Thakur Educational Trust's (Regd.)

THAKUR COLLEGE OF SCIENCE & COMMERCE
(NAAC Re-Accredited Grade –A, CGPA -3.10 & ISO 9001:2008 Certified)
Shyamnarayan Thakur Marg, Thakur Village, Kandivali (East), Mumbai -400 101.

Project Documentation on

Online Food Delivery App

Submitted by:

Manisha Yadav	**22624**
Tanya Singh	**22628**
Lokesh Vyas	**22645**

March, 2022-2023

Under the guidance of:

Mrs. Poonam Jain

Submitted in partial fulfilment of requirements for qualifying

B.Sc.- (IT), Semester VI Examination

Thakur Educational Trust’s (Regd.)

THAKUR COLLEGE OF SCIENCE & COMMERCE
(NAAC Re-Accredited Grade –A, CGPA -3.10 & ISO 9001:2008 Certified)
Shyamnarayan Thakur Marg, Thakur Village, Kandivali (East), Mumbai -400 101.

Project Certificate

This is to certify that the project entitled **ONLINE FOOD DELIVERY APP** undertaken at the Thakur College of Science and Commerce by **Ms. Manisha Yadav (Roll No. 22624)**, **Ms. Tanya Singh (Roll No. 22628)**, and **Mr. Lokesh Vyas (Roll No. 22645)**in partial fulfilment of B.Sc. IT degree (Semester VI) Examination had not been submitted for any other examination and does not form part of any other course undergone by the candidates.

It is further certified that they have completed all the required phases of the project.

________________________ ________________________

Signature External Examiner Signature Internal Examiner

________________________ ________________________

Signature Project Guide Signature HOD/In-charge/Co-ordinator

ACKNOWLEDGEMENT

We have taken efforts in this project. However, it would not have been possible without the kind support and help of our Project Guide **Mrs. Poonam Jain, and many Faculties.** We would like to extend our sincere thanks to all of them.

We are highly indebted for their guidance and constant supervision as well as for providing necessary information regarding the project & also for their support in completing the project.

We would like to express our gratitude towards my parents & member of the teacher for their kind cooperation and encouragement which help us in the completion of this project. Last but not least, many thanks go to the head of the project, whose have invested his full effort in guiding the team to achieving the goal.

We would like to express our special gratitude and thanks to all the above-mentioned people for giving us such attention and time. Our thanks and appreciations also go to our colleagues in developing the project and the people who have willingly helped us out with their abilities.

PREFACE

This Report has been prepared as a part of our final project. The report is prepared with the view to include all the details regarding the project we carried out.

Every organization, whether big or small, has challenges to overcome and manage the information of Category, Food Item, Order, Payment, and Confirm Order. The aim is to automate its existing manual system with the help of computerized equipment and full-fledged computer software, fulfilling their requirements, so that their valuable data/information can be stored for a longer period with easy access and manipulation of the same. The project describes how to manage for good performance and better services for the clients regarding Food Delivery.

In this project, we had taken various types of concerns in mind regarding Food Delivery and have developed an application to solve most of the problems.

❖

TABLE OF CONTENTS

LIST OF FIGURES

1

INTRODUCTION

1.1 Background

In this modern generation, ordering online has become a norm, whether it is ordering your clothes, gadgets, accessories, or even food. The rapid growth of electronic payment has made buying things on the internet must fast and easier. During the pandemic, online ordering was the safest way to avoid contact with others, and since then the trajectory of online ordering has increased much more. For a seller to run his business successfully, they need to put their business online. The food industry has also adapted to this change. Whether it is large food chains like Domino's, McDonald’s, and Subway, or small restaurants have started selling online. Due to this, the restaurants have been able to expand their businesses and gain customers from larger areas. This system is efficient as it reduces time, improves management, and reduces human errors. It is also beneficial for the users, as they do not have to travel to the restaurant to get their food, they can just order with their phone.

The typical mechanism behind food delivery is as follows: the user on the food delivery application chooses a restaurant to order food from, checks the menu list, selects food to order, and proceeds to payment. Once payment is done, an employee i.e., the rider nearby the location picks food from the restaurant and delivers it to the user’s home. This also increases employability as a platform is provided to deliver food to the houses.

The basic features that are needed by the customers in an application are making orders, food reviews, order history, restaurant profile, profile setting, order status, and tracking orders.

