**ShopiRound (an application to find products nearby)**

By

**Akik Ashraf Rafi**

Roll: 1907009

&

**Md. Mubtashim Abrar Nihal**

Roll: 1907010



**Supervisor:**

Abdul Aziz

Assistant Professor

Dept. of Computer Science and Engineering Signature

Khulna University of Engineering & Technology

**Department of Computer Science and Engineering**

**Khulna University of Engineering & Technology**

**Khulna 9203, Bangladesh**

**30 November 2023**

**Acknowledgement**

We express our deepest gratitude to Allah for granting us the strength and wisdom to complete this project. We, Akik Ashraf Rafi and Md. Mubtashim Abrar Nihal, are presenting a project report titled “ShopiRound” as a part of “CSE 3200: System Development Project” course.

We wish to express our sincere thanks to our supervisor, Abdul Aziz sir, Assistant Professor, in the Department of Computer Science and Engineering. We are extremely thankful and indebted to him for sharing expertise, sincere, valuable guidance and encouragement extended to us.

Akik Ashraf Rafi

&

Md. Mubtashim Abrar Nihal

**Abstract**

One of the challenges faced by individuals is the immediate need to purchase a product without knowing where to find it. Often, time constraints prevent the exploration of nearby shops, leading to difficulty in locating rare items. This time constraint has become a significant issue in our modern lifestyle. While online ordering is an option, it also involves waiting for the product to arrive. To address this issue, an application has been designed specifically for swiftly locating products available in nearby retail shops. This application offers a quick and effortless way to find any item within your area by showing you available products in your area and showing their distance. Consequently, the need to expend energy and time by inquiring at multiple shops for a specific product is eliminated. The primary objective of this app is to enhance convenience by saving time and energy for users. Moreover, not only does this application benefit buyers, but it also offers advantages to sellers. Shops and their products gain increased visibility, thereby potentially attracting more customers and recognition. This platform aims to streamline the process of product search and purchase, benefiting both buyers and sellers in the process.

**Contents**

**Pages**

Acknowledgement ii

Abstract iii

Contents iv

List of Tables vii

List of Figures viii

**1 Introduction**............................................................................................................................... 1

1.1 Backgorund........................................................................................................................... 1

1.2 Objective............................................................................................................................... 1

1.3 Contribution.......................................................................................................................... 2

1.4 Unfamiliarity of the problem................................................................................................ 2

1.5 Project Timeline.................................................................................................................... 2

**2 Related Works**............................................................................................................................ 4

2.1 Existing Solutions................................................................................................................. 4

2.1.1 Daraz.................................................................................................................................. 4

2.1.2 Rokomari.com.................................................................................................................... 4

2.1.3 Pickaboo............................................................................................................................. 4

2.1.4 Chaldal................................................................................................................................ 4

2.1.5 AjkerDeal............................................................................................................................ 4

2.2 Limitations in Existing Solutions.......................................................................................... 4

**3 System Design**............................................................................................................................. 5

3.1 Analysis of The System........................................................................................................ 5

3.1.1 Dataflow Diagrams........................................................................................................ 5

3.1.1.1 Authentication......................................................................................................... 5

3.1.1.2 Offline Shoping....................................................................................................... 6

3.1.1.3 Online Shoping........................................................................................................ 7

3.1.2 Use case Diagrams.......................................................................................................... 8

3.1.2.1 Use case Diagram from Customer’s Perspective..................................................... 8

3.1.2.2 Use case Diagram from Shop Owner’s Perspective................................................. 9

3.2 System Architecture.............................................................................................................. 10

3.2.1 Routing Algorithm (TSP).............................................................................................. 12

3.2.2 Database Design............................................................................................................ 13

3.2.3 Class Diagram............................................................................................................... 15

3.3 Tools……………….............................................................................................................. 16

3.3.1 ASP.Net MVC…………............................................................................................... 17

3.3.2 Android Studio.............................................................................................................. 17

3.3.3 Geoapify……................................................................................................................ 18

3.3.4 SignalR.......................................................................................................................... 19

3.3.5 Leaflet JS....................................................................................................................... 19

3.3.6 SmarterASP.................................................................................................................... 20

**4. Project Implementation**.......................................................................................................... 21

4.1 System Implementation...................................................................................................... 21

4.1.1 Planning and Preparetion.............................................................................................. 21

4.1.2 Technology Stack and Infrastructure............................................................................ 21

4.1.3 Implementation of Key Features.................................................................................. 22

4.2 User Manual....................................................................................................................... 23

4.3 Morality and Ethical Issues................................................................................................ 33

4.4 Socio-economic Impact and Sustainability........................................................................ 34

4.5 Financial Analysis and Budget…………........................................................................... 35

**5 Conclusion**............................................................................................................................... 37

5.1 Conclusion and Challenges Faced....................................................................................... 37

* 1. Future Study......................................................................................................................... 37

**References**……….........................................................................................................................39

**List of Tables**

|  |  |  |
| --- | --- | --- |
| **Table No.** | **Description** | **Page** |
| 4.1 | Estimated Cost for complete development,  deployment and maintenance of the app for one year | 35 |

**List of Figures**

|  |  |  |
| --- | --- | --- |
| **Figure No.** | **Description** | **Page** |
| 1.1 | Project timeline | 3 |
| 3.1 | Data flow diagram of Authentication System | 5 |
| 3.2 | Data flow diagram of Offline Shopping Service | 6 |
| 3.3 | Data flow diagram of Online Shopping Service | 7 |
| 3.4 | Use case diagram for Customer | 9 |
| 3.5 | Use case diagram for Shopkeeper | 10 |
| 3.6 | System Architecture | 11 |
| 3.7 | Travelling salesman problem | 12 |
| 3.8 | Entity Relationship Diagram | 14 |
| 3.9 | Class Diagram | 16 |
| 3.10 | ASP.NET MVC Logo | 17 |
| 3.11 | Android Studio Logo | 18 |
| 3.12 | Geoapify Logo | 18 |
| 3.13 | SignalR Logo | 19 |
| 3.14 | Leaflet JS Logo | 20 |
| 3.15 | SmarterASP Logo | 20 |
| 4.1 | Different Views of Products | 24 |
| 4.2 | User Login & Registration | 26 |
| 4.3 | Product Searching | 27 |
| 4.4 | Showing products, shop location and other details | 27 |
| 4.5 | Rating & review system | 28 |
| 4.6 | Adding product to cart | 29 |
| 4.7 | Showing route | 29 |
| 4.8 | Shop Registration | 30 |
| 4.9 | Shopkeeper Actions | 31 |
| 4.10 | Online order system | 32 |
| 4.11 | Messaging between customer and shopkeeper | 33 |

**1 Introduction**

ShopiRound is an application that used users GPS to show him the products that are available in the shops that are nearby him. The core objective of this application is to allow user be aware of the shops around him, also if he ever need something immediately and he has to buy it himself, he can search for that product in this application rather that find it in every shop.

* 1. **Background**

Main objective of an e-commerce shop application provides a service of finding any product online and buy or sell it. It can be a website, desktop application or mobile application which work as a platform that a person can use to buy or sell goods. However, the majority of the e-commerce shop offers customer to place an order for a product which he wants to buy, and the service provider of the shop ship the ordered good to the customer. So, it takes time to get the product in hand which might be inconvenient for some customers. Many users might not trust those services as some of them ask for the price of the product before delivering it. ShopiRound is an application that can address these problems very well. It only let user know what are the product that are available in the shops in his area. In other word, if a user needs anything, he will instantly know if he can buy that product in a nearby shop.

* 1. **Objectives**

Some of the main objectives of our project are:

1. Help user find a product in local shop quickly.
2. Let a user know about all the products and shop available in his area.
3. Let user know about location of any shop or product.
4. Help sellers by boosting their sells.
5. Save time and making life a bit easier.
   1. **Contribution**

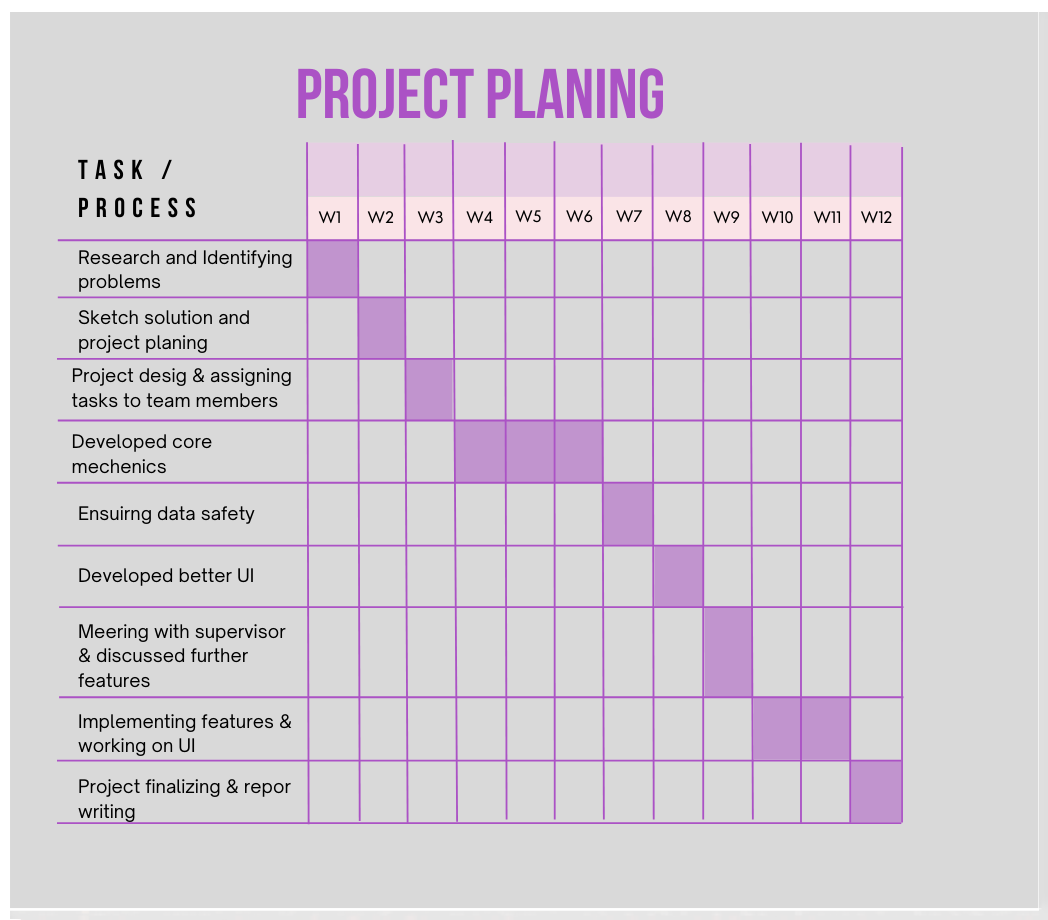
We, a team of two, has developed this application collaboratively throughout 3.2 semester. With some assists from our advisor, we planned, researched and came up with this amazing idea for our system project. During the development phase, we did collaboration, and version control using GitHub. One of the main features of our system is showing location product on map. A free geocoordinate service named Geoapify was a great service to our system.

**1.4 Unfamiliarity of the problem**

There are several online e-commerce shop applications but none of them provide such service that user can buy product locally instantly. People spend hours from shop to shop to find a product he wants. Sometimes he in a new area doesn’t know where are shops located. To address these issues our project comes in action. User can easily overcome these problems by navigating each and every product one by one. User can not only locate a product; he will also be able to find shops near him or in an area. Which will allow him to buy anything from specific local shop.

**1.5 Project Timeline:**

We did this project though out 3.2 semester. In this 13 week we conducted our research, design, development, deployment of this project segment by segment. A brief description of week-wise development is pictured below.



**Figure 1.1: Project timeline**

**2. Related Works**

**2.1 Existing Solutions**

**2.1.1 Daraz:** Daraz [1] is now one of the biggest online shopping applications in our country. We can find almost any product in here. A user can search, find product, place an order and Daraz will ship the ordered item to user destination after some days. User need to pay for the product plus some extra fees for the delivery.

**2.1.2 Rokomari.com:** Rokomari.com [2] is another biggest online book shopping platform in Bangladesh. It works in similar ways as Daraz. User need to play extra to get his desired good.

**2.1.3 Pickaboo:** Pickaboo [3] is another name in the list of online shopping application. It works the same way other online platform do.

**2.1.4 Chaldal:** Chaldal [4] is a popular online grocery store that helpuser by providingservice of delivering food and grocery at door step. User has to order and pay online to get the service.

**2.1.5 Ajkerdeal:** Ajkerdeal [5] is another big online shopping marketplace in Bangladesh. It’s a B2C online marketplace.

