This will reduce the human efforts and promoting better living and cost effective as the users would have interactive and attractive experience
- Objective #3: Our application will increase the sales as well as a good image for marketing purposes.
- Objective #4: People would be able to purchase, pay and get at their doorsteps.

Project scope

The scope of this project includes and excludes the following items.

In scope:

- Virtual mart
- Placement of products
- Adding products to cart
- Payment method
- Bill generation
- Inventory management
- Admin panel

Out of scope:

- Deliver order (physically)

Deliverables produced.

• Project Deliverable 1:
Following Chapters:
1) Introduction.
2) Problem Statement.
3) Literature Review (Comparative Analysis)

Similar Applications, 3D modeling, Form Research Papers & Books. Actor use case diagram, Activity Diagram(workflow) System diagram.

• Project Deliverable 2:
20% technical use cases. Along with behavioral diagram (sequence, collaboration, state transition, or activity diagram. Based on each use case Class Diagram, Front-end (GUI/ Mobile), Sequence Diagram, Object Diagram and Entity Relation Diagram.

• **Project Deliverable 3:**
Following Chapters:

4) Software Analysis & Requirements.
5)System Architecture.
6)System Design.
7)Algorithm analysis and pseudo code complexity with Plagiarism Check.

Project estimated effort/cost/duration

The total estimated time for this project is 6 months. There will be 6 modules in our project with approximately no such cost except for the cost of uploading apps on Google play store and apple play store. The estimated efforts are of 5 students spending up to 5 till 6 hours each day on the project.

GANTT Chart:

The detailed (tentative) GANTT Chart must be attached with this document. All your deliverables & Milestones are clearly mentioned on it. Supervisor(s) and the external committee member(s) make sure your progress according to your GANTT Chart.

Estimated effort hours:

The estimated hours are 4-5 hours each day for six months.

Estimated duration:

Milestone	Date completed	Deliverable(s) completed
Project planning	09/23/2020	• Project definition • Work plan
Milestone 1	8th week of fall semester	• 20 % Technical UML diagrams
Milestone 2	14th week of fall semester	• Working proof of concept milestone
Milestone 3	12th week of spring semester	• 100 % code of application along with testing
Milestone 4	12th week of spring semester	• Complete working application with deployment
Project conclusion		

Project assumptions

In order to identify and estimate the required tasks and timing for the project, certain assumptions and premises need to be made. Based on the current knowledge today, the project assumptions are listed below. If an assumption is invalidated at a later date, then the activities and estimates in the project plan should be adjusted accordingly.

- Assumption #1: The first assumption we have made that there would be a virtual 3D projection of a hand while selecting the product and the selected product will have complete details of a product such as description, expiry date etc.

- Assumption#2: Two people can interact with each other while shopping.
  For example: if one wants to take suggestions about a product, she/he can interact with the person found nearby.

- Assumption #3: If a person is not able to find a particular corner of a store such as dairy items, there would be a live representative to guide the route.

Project risks

Project risks are characteristics, circumstances, or features of the project environment that may have an adverse effect on the project or the quality of its deliverables. Known risks identified with this project have been included below. A plan will be put into place to minimize or eliminate the impact of each risk to the project.

Risk Area	Level (H/M/L)	Risk Plan
Cyber Attack	H	SSL Certification through API
Server down	H	Maximum number of customers at a time according to server capabilities
Data loss due to cyber attack	H	RAID 5 on the server

Project approach

- Agile Software Development

Tools and technologies

- Visual Studio
- C#
- PhpMyAdmin
- Unity

Expected Final product

Screen shots can be used

Similar products available

- Mine Craft
- Tesco Home plus

List down similar products available online with URL

Clearly mention the URL with date in list order. Also mentioned the SIMILARITY FACTOR of your Project with the available online Project.

Project approvals:

Add any signatures that are important for the approval of the project.

Name:
Project Supervisor Signature

Member, FYP Committee, UIT Signature