The online food ordering system sets up a food menu online and customers can easily place the order as per they like. Also with a food menu, online customers can easily track their orders. The management maintains the customer database and improves food delivery service. The Restaurant management system motivates us to develop the system. There are various facilities provided so that the users of the system will get service effectively. Also, the system considers Restaurants facilities for the customers. Increasing the use of smartphones is also considered as motivation so that any users of this system get all services with a single click.

The main aim of the application is to provide simplicity and improve the efficiency of the ordering process for both customers and restaurants, minimize manual data entry, and ensure data accuracy. Customers can easily view the product menu of the restaurant and placed the order accordingly with the acknowledgment of the confirmation of the order. When considering online food delivery applications there are several scopes right from the selection of the food to the delivery process.

1.2 Objectives

- To design a user-friendly system that provides the latest information to customers:

  If the food is not available, the food will not appear on the menu. When the customers view the menu, they cannot view the food. This can solve the problem of staff forgetting to inform the latest information to customers. The user interface of the system should also be clean, clear, and attractive to the user. The system should be easy for to use the users. A user-friendly system is when the customers use the system for the first time, they know how to view the menu and make their order. The system also should not be complex as the customers might not know how to use it.

- To develop a system that includes keeping track of the order:

  Using this application users can keep track of the order. The delivery boy will keep updating the order steps and the customer will get notifications on the application and view how much more time is required for delivery to be done.

- To ensure the customers can cancel their order:

  The restaurant that uses the traditional method includes many steps when customers wish to cancel their order. The customers need to inform the staff, then the staff will inform the chef. In this restaurant ordering system, the customers can cancel their orders without interacting with the staff. They can just click on the cancel button to cancel their order. The customers can cancel their order with one condition within an allotted time.

- To provide an easy payment procedure:

This application also contains the payment method to order the food. This includes COD (Cash on Delivery) and Card payment.

Existing System:

Earlier customers had to go to the restaurants to eat, or they had to browse through the restaurant's brochures to choose their food and place an order by calling the restaurant. This is a time-consuming process and the customer should have the brochure to place the order. Customers can't even track their orders or know their status of it. Then would have to call the restaurant repeatedly to know the status of their order. Also, there were no options to navigate to other restaurants and find other options too for the food items.

Proposed System:

With the help of the food delivery app, the user doesn't have to call the restaurant to order food, they can browse through the menu on the app and place an order directly. They can also make payments online and can track the status of the order. The app also has a GPS by which the users can track their delivery and the details of the delivery man. This app is also beneficial for the owner as they can expand their businesses and gain customers from a wider area and create a brand image for themselves.

1.3 Scope

The main scope of the application is to provide simplicity and improve the efficiency of the ordering process for both customers and restaurants, minimize manual data entry, and ensure data accuracy. Customers can easily view the product menu of the restaurant and placed the order accordingly with the acknowledgment of the confirmation of the order. When considering online food delivery applications there are several scopes right from the selection of the food to the delivery process.

- The convenience of feedback:

Once the order has been placed the customer can provide their feedback in the form of reviews and by rating the app. This feedback allows the other customer to know about the application more accurately and the ratings to attract them to the app.

- Time-Saving

The customer can easily place an order at any specific corner with just one click. Instead of being engrossed in the kitchen for many hours the customer can place the order and save

time. No need of waiting long in the queue for placing the order.

- Easy tracking

Providing customers with accurate shipping information and tracking updates. An order tracking process lets you know if the order is still waiting to be shipped or if it is en route to your receiving department.

- Managing menu

It is much easier and considerably cheaper to create and maintain a great-looking menu that will impel your customers to order from the menu list. Not only do you get rid of the burden of printing and relieve yourself of the printing fees, but you also gain a great deal of flexibility in changing the menu whenever you want.

- Easy Payment

It is fast, easy, and comfortable. It provides various options for payment like online payment (credit card) or cash on delivery. Moreover, online payments benefit discounts and coupons with the restaurant returning to attract a customer.

1.4 Problem Definition

Nowadays, many restaurants use traditional restaurant ordering systems to serve customers. In the traditional restaurant ordering system, the staff writes down the foods that the customer order. The paper will then pass to the kitchen and the chef will start to cook. This has caused a few inconveniences. The staff might make some errors while writing down the order. Sometimes, when the staff writes in hurry will make the handwriting difficult to understand. The staff might lose the order paper and customers might also receive an incorrect bill.

One of the problems faced due to restaurants is that while using the traditional ordering system the customer does not know the time for the preparation of the food. Some of the customers might have the next schedule after their lunch or dinner. They need to know the time preparation so that they can plan their schedule wisely. Especially when there are a lot of customers, the customers might think their order has been forgotten if their food has still not yet been served in a long time. It will be good if there is an estimated time to prepare the food shown to the customers.