**2.2 Limitation in existing solution**

All these existing popular solutions only provide service about selling product online but none of them provide physical location of the product or a shop so that user could buy the product himself from the shop himself. Through our service, user will be able to do that. Not just know location of a product or a shop but also be aware of all the available shops available in his area. It saves user's time and save him from spending extra money for delivery. The users have full control over the application. Seller can show the user that product from his store costs less that another store. This way seller can boost his sells, at the same time users are also benefited for spending less money. Moreover, will a simple and easy to navigate and user friendly Ui user can easily master the application and start using for their own sake.

**3 System Design**

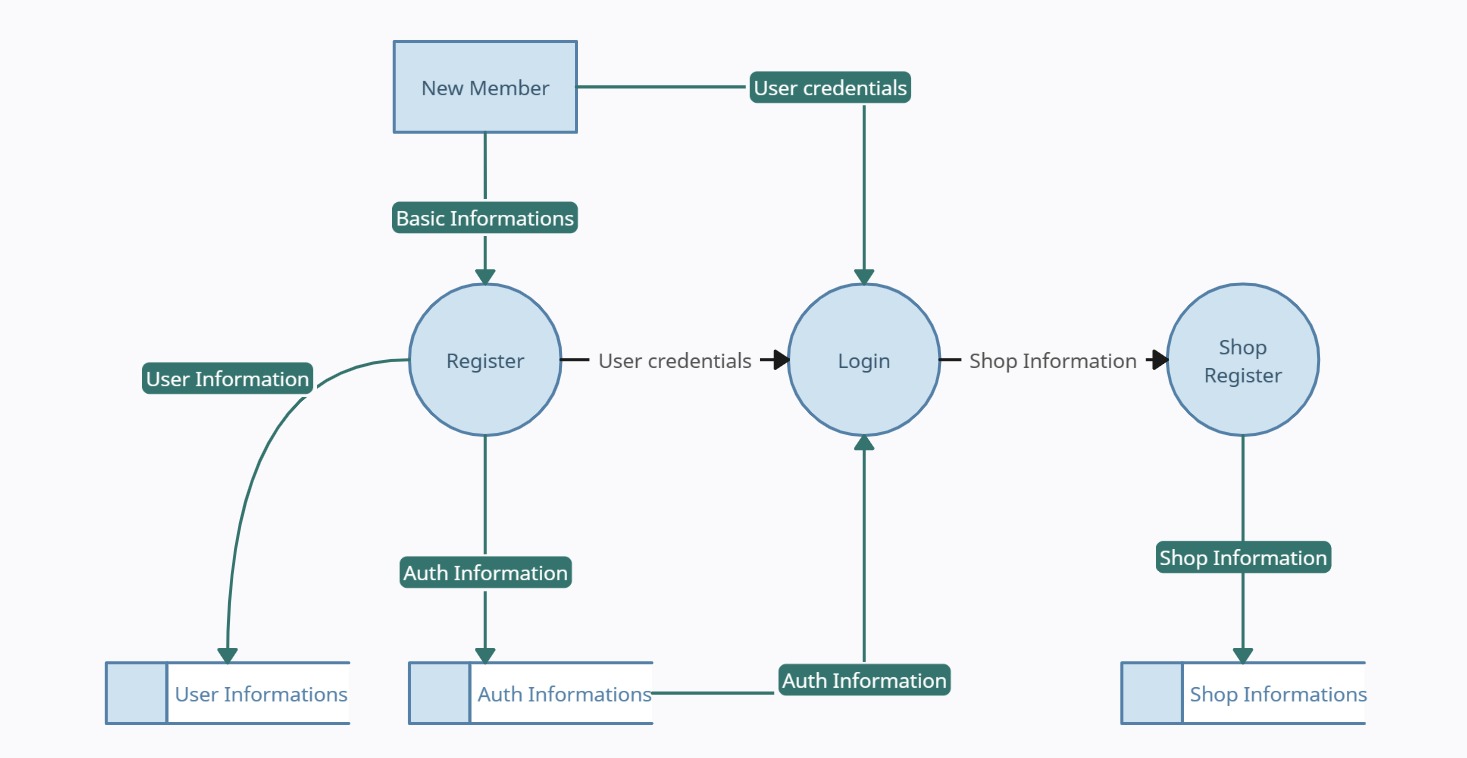
**3.1 Analysis of the system:**

System analysis for ShopiRound involves a comprehensive evaluation of its structure, functionality, and interactions to ensure effective operation of the system. We have discussed about Data Flow Diagram (DFD) and Use case Diagram of our system below.

**3.1.1 Data Flow Diagrams:**

In the context of system analysis, Data Flow Diagrams (DFDs) serve as a vital tool for visualizing and depicting the flow of data within a system. These diagrams aid in comprehending the interactions between various components and illustrate how data moves through the system.

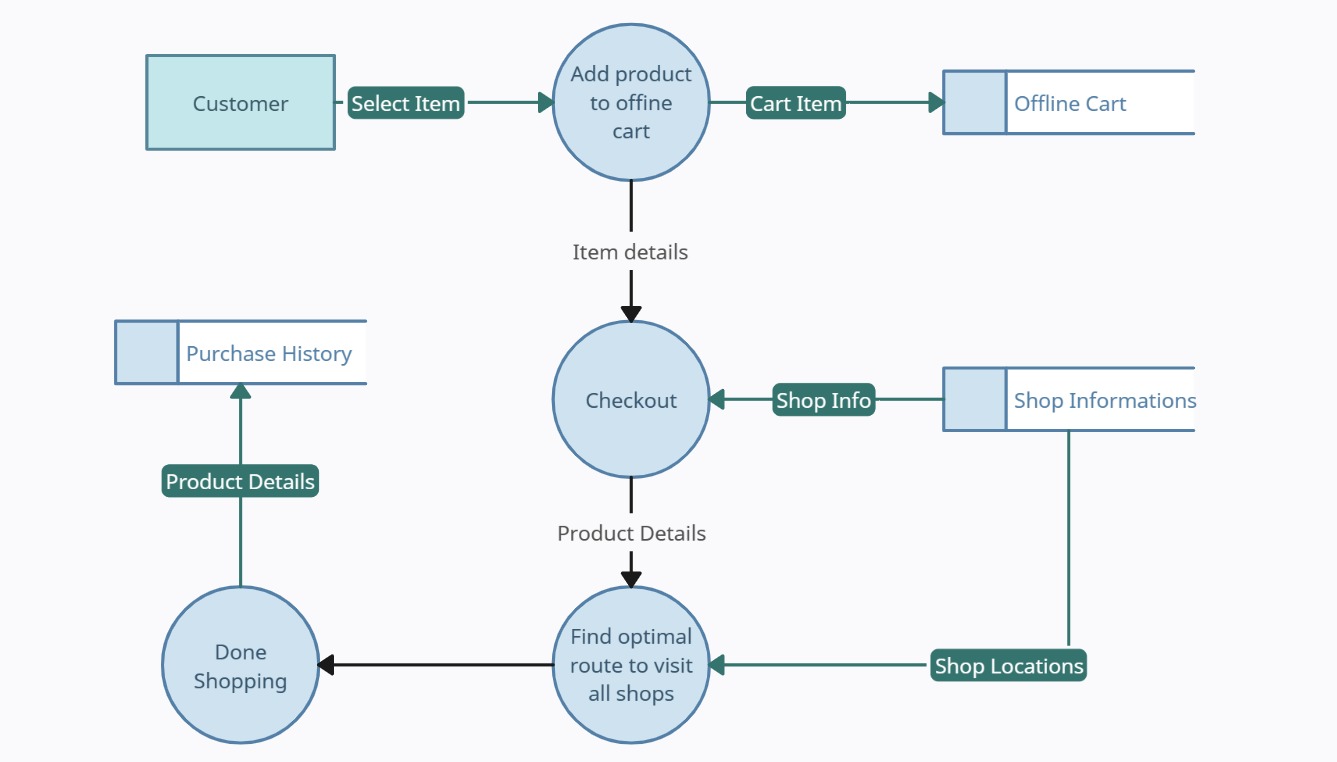
**3.1.1.1 Authentication:**



**Figure 3.1:** Data flow diagram of Authentication System

The figure 3.1 above shows the flow of data during user registration, login and shop registration. When a new user wants to buy any products from our website, he will have to register for an account. Through register process, he will provide necessary information and those will be stored in User Information table and Auth information table. Right after register, he will be automatically logged into the system. In future, while user wants to login, he can give the same user credentials which will be checked through Auth system. A logged in user can register for a shop by giving required information to the Shop Register class and it will be stored in Shop Information table.

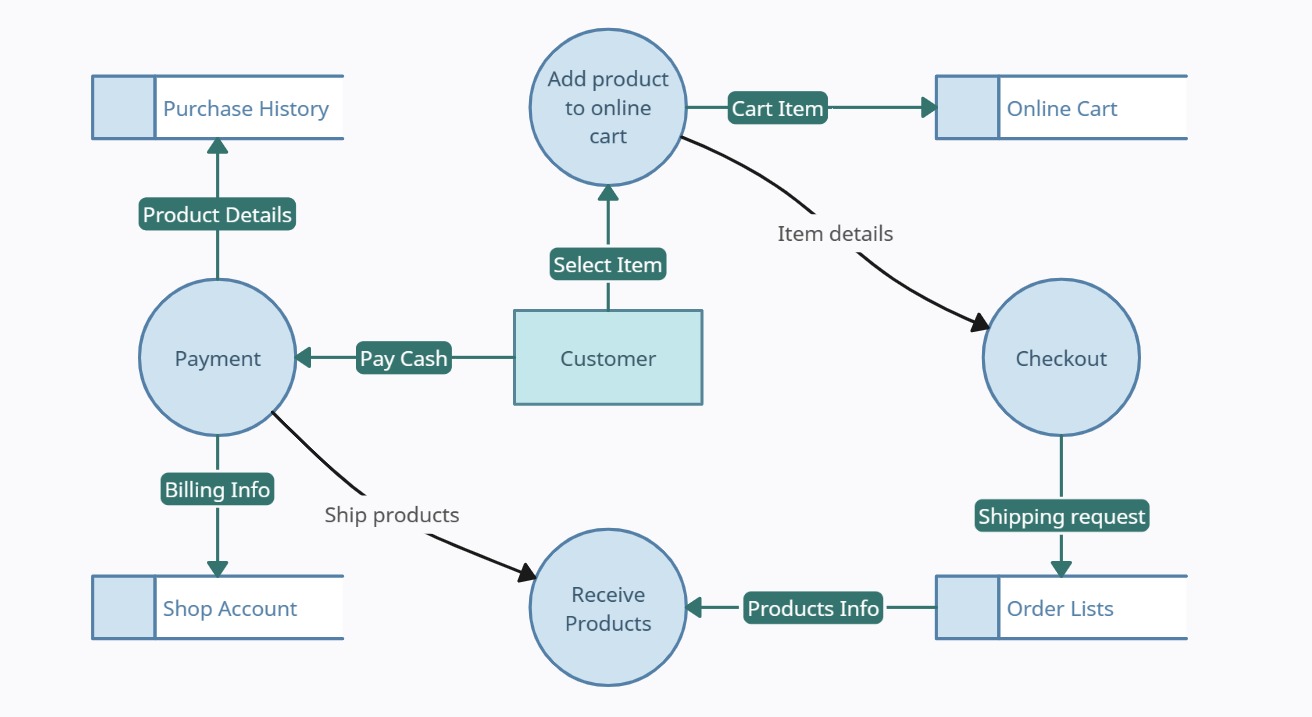
**3.1.1.2 Offline Shopping:**

****

**Figure 3.2:** Data flow diagram of Offline Shopping Service

The above diagram shows the flow of data during buying a product offline. Customer has to add the product to offline cart. This cart item will be stored in Offline Cart table. After adding all the products, he can view all the cart items from offline cart table and proceed to check out. In checkout, shop information will be fetched and show its location on map with optimal route from user to the shop. There can be multiple shops from where customer wants to buy, in that case all information of all shops will be shown one by one with route connecting all of them. When customer done his shopping, a copy of all the information of the products he purchased will be saved in a Purchase History table.