Furthermore, some of the customers might want to change their food or cancel their food.

The customers are only allowed to cancel their order if the chef has not yet started cooking. The customer first needs to inform the staff of the cancellation and then he can inform the chef of the cancellation. Though the order is cancelled sometimes in a situation when there are a lot of customers in the restaurant the staff may forget to inform the chef and the customer is unable to cancel the order. They do not need to wait for the staff to serve them and waste the time. A cancel button should be displayed so that the customers can cancel their order if the chef has not yet started cooking.

Moreover, it sometimes becomes difficult for the staff to manage every customer and give the proper attention to the ordering of the food. Therefore, an online food ordering system must be required to overcome these problems and for the satisfaction of the customer.

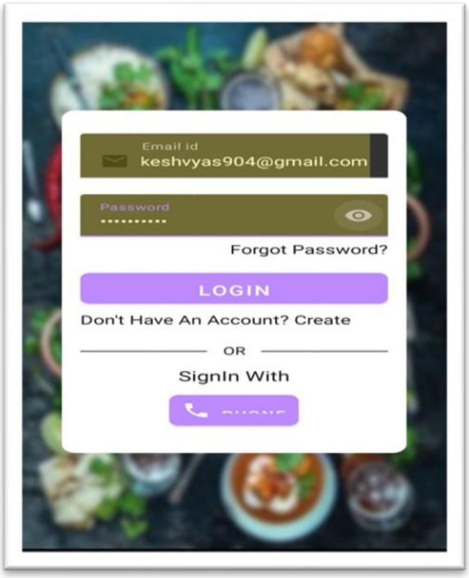
1.5 Functional Requirements/SRS

Software Requirement Specification (SRS) as the name suggests, is a complete specification and description of requirements of the software that needs to be fulfilled for the successful development of a software system. These requirements can be functional as well as non-functional depending upon the type of requirement. This helps to understand the modules more accurately and their working. Depending upon information gathered after the interaction, SRS is developed which describes the requirements of the software that may include changes and modifications that are needed to be done to increase the quality of the product and to satisfy customer demand.

- Functional Requirements

The functional requirements form the core of the entire software development process. It is very important to understand what the client's needs are, the need for a software product, output expected from the system. In short, the key functionalities a software system must deliver. The various functionalities are been performed by the user as well as the delivery boy in the application.

1. **Login/Registration Module:** once the registration is done then this module is provided to determine whether the person logging in is a user/customer or delivery boy.
   a. *Input:* login details and password
   b. *Process:* Once the email and password are entered, the process of validation occurs to identify whether the email and password is matching with the one saved in the database.