**3.1.1.3 Online Shopping:**



**Figure 3.3:** Dataflow Diagram of Online Shopping Service

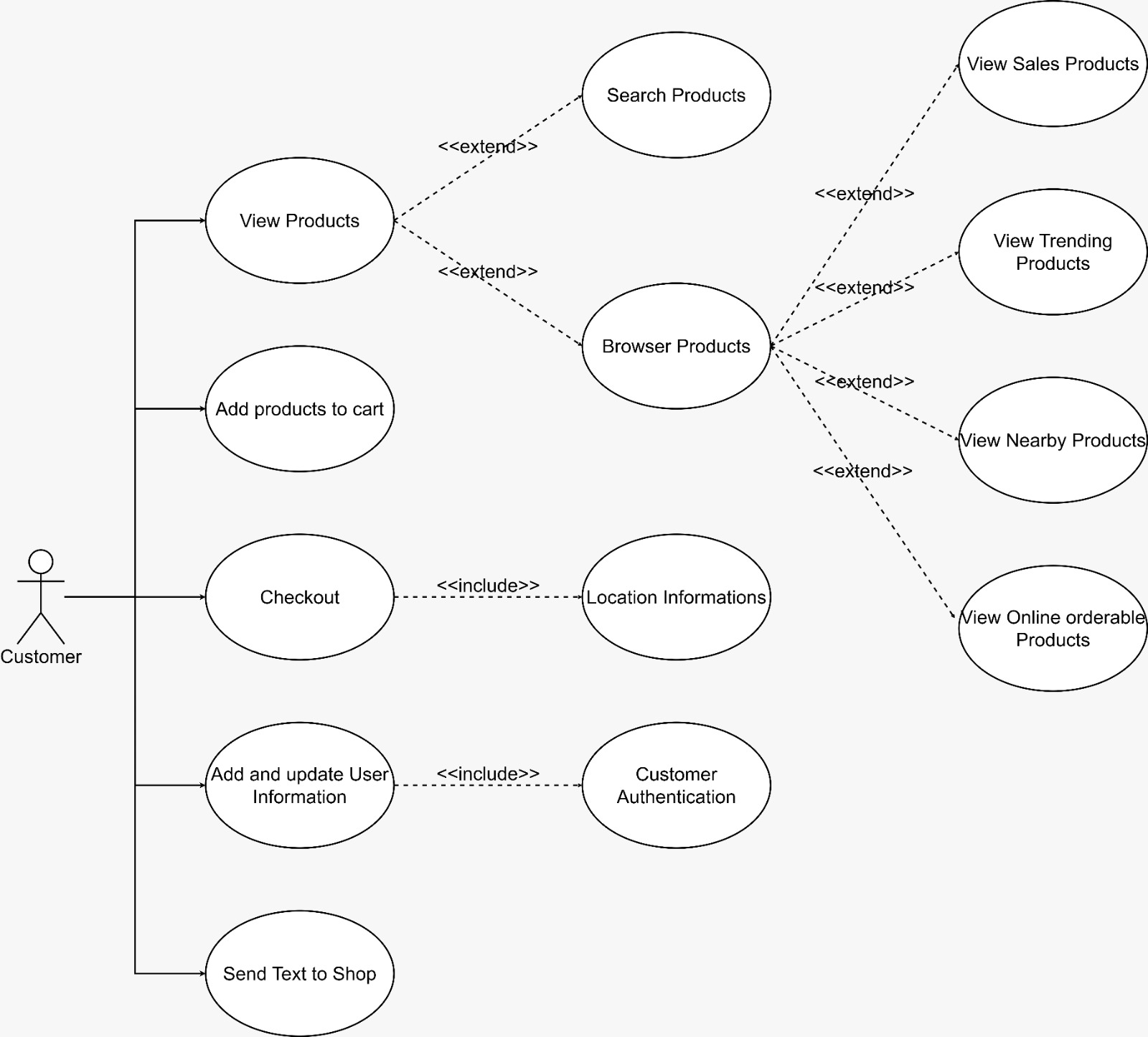
In this diagram, we showed the data flow during an online order of product. Customer can add product to online cart. After adding all the products, he will checkout, and the information about those products will to move to Order List and Shopkeeper has access to this list. Now shop owner can deliver those products. After receiving the goods, customer will complete payment and Billing information will be stored in Shop Account and the information about those goods will be stored in Purchase History of customer.

**3.1.2 Use case Diagrams:**

Use case diagram of ShopiRound showcases the functional requirements and interactions between users, shopkeepers, and the system itself. It illustrates various use cases or scenarios depicting how actors (users, shopkeepers) interact with the system to accomplish specific tasks or functionalities. Here we are showing use cases form user or customer and shop owner is shown.

**3.1.2.1 Use case Diagram from Customer’s Perspective:**

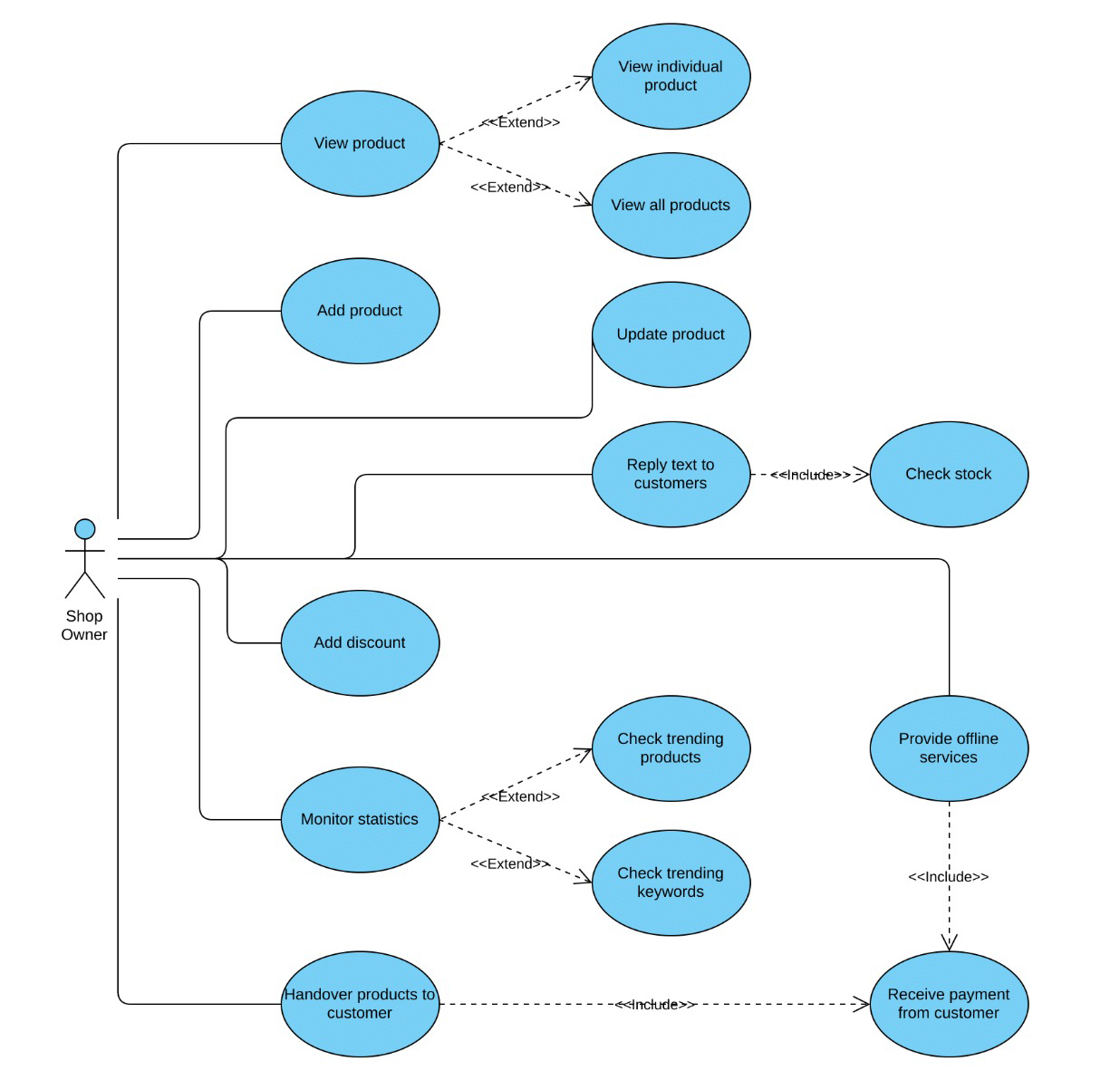
The diagram above shows the use cases of a customer. It shows what are the capabilities we gave to customer inside our system. Customer can search for a product and view it. There is other several options to brows product like viewing the Trending Products that shows the products that are in trending list currently. Customer can see Nearby Products, that are the products near to user. Customer can view Online Products if any shop provides online services. Customer can add the products to cart that he wants to buy. After adding all products to his like, he can checkout, which will assist him to go to the shops where he will buy the products. This assistance will be done by showing location of the shops on a map will proper route to go there. We have also given customer the capability sends direct message to shopkeeper for enquiry. Customer can also update his profile information.



**Figure 3.4:** Use case diagram for Customer

**3.1.2.2 Use case Diagram from Shop Owner’s Perspective:**

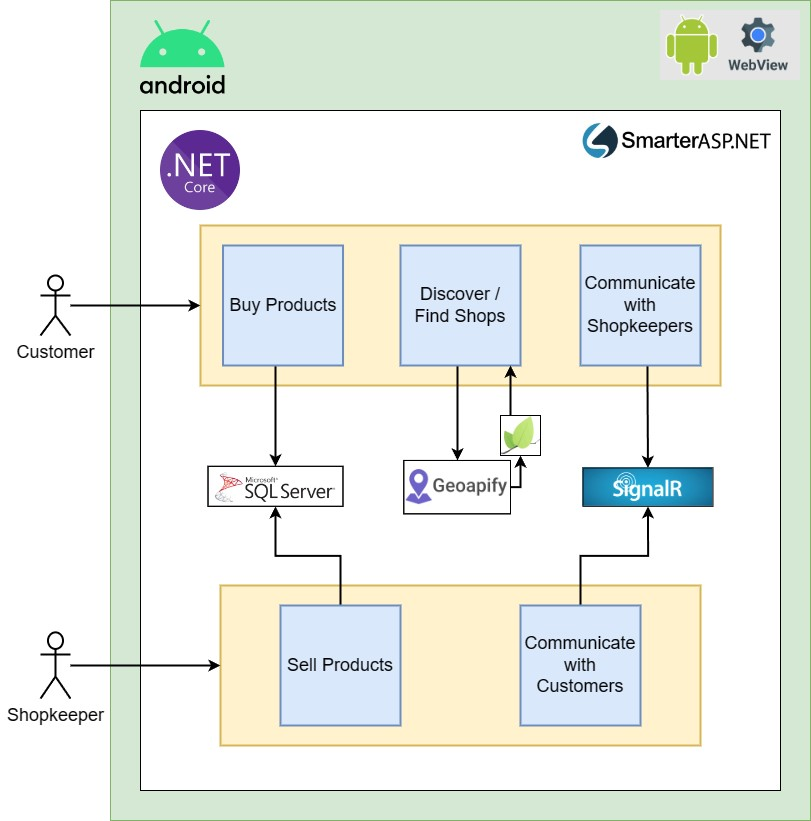
Here we are showing the use case of a shop owner. A shopkeeper’s activity is very different from a customer. As a normal customer, he can see any product and buy for himself. But additionally, he can add product, update or delete product. He can reply to the message sent by customers. He can add discount to specific product of his to increase publicity of his products as they will be shown in a separate page. He can receive online orders and deliver them.



**Figure 3.5:** Use case diagram for Shopkeeper

**3.2 System Architecture:**

System Architecture of ShopiRound is the conceptual model that defines the structure, behavior and more views of the system. It’s a formal description and representation of the system that can consist of system components and sub-system developed and that will work together that will work together to implement the system.

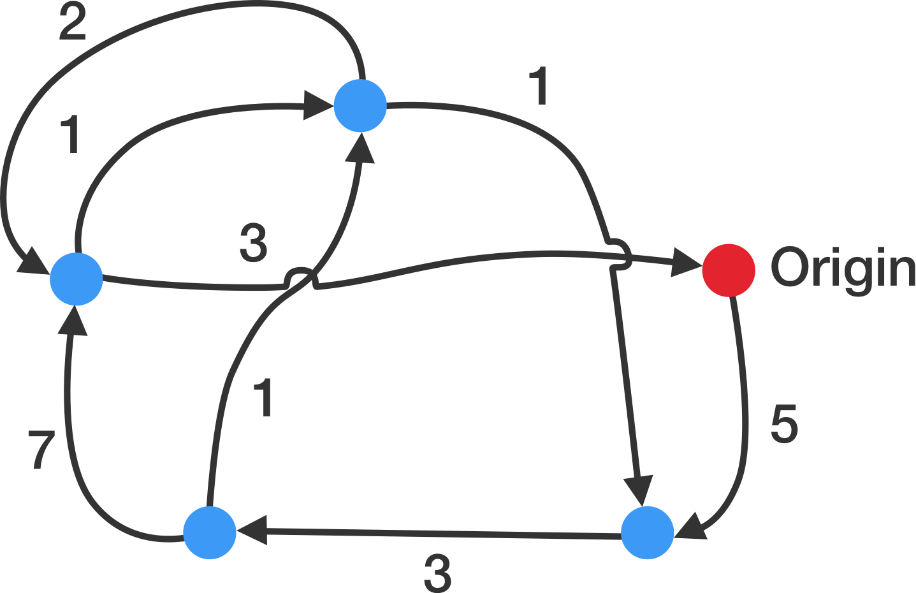


**Figure 3.6:** System Architecture

In the figure above, we have indicated various features for Customer and Shopkeeper and show the components that works together to implement those features. Customer can buy products and Shopkeeper can sell products. In both operations data of products is being created, updated or deleted through SQL Server. Customer can find nearby shops in our application. Retrieving geo coordinates of shops and calculating the distance is done by using Geoapify[7] Api, and the location of shops is shown on map using Leaflet Js. Customer and Shopkeeper can establish communication over a product using SignalR[8]. This whole system is implemented using ASP.Net [6] Core. We hosted this website in SmarterAsp.net [10] to have remote access to our system and an android application is made which works as a browser that only shows our website. We used web-view in android to implement this feature to be specific.