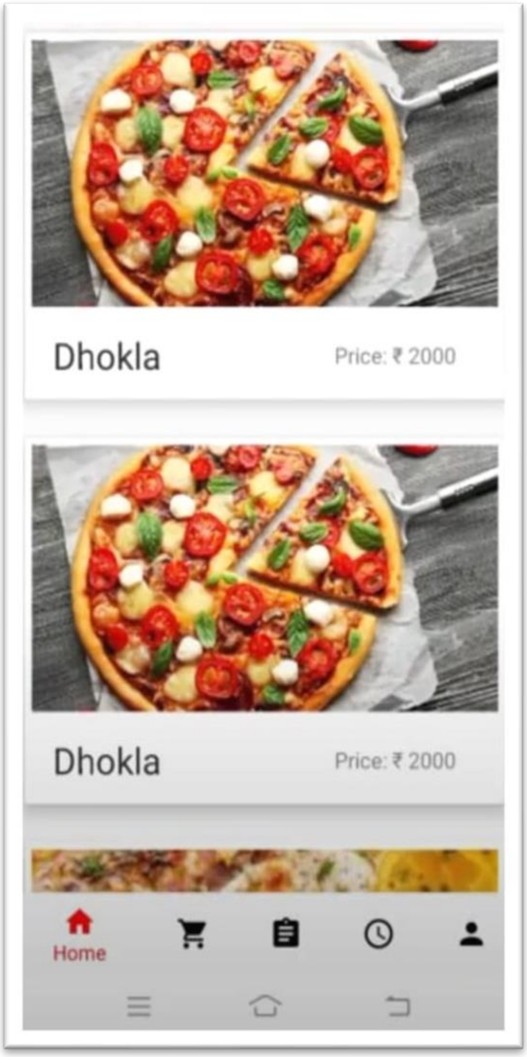
Email id keshvyas904@gmail.com

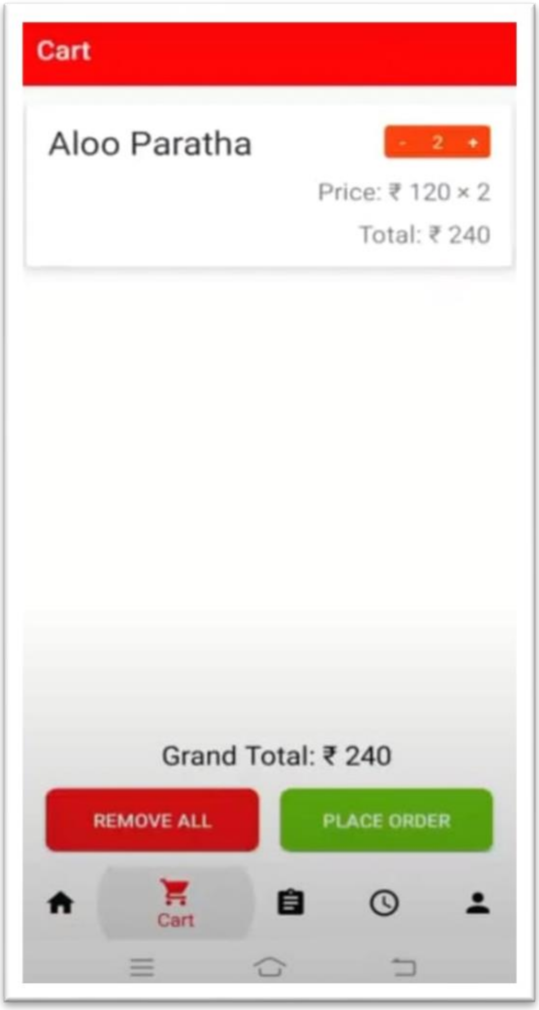
Password ••••••••••

2. **Admin Module:** In this module, the admin can do the following functionalities when he/she logs in:
    - **a.** *View the customers:* can view the customer information.

    - **b.** *Update delivery boy information:* can add, delete, and update information related to the delivery boy.

3. **User Module:** In this module, customers can do the following functionalities when he/she logs in.
    - **a.** *Registration/Signup*: When the customer wants to order the food, he/she must be registered to the application.
    - **b.** *Log in to the system:* The user can easily log in to the application whenever required and place the order.
    - **c.** *Navigate the restaurant's menu:* Once login the customer can view the list of the food on the menu. It consists of various varieties of food as per the customer's choice.
    - **d.** *Add an* item *to the cart:* The customer can add the item/items to the cart after the selection from the menu list.
    - **e.** *Delete the item/items:* The customer can also delete the food item from the cart whenever the customer wants.
    - **f.** *Payment details*: Lastly the final step is to proceed with the payment part which provides the option for current payment or payment on delivery.
    - **g.** *Logout:* The customer can log out from the application whenever he wants to.

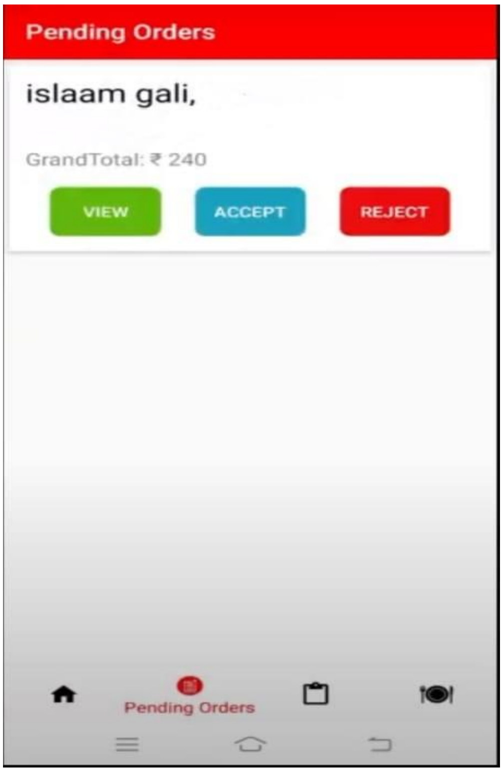




4. Delivery Boy Module: In this module, the delivery boy is enabled to perform the following functionalities:

- **h.** *Registration/Signup:* whenever the delivery boy first joins the restaurant he needs to register with the app or sign up.
- **i.** *Log in to the system:* after signing up, he can now login into the app.
- **j.** *New orders:* receives the notification of new orders.
- **k.** *View orders:* can view active orders.

l. *Confirm the delivery:* send a notification to the customer that the delivery is accepted.