**3.2.1 Routing Algorithm (TSP):**

To enhance the efficiency of delivering products from multiple shops to a customer's home, we employ the Traveling Salesman Problem (TSP) approach. Initially, we compile a list of shops where the customer has items in their cart, and then retrieve location information for each shop from our database. Utilizing the Geoapify[7] map routing API, we generate a route matrix, establishing optimal routes between all the shops. The TSP algorithm is then applied to this matrix, strategically determining the most efficient route considering the customer's home and all the shops involved. This meticulous planning aims to minimize the overall distance traveled, optimizing the delivery journey in terms of both time and cost.



**Figure 3.7:** Travelling salesman problem

**Travelling Salesman Problem**

**(TSP)Algorithm**: Given a set of cities and distances between every pair of cities, the problem is to find the shortest possible route that visits every city exactly once and returns to the starting point.

Let the given set of vertices be {1, 2, 3, 4,….n}. Let us consider 1 as starting and ending point of output. For every other vertex I (other than 1), we find the minimum cost path with 1 as the starting point, I as the ending point, and all vertices appearing exactly once. Let the cost of this path cost (i), and the cost of the corresponding Cycle would cost (i) + dist(i, 1) where dist(i, 1) is the distance from I to 1. Finally, we return the minimum of all [cost(i) + dist(i, 1)] values.

Now to calculate the cost(i) using Dynamic Programming, we need to have some recursive relation in terms of sub-problems.

Let C(S, i) be the cost of the minimum cost path visiting each vertex in set S exactly once, starting at 1 and ending at i. We start with all subsets of size 2 and calculate C(S, i) for all subsets where S is the subset, then we calculate C(S, i) for all subsets S of size 3 and so on. Note that 1 must be present in every subset.

**Pseudocode:**

If size of S is 2, then S must be {1, i},

C(S, i) = dist(1, i)

Else if size of S is greater than 2.

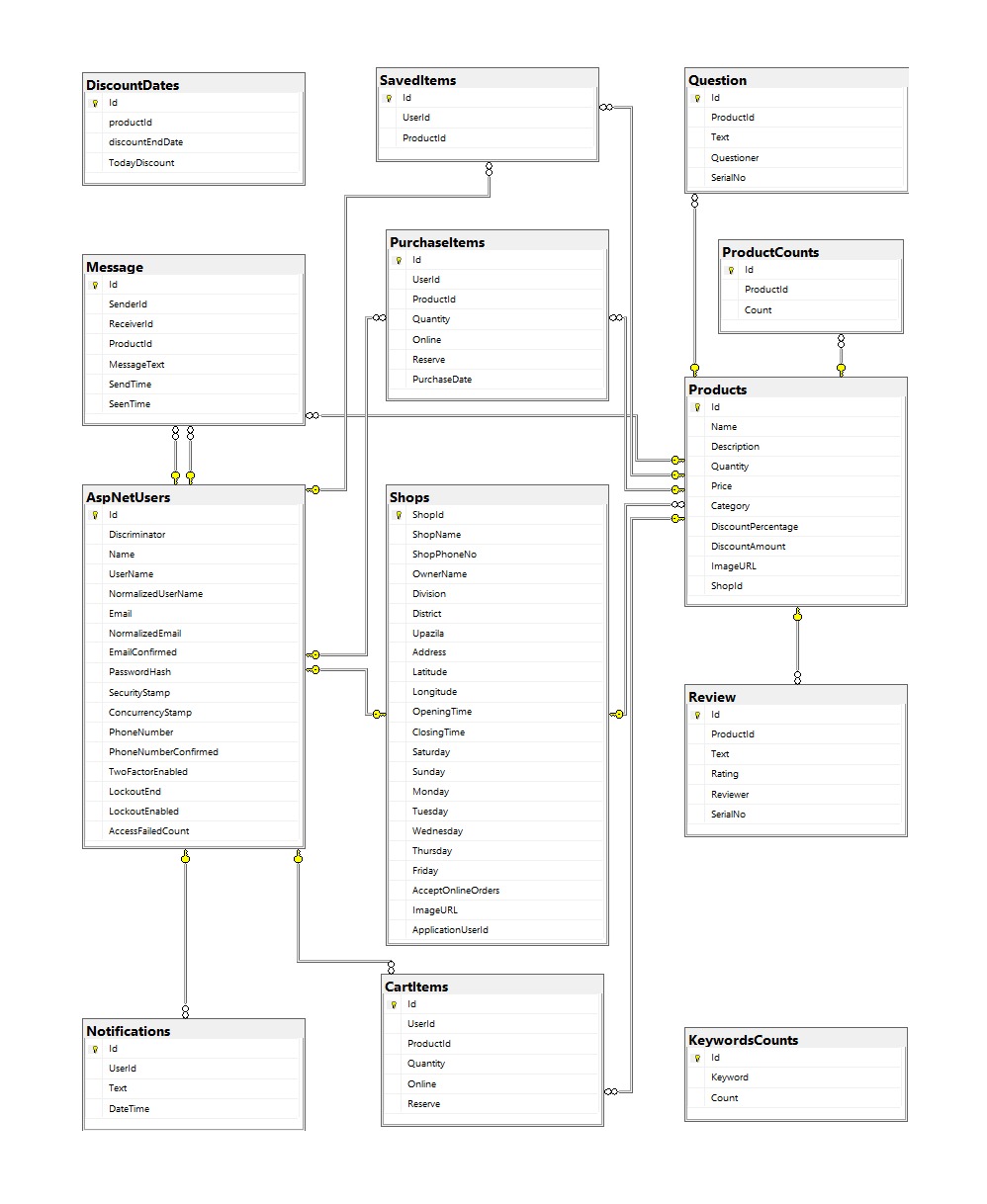
C(S, i) = min { C(S-{i}, j) + dis(j, i)} where j belongs to S, j != i and j != 1.

**Time Complexity: O(n2\*2n)**where O(n\* 2n)are maximum number of unique subproblems/states and O(n) for transition (through for loop as in code) in every states.

**Auxiliary Space: O(n\*2n),**where n is number of Nodes/Cities here.

**3.2.2 Database Design:**

Database design shows the overall architecture of all the data table used in the system and relations among them.

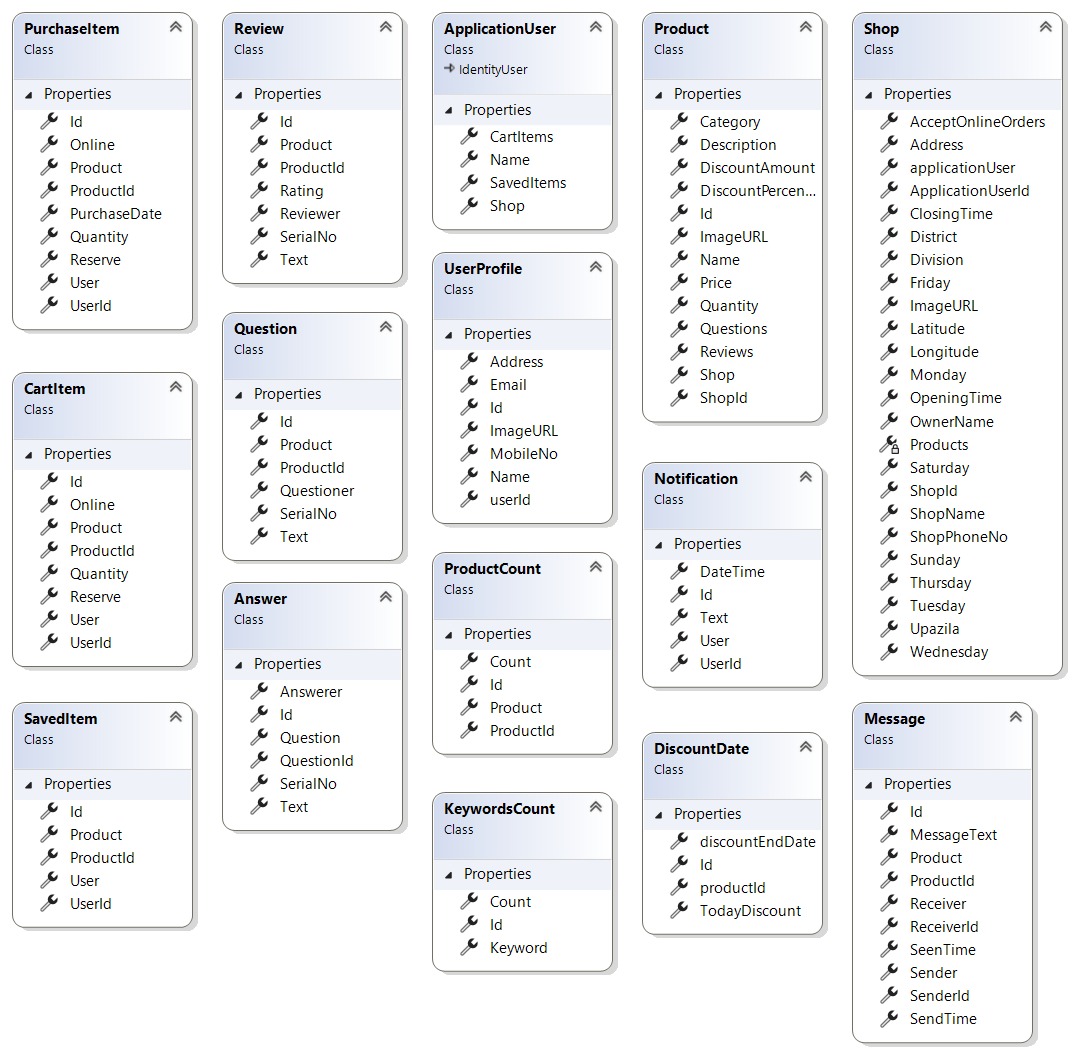
****

**Figure 3.8:** Entity Relationship Diagram

In the diagram above, we are showing the database design of our system. Starting with AspNetUsers table, containing all the information about a user like Name, Email, Password etc. of our system. Any user can register shop by providing additional shop information. That information is stored in Shops table. Among the information some of them are ShopName, OwnerName, ShopPhoneNo, OpeningTime, ClosingTime, OpeningDays and Latitude Longitude which are geo co-ordinates to locate shop location. The Opening days are separated into seven days to store yes or no, meaning if the shop is open on that day or not. There is a Notification table, that is essentially a table to store notification data for a user. There is a Message table, which stores messages between two people. The message table contains ProductId, because customer can message under a specific product, SenderId, ReceiverId and the TexMessage. CartItem table is for storing data about the product that are added to cart buy customer. It contains, UserId , ProductId, the added product, Quantity and Online which indicates if the product is added to online cart or the offline cart. Saved table is used to contain data about those products user saved. It contains ProductId and UserId. PurchasedItems table is used to store the products data that are been bought by the customer. Product table is used to store data about specific product. It contains

**3.2.3 Class Diagram:**

Class diagram shows all the classes we used in our project. They were use to manipulate data in table. So, they are similar to those table we discussed earlier. In shopiRound we used a complex database architecture to ensure proper implementation of our system.

****

**Figure 3.9:** Class Diagram

**3.3 Tools**

In this section we have discussed about the tools we used during the development of our project.

**3.3.1 ASP.Net MVC:**

ASP.NET MVC [6] (Model-View-Controller) is a web development framework designed by Microsoft, offering a structured approach to building dynamic web applications. It separates the application into three interconnected components: the Model (data), the View (user interface), and the Controller (logic and interactions). This architecture promotes clean code, testability, and easier maintenance. ASP.NET MVC[6] supports rapid development through features like routing, which enables clean URLs, and robust data-binding capabilities. It empowers developers with powerful tools and libraries, facilitating seamless integration with other Microsoft technologies. The framework's extensibility allows for customization and scalability, catering to diverse project requirements. Its adherence to the principles of separation of concerns enhances code modularity and promotes agile development practices. ASP.NET MVC[6]

remains a popular choice for creating scalable and responsive web applications in the .NET ecosystem.



**Figure 3.10:** ASP.NET MVC Logo

**3.3.2 Android Studio:**

Android Studio[9] is the official Integrated Development Environment (IDE) for Android app development. Developed by Google, it provides a comprehensive set of tools and features to streamline the creation of Android applications. With an intuitive interface, it offers code editing, debugging, and testing functionalities within a single platform. Android Studio[9] supports various programming languages like Java, Kotlin, and C++, enabling developers to create diverse and innovative apps for the Android ecosystem. Its built-in e emulators allow real-time previews of apps across multiple devices, facilitating optimal user experience testing. As a preferred choice for Android developers, Android Studio[9] continues to evolve with frequent updates and improvements.



**Figure 3.11:** Android Studio Logo

**3.3.3 Geoapify:**

Geoapify[7] is a versatile geolocation platform offering a suite of APIs and tools designed to facilitate location-based services. It provides developers with comprehensive geocoding, mapping, routing, and location intelligence APIs, allowing seamless integration of location data into applications. With its geocoding API, Geoapify[7] enables accurate address conversion into geographic coordinates and vice versa. The platform's mapping API offers customizable maps with various layers and styles, empowering developers to create visually engaging mapping solutions. Geoapify's routing API provides efficient directions and routing functionalities, optimizing travel routes. Additionally, its geolocation and geofencing capabilities offer precise location tracking and boundary-based triggers, enhancing location-based experiences for applications across diverse industries. The platform's scalable infrastructure and user-friendly documentation make Geoapify[7] a valuable asset for developers seeking robust location-based services integration



**Figure 3.12:** Geoeapify Logo

**3.3.4 SignalR:**

SignalR[8] is a real-time library by Microsoft that simplifies building web applications with live, bi-directional communication. It enables seamless WebSocket-based communication between the server and client, facilitating instant updates and interactions. With its ability to handle various transport mechanisms, SignalR[8] ensures wide compatibility across browsers and devices. This library supports persistent connections, allowing servers to push content instantly to connected clients, making it ideal for implementing chat functionality, live updates, and real-time notifications in web applications. SignalR's[8] intuitive APIs and robust features make it a preferred choice for developers seeking efficient real-time communication solutions.



**Figure 3.13:** SignalR Logo

**3.3.5 Leaflet Js:**

Leaflet.js is an open-source JavaScript library widely used for interactive mapping in web applications. It offers a lightweight and versatile platform for creating customizable, mobile-friendly maps. With a simple and user-friendly API, Leaflet enables developers to integrate various map layers, markers, and interactive features effortlessly. It supports multiple tile providers, allowing easy integration of maps from different sources. Leaflet's modular design and extensive plugin ecosystem make it a popular choice for building responsive and interactive mapping solutions across various web platforms.



**Figure 3.14:** Leaflet JS Logo

**3.3.6 SmarterASP:**

SmarterASP[10] is a web hosting service renowned for its reliability and user-friendly solutions. It offers a range of hosting plans tailored to various needs, including shared hosting, VPS hosting, and dedicated servers. With robust infrastructure and 24/7 customer support, SmarterASp[10] ensures high uptime and dependable performance for websites and applications. Its intuitive control panel simplifies server management, allowing users to deploy and manage their web projects efficiently. SmarterAsp's[10] scalable hosting options and competitive pricing make it a favored choice among individuals and businesses seeking reliable web hosting services.

****

**Figure 3.15:** SmarterASP Logo

**4 Project Implementation**

**4.1 System Implementation:**

Over the past 12 weeks we have implements our system. During implementation we came across different challenges like what to build, how to build, which API to use, which tools to use. But we overcome all those challenges with the help of our supervisor and successfully implemented the system perfectly. Here are some of the stages, features and tools we used for our system, mentioned below.

**4.1.1 Planning and Preparation:**

* **Initial Research and Analysis:**
  + We did some researches and identified that there is a need for a platform that simplify our shopping by locating nearby shops and their available products.
  + We did market research to understand user requirement and performance.
* **Product Scope and Reequipments:**
  + We defined scopes and features for our application such as map integration, product display, distance calculation, and chat functionality with shopkeepers.

**4.1.2 Technology Stack and Infrastructure:**

* **ASP.NET MVC Development:**
  + We decided to develop our application using ASP.NET MVC[6] framework as it can offer scalability and robust architecture for our application easily.
  + We used SQL Server for data storage and management. We used it to handle product details, shop information, user interaction.
* **Hosting and Deployment:**
  + We hosted our website on SmartAsp, ensuring reliable and secure deployment.
  + Set up the necessary configuration for seamless operation.
* **Android App Development:**
  + As our advisor suggested, we developed an android app that works as a browser specifically for our website ShopiRound offering mobile friendly interface.
  + We ensured compatibility with various Android devices.
* **Integration of Map and Location Services:**
  + We integrated Geoapify[7] for geocoding, obtaining coordinates of users and shop locations.
  + Incorporated Leaflet.js for displaying maps, enabling users to view nearby shops and their locations.
* **Real-time Chat Functionality:**
  + Implemented SignalR[8] for real-time communication between users and shopkeepers, facilitating inquiries and assistance.

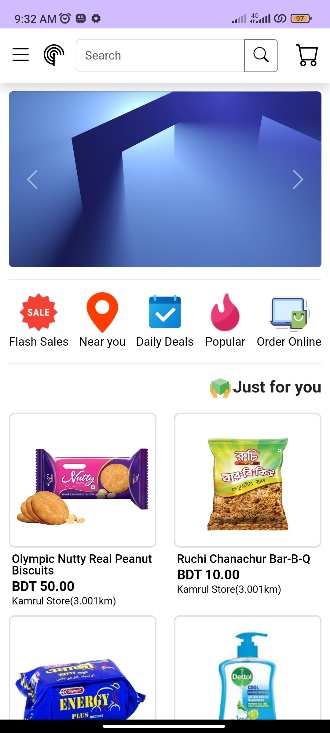
**4.1.3 Implementation of Key Features:**

* **User Interface and Experience:**
  + We have designed and implemented interface for our application in a very user-friendly way so that user can have a good experience. We followed essential UI and UX principles during making interface for both our website and android.
* **Location-based Search:**
  + We developed our search functionality that shows the products which are closer to user location first. It is done to make shopping a bit ease for the user.
* **Product Display and Search Functionality:**
  + User can have a good look at any product, read its description for inspection. User can search for specific product and find its location to buy it.
* **Chat System with Shopkeeper:**
  + User can negotiate with shopkeeper through our application. It can be used as real-life bargaining over a product. Also, it is a good way to establish a good connection between user and his nearby shopkeeper.

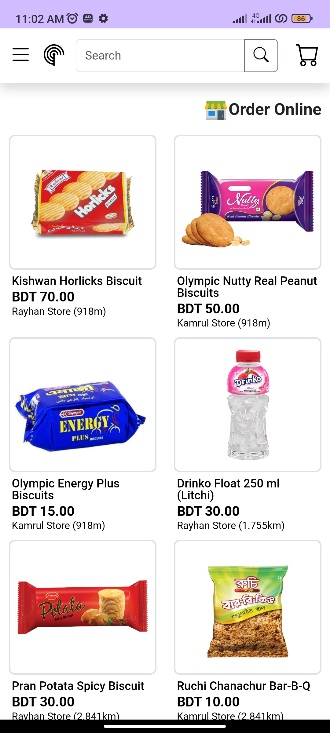
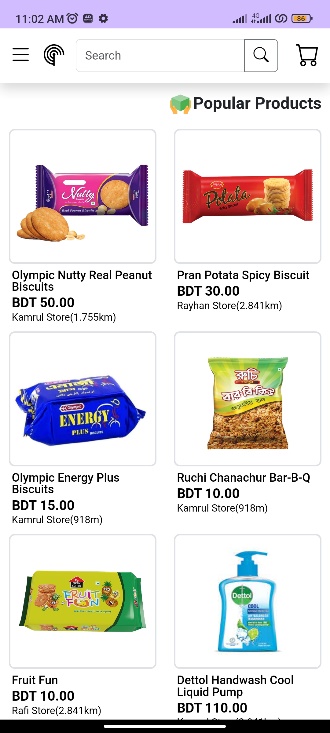
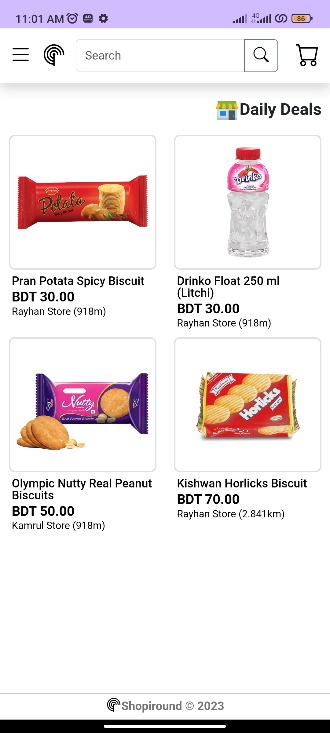
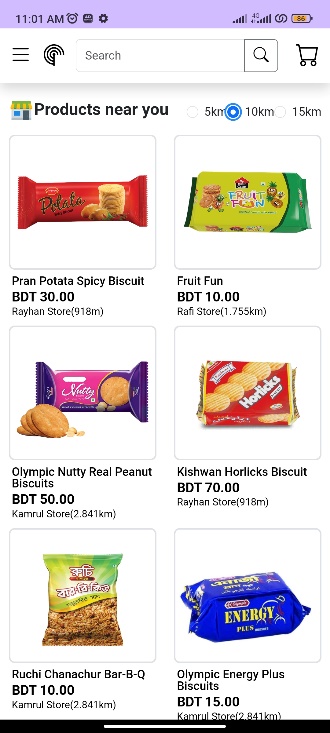
**4.2 User Manual:**

In this section, we discuss how different features of this can be accessed and how the users interact with the system.

**Home Page:**

****

Home Page



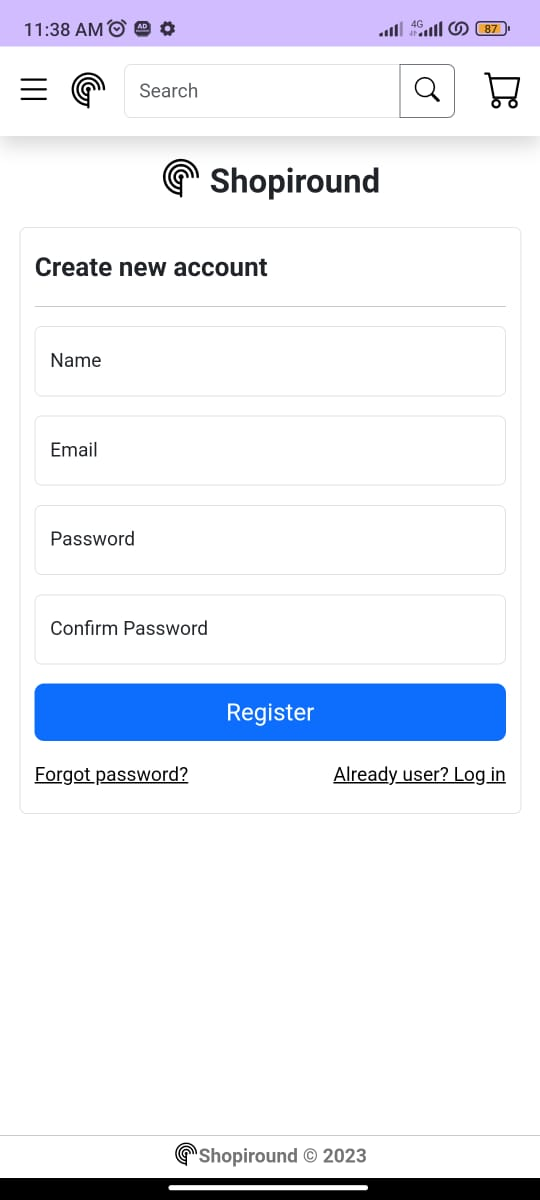
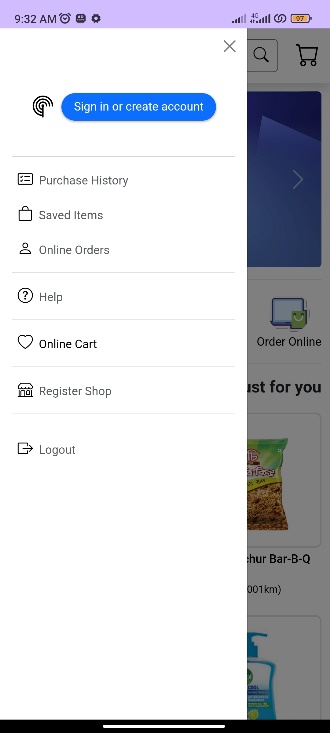
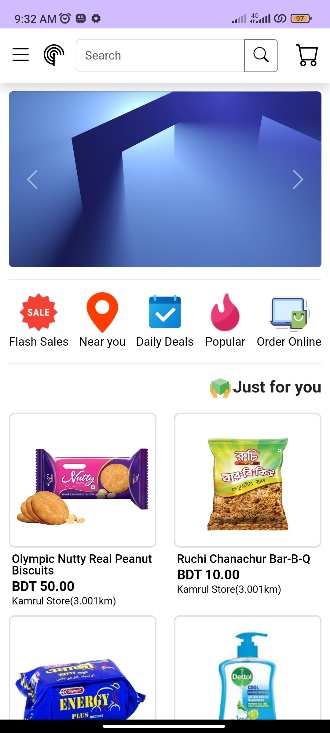
Near You page Daily Deal Page Popular Page Online Order Page

**Figure 4.1:** Different Views of Products

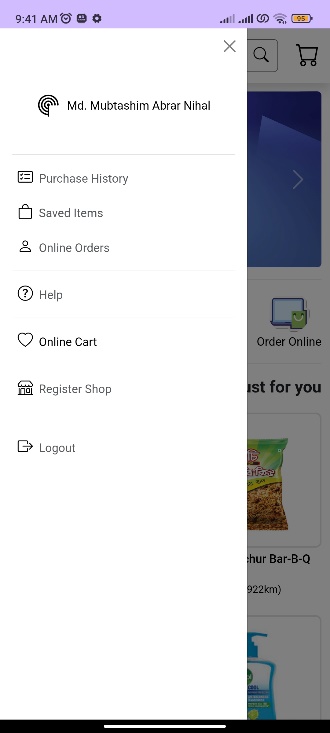
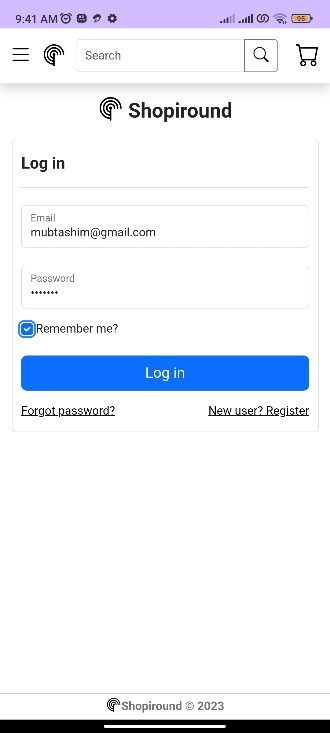
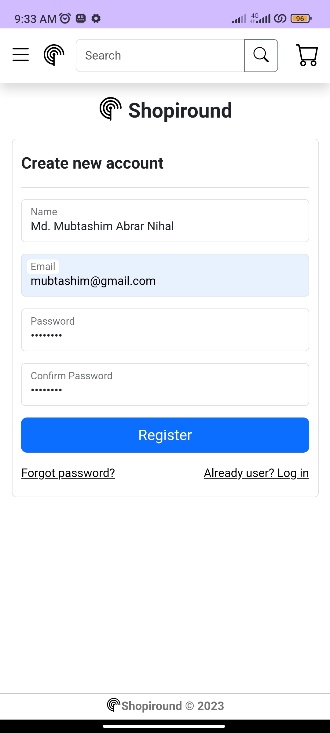
After visiting our website, user will be taken to home page where he can find all the products in his area with the shop name and distance of the product from user, a search bar where he can search a specific product and some other options that can assist him in his shopping, Like “Flash Sales”, to show the products that are in discount for a time being, “Daily Deals”, to show the products that are discounted only for today, “Popular Page” that shows the popular products based on other users search history, “Online Order Page”, that shows the products that the user can buy online.

**User Login and Registration:**

From top left navigation bar, user can sign in or register for account. He will have to provide name, email and password. After registering, he will be loggin to the system and directed to the home page. Now he can buy products.



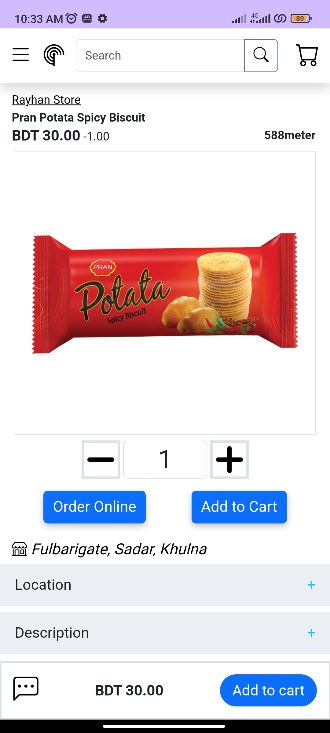
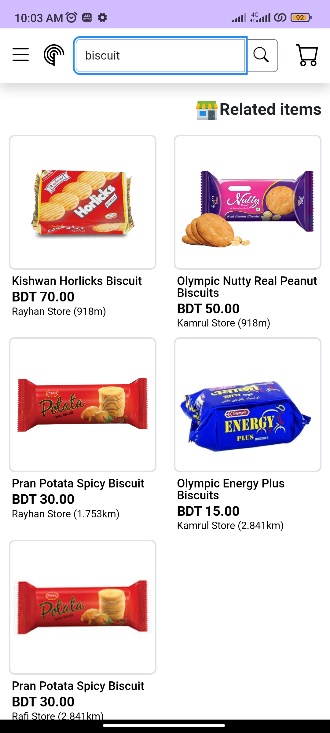
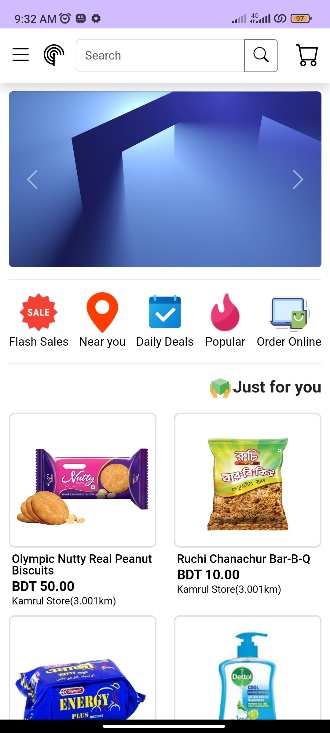
Home Page Side Navigation Bar Registration Page



**Figure 4.2:** User Login & Registration

**Search and Show Products:**

In home page, in the top search bar, user can search for a product and products from all the nearby shops will be shown. Our products are shown in an order that closer product is shown first than the other. From the search result, user can click an product to view detail about the product, like how far is it, name of the shop, ratings, and he can add that product to cart to navigate to that shop.

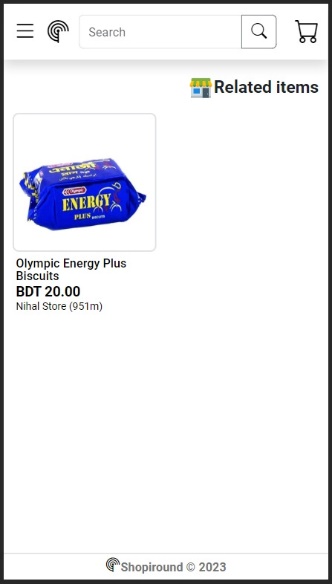
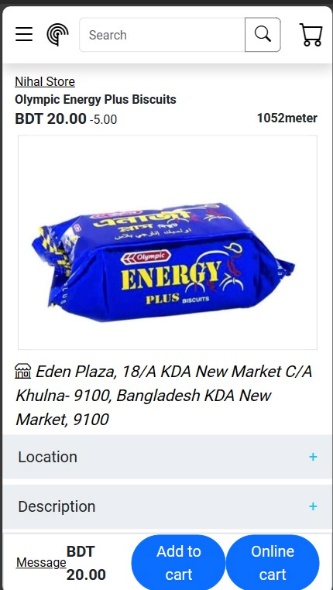
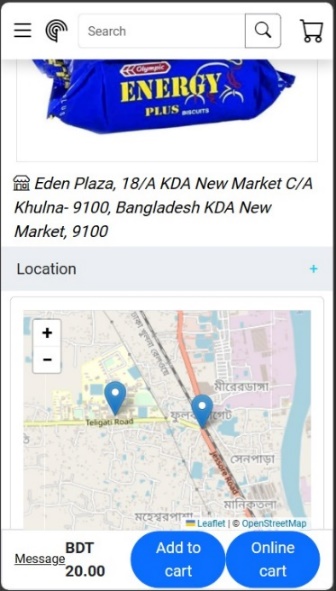


Home Page Search Result Show Product

**Figure 4.3:** Product Searching

**Product Show:**

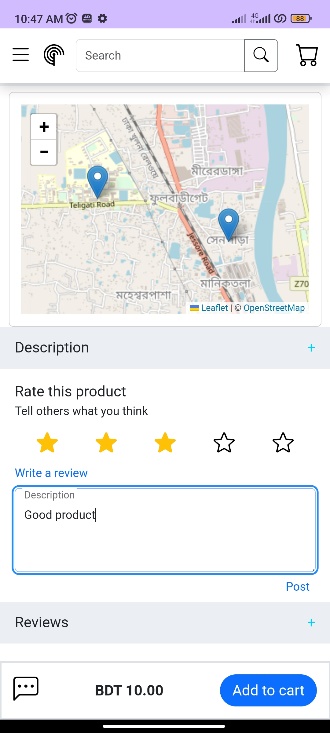
The interface the view all the details about a shop.

**** **** 

**Figure 4.4:** Showing products, shop location and other details

User can click on a product to see details about the product. Here he can observe exact location of the product. Here user can have the location of the shop on a map, if he presses the plus icon beside “Location”.

**Product Rating & Review:**



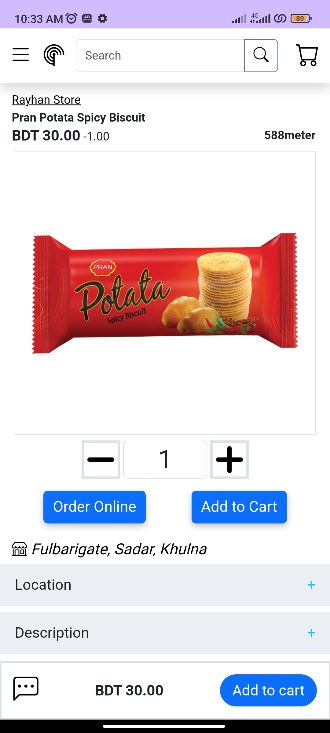
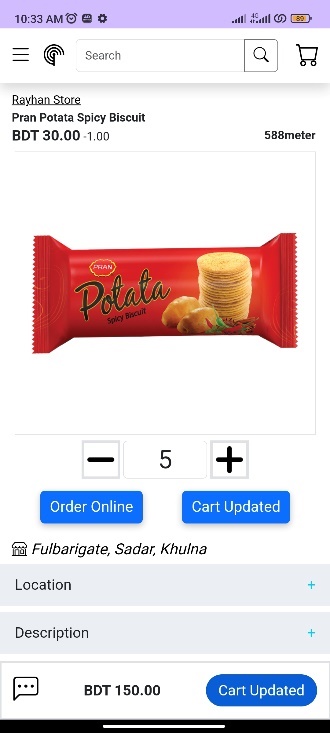
**Figure 4.5:** Rating & review system

User can view rating about the product given by other users previously bought this product from that shop. This way user can have a good overview how the product will be before buying it. He can also add rating himself.

**Add to Cart:**

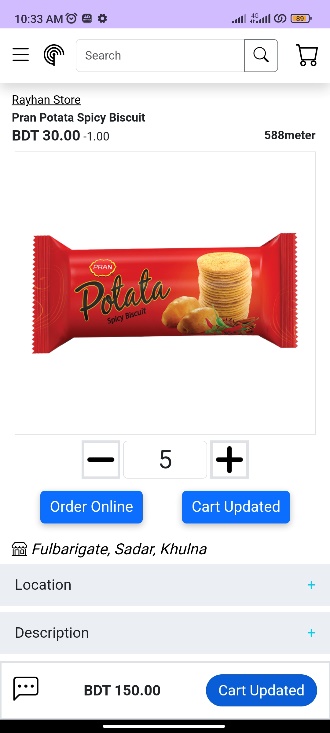
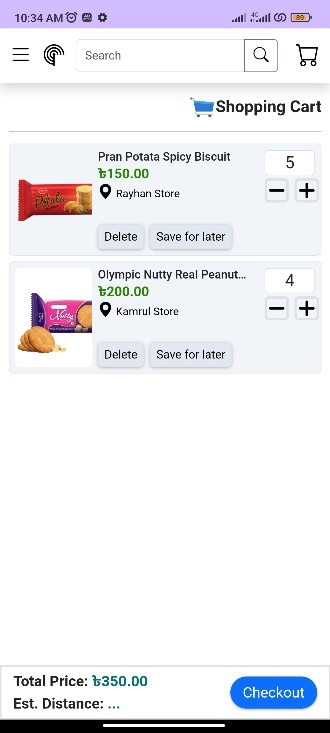
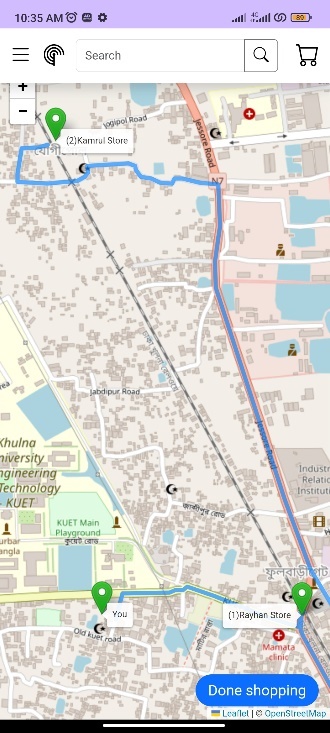
User can add product to cart. Cart is used to keep track of single or multiple product he wants to buy. He can also define the quantity of the product.



**Figure 4.6:** Adding Product to Cart

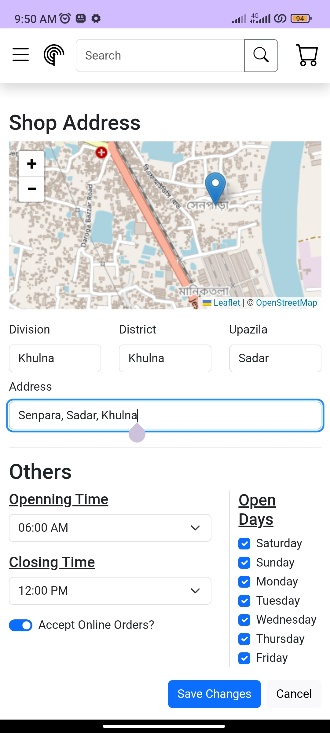
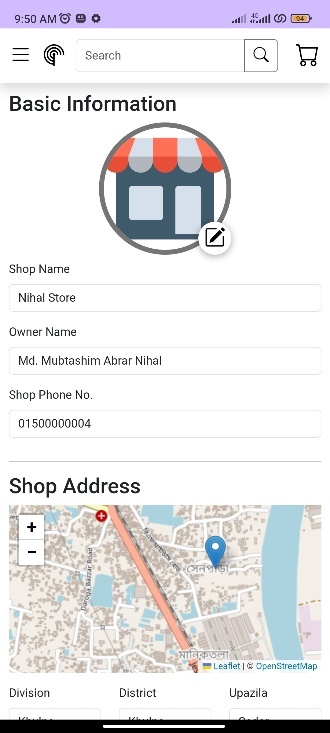
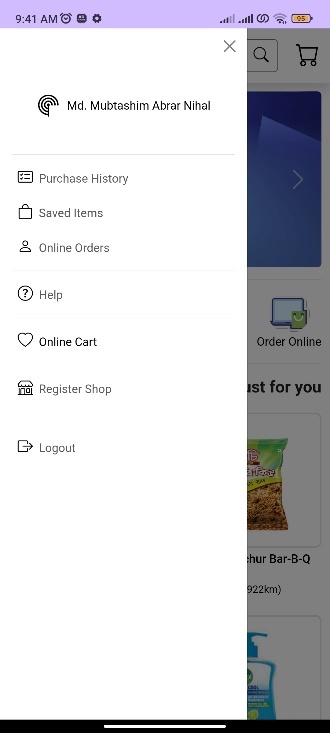
**Cart Checkout & Show Route:**

**Figure 4.7:** Showing Route

After adding all the products user wants to buy, he can click the cart icon in the head bar, it will show him all the products he added. Now he can click “Checkout” to show all the products on map with proper route. It will help user to go to the shop one by one to buy goods.

**Registering Shop:**

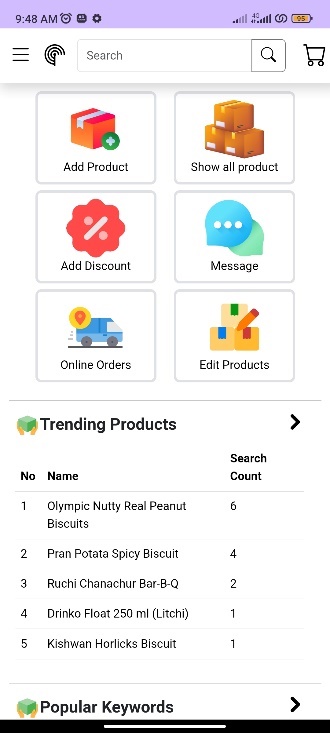
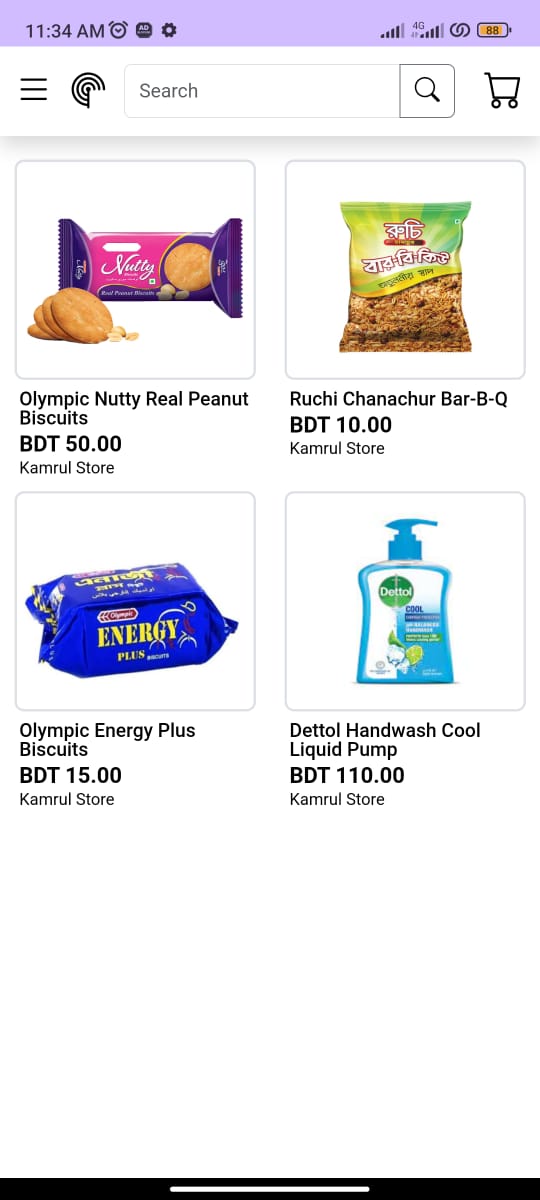


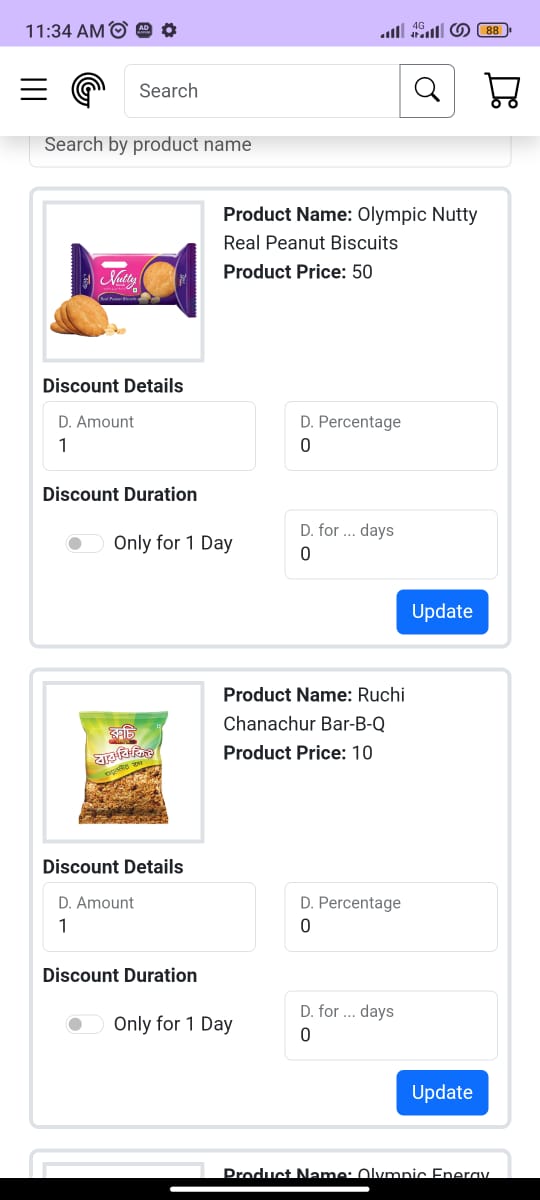
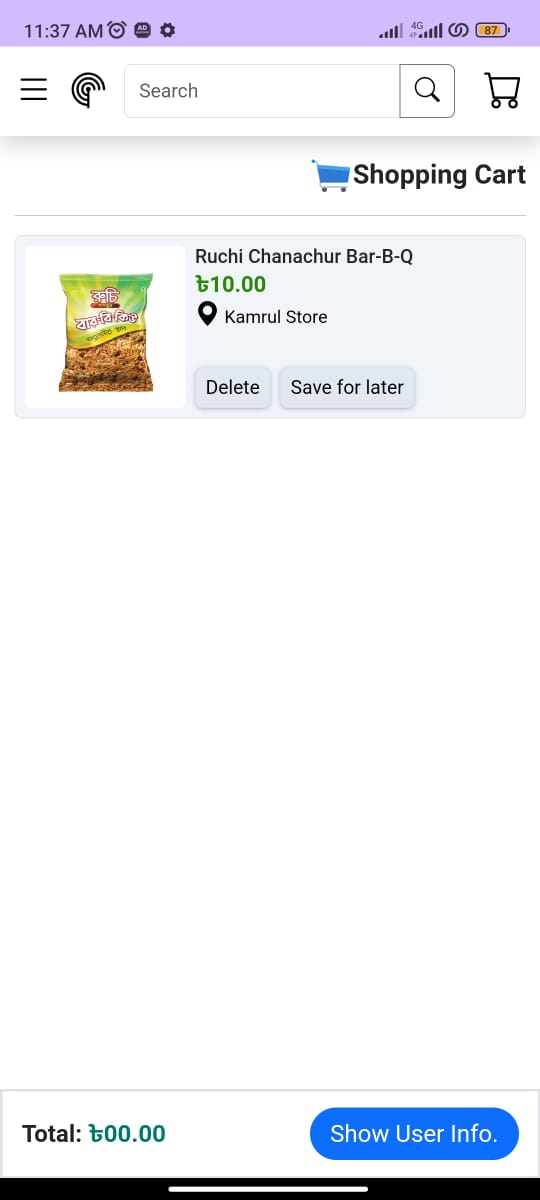
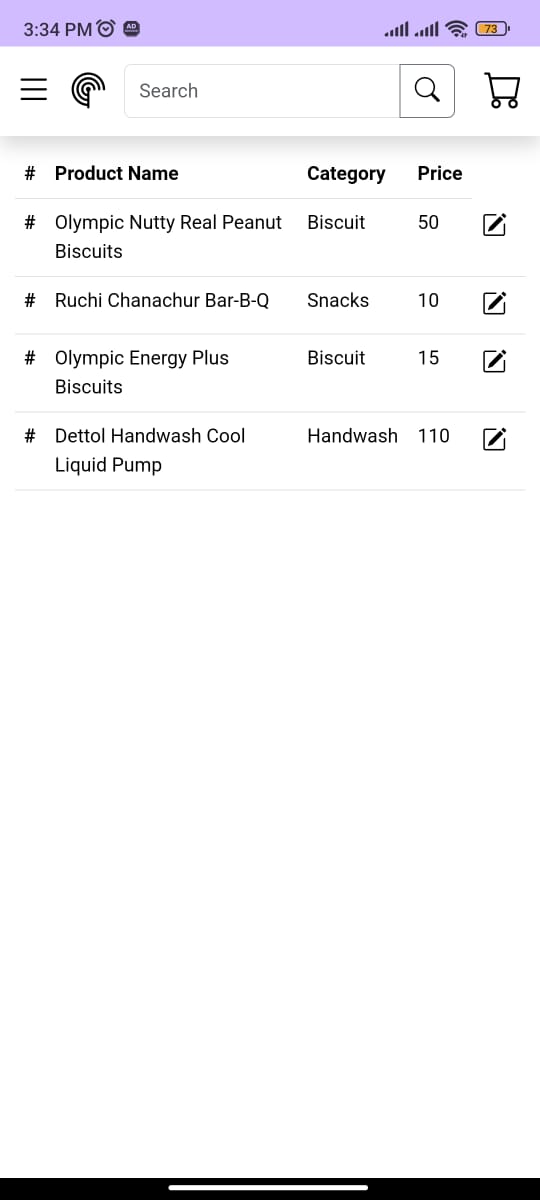
**Figure 4.8:** Shop Registration

Any user can register for a shop by providing necessary information and start selling products. The address of the shop defines the location of the products he will sell. Also some additional information are opening time, closing time, online orders, that also controls the visibility of its products to users. On Sunday, out system won’t show the products of a shop that is off on Sunday. Same goes for opening and closing time.

**Shopkeeper Page:**

After registering a shop, user can go to shop page where he has all the capabilities to manipulate his products. He can add product, view all the available products of his shop, add discount to specific product that will be shown in “Flash sell” or “Daily Deals” section, edit any product, reply to message of any user, view all the online orders and ship them to proper destination.

 **** 

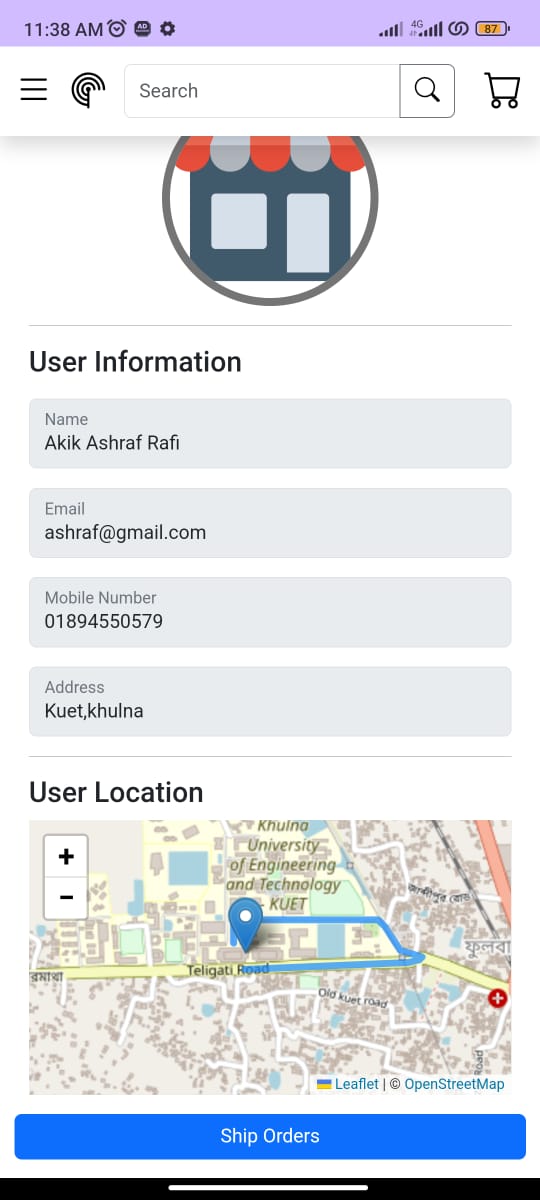
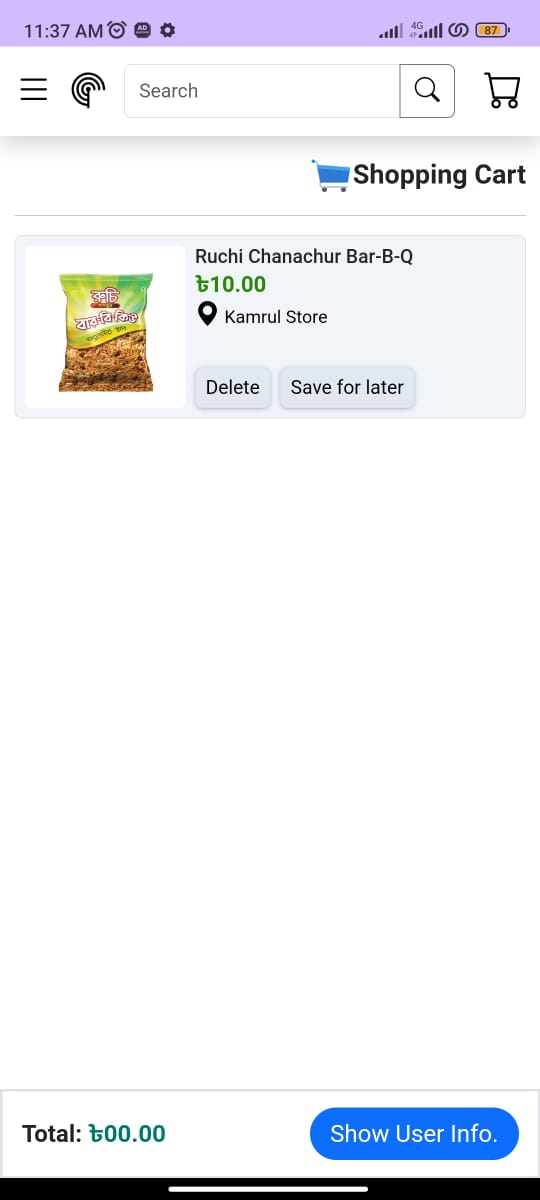
**  **

**Figure 4.9:** Shopkeeper Actions

**Online Orders:**

User can see all the online orders, view the product what user ordered for, view users location

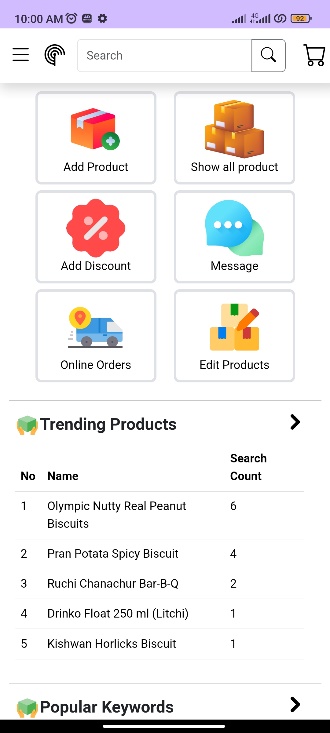
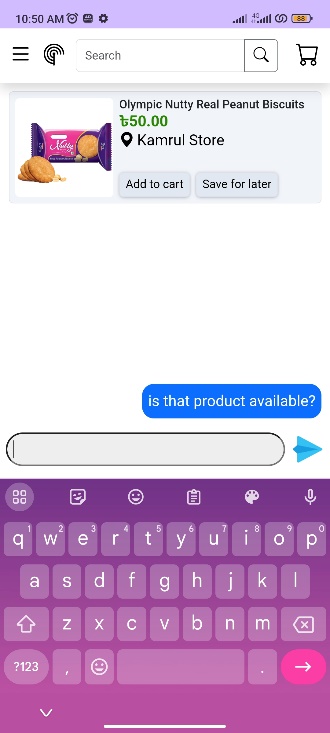
and finally ship the product to that location.



**Figure 4.10:** Online order system

**Messaging System:**

Customer can send message to shopkeeper from a product view, and shopkeeper can view the message form his message option in Shopkeeper page.

 ****

**Figure 4.11:** Messaging between customer and shopkeeper

**4.3 Morality and Ethical Issues and Approaches:**

During the development of ShopiRound, morality and ethical issues were major concern to us. So, we tried to investigate and addressed some of the issues like user privacy, data security, transparency and fair business practices.

**User privacy and Data Security:**

* **Data collection and usage:**

We collected user data like user location and browsing behavior to provide personalized services while ensuring user consent and respecting their privacy.

* **Data Security Measure:**

We safeguarded user information and prevented unauthorized access.

**Fair Business practice:**

* **Fair Representation of Products and Shop:**

While showing products and shop, we endured fairness and accuracy and their offers without biases or paid promotion. Our advanced algorithm ensured it.

* **Transparent Communication Between User and Shopkeeper:**

To avoid deceptive practiced and false representation, we ensured transparent commination system between user and shop owner.

**Social Impact and Accessibility:**

* **Accessibility for All Users:**

We ensured that our website and app is accessible to users with different technological capabilities.

* **Impact on Local Business:**

When other online platforms are more reassuring to larger shop over smaller shop, we ensured fair and equal representation of products from all kind of shop. Which makes our platform a good opportunity for small business to grow.

**4.4 Socio-economic impact and Sustainability:**

**Societal Impact:**

* **Convenience and Time Saving:**

One of the goals of our project is to avoid extensive physical search for any product and save user’s time.

* **Enhanced Access to Goods:**

Our application can provide user broader access to various products available in nearby shops.

* **Communication Engagement:**

Our application facilitates interaction between user and business man which eventually fosters communication engagement.

**Health and Safety consideration:**

We tried to minimize unnecessary travel and exposure to crowded area during shopping, contributing to public health, especially during health crises.

**Environmental Impact and Sustainability:**

* **Reduced carbon Footprint:**

Without a proper aim, user would have had to visit multiple stores, which may associate carbon emission to environment. Our application is saving environment by reducing unnecessary travels.

* **Sustainable Practice:**

ShopiRound encourages sustainability by promoting local businesses and reducing the need for long-distance transportation of goods, supporting a more sustainable supply chain.

**Boost in Shopkeeper’s Sales:**

Our application enhances shops visibility to potential customers who might not even know about the shop otherwise. Also, it empowers local small business by providing them a platform to showcase their goods to buys.

**4.5 Financial Analysis and Budget:**

**Table 4.1:** Estimated Cost for complete development, deployment and maintenance of the app for one year

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Task** | **Details** | **Estimated Time(hr/developor)** | **Expense rate(BDT/hr)** | **Estimated Cost(BDT/year)** |
| Planning | Requirment analysis | 3 \* 8 = 24 | 250 | 6000 |
| Project Planning |
| Design | Web UI | 3 \* 8 = 24 | 6000 |
|  | Android UI | 1 \* 8 = 8 | 2000 |
| Implemantion | Website | 30 \* 8 = 240 | 60000 |
|  | Android App | 2 \* 8 = 16 | 4000 |
| Deployment | Website | 4 | 1000 |
| Others | Map API |  | 12 \* 1000 = 12000 |
|  | Website Hosting |  | 1000 |
|  | Maintenance |  | 12\*9000 = 108000 |
| Total Estimated Cost | | | | 200000 |

**5.Conclusion:**

**5.1 Conclusion and Challenges Faced**

The development and implementation of ShopiRound have been a significant endeavor aimed at revolutionizing the way users interact with local businesses and streamline the shopping experience. Throughout the project lifecycle, several challenges were encountered and overcome:

**Technical Challenges:**

Integrating various technologies like ASP.NET MVC[6], Leaflet.js, SignalR[8], and Geoapify[6] all together was a challenging task. But to ensure good capabilities for our application, we had overcome this challenge and successfully integrated them and they are working as intended.

**Data Privacy and Security Challenges:**

Giving user privacy and data security while collecting location data and preventing potential threat was a challenge we faced. We implemented proper encryption technique and asked user permission and allowing transparency of the application to tackle these challenges.

Despite these challenges, the team's dedication and strategic problem-solving enabled the successful launch and operation of ShopiRound.

**5.2 Future Study**

We ensured enough room for improvement for our application. Some of them:

**Enhanced User Experience:**

We can improve user experience for our application by taking user feedbacks and demands. It’s a continuous process and we are planning on taking user opinion for our app betterment.

**Expansion and Scaling:**

We can expand our platform by reaching out to local business of different region and collaborate with them by showcasing how using our application can enable their business growth. Our application will be scaled to accommodate increased user demand.

**Innovation and technology integration:**

To cope up with modern innovation and improve our app’s capability, we are planning on integrating Ai to our system to provide automated services.

**References:**

1. Daraz, Online shopping platform,

<https://www.daraz.com.bd/> (Last access: 30 November 2023)

1. Rokomari, Online book selling platform,  
   <https://www.rokomari.com/book> (Last access: 30 November 2023)
2. Pickaboo, Online gadget selling platform,  
   <https://www.pickaboo.com/> (Last access: 30 November 2023)
3. Chaldal, Online grocery selling platform  
   <https://chaldal.com/> (Last access: 29 November 2023)
4. Ajkerdeal, Online e-commerce platfrom,  
   <https://www.ajkerdeal.com/> (Last access: 29 November 2023)
5. ASP.NET MVC, Web application framework,  
   <https://learn.microsoft.com/en-us/aspnet/mvc/> (Last access: 28 November 2023)
6. Geoapify, Map and routing API,  
   <https://apidocs.geoapify.com/> (Last access: 28 November 2023)
7. SignalR, Real time communication framework,  
   <https://dotnet.microsoft.com/en-us/apps/aspnet/signalr> (Last access: 28 November 2023)
8. Android Studio, Android Software Development Kit,  
   <https://developer.android.com/docs> (Last access: 28 November 2023)
9. SmarterASP– Free ASP.NET hosting website,  
   <https://www.smarterasp.net/> (Last access: 28 November 2023)