Pending Orders

islaam gali,

GrandTotal: ₹ 240

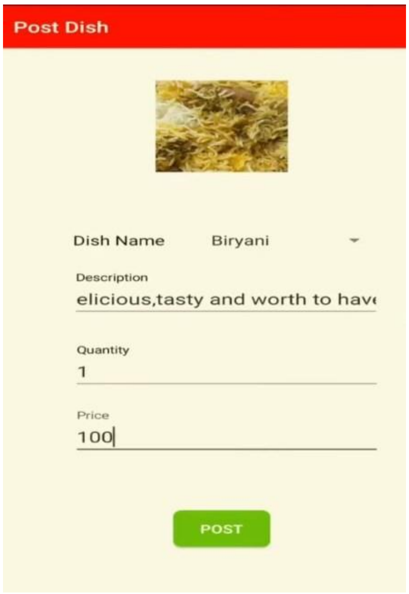
4. **Restaurant Module:** Restaurant modules are individual pieces of tech that help restauranteurs manage a certain aspect of running their business. In this, we have more than one option for restaurant search.

   a. Provide the options for restaurants.

   b. *Update the information:* can update information related to the food items.

   c. *View orders:* can view active orders.

   d. Add/Delete food items: can add and delete food items from the list.



Post Dish

Dish Name Biryani

Description
elicious,tasty and worth to have

Quantity
1

Price
100

7. **Payment Module:** It contains different payment gateways to pay for food delivery.

    a. Provides COD (Cash on Delivery) and card payments for customers.

    b. The OTP will be generated after the payment is processed and once the OTP is successfully confirmed and verified the payment will be completed and the order will be placed.

- Non-functional Requirements:

When we order food online, the food must be the one that is selected, but beyond that many companies offer delivery within 30 minutes, else free delivery. This free delivery can be thought of as a non-functional requirement that extends beyond providing the basic functionality, that is food delivery.

Similarly, the following are the non-functional requirement:

- **Security:** Secure access to confidential data of the customer considering the payment mode and location.
- **Efficiency:** The system should be efficient so that it won't get hung even if there are multiple orders placed at the same time.
- **Performance:** The system should perform smoothly and fast without any obstacles.
- **Usability:** Online food ordering applications should be user-friendly and usable by their intended users, i.e., the customers and administrators of the system.

1.6 Details of Hardware and Software

- Hardware Used*:*
  Processor: Intel Core i5
  Operating System: Windows 10
  RAM:8GB.
- Software Used*:*

1. Android Studio:

Android Studio serves as an Integrated Development Environment (IDE) for Google's Android Operating System. The studio is built on IntelliJ IDEA and has been specifically designed for Android Development. The Android Studio is open source, which means anyone can download and use it.

Android SDK is the software development kit that is used for developing an Android application. The Android SDK kit includes a set of development tools such as required libraries, a debugger to help you spot errors and rectify them, an emulator to run your application on a virtual machine, relevant documentation for Android application program interfaces, and a sample source code.

We used android as our platform because:

- It is open-source software available to build android applications.
- We were already aware of the basics of the Android and Java languages, which made it easy to get to the advanced codes and techniques in Android.
- Many in-built libraries helped to integrate various functions easily.

2. Firebase:

The Firebase Realtime Database is a NoSQL database from which we can store and sync the data between our users in real-time. It is a big JSON object which the developers can manage in real time. By using a single API, the Firebase database provides the application with the current value of the data and updates that data. Real-time syncing makes it easy for our users to access their data from any device, be it web or mobile.

We used Firebase as our platform for the backend/database because:

- Realtime Database Provides Synchronize Data
- Web Hosting Firebase provided us to deploy a single-page web app, a mobile app landing page, and progressive web apps with ease.
- Multi-Platform Authentication with Firebase
- Firebase offers many exceptional integration services and allows users to synchronize the real-time data without refreshing the screen.
- It provides google analytics.

3. Gradle:

Gradle is a build system (open source) that is used to automate building, testing, deployment, etc. "build.gradle" are scripts where one can automate the tasks. Every Android project needs a Gradle for generating an APK from the .java and .xml files in the project. Simply put, a Gradle takes all the source files (java and XML) and applies appropriate tools, e.g., converts the java files into hex files and compresses all of them into a single file known as APK that is actually used.

- User Interface Constraints*:*
  Using this app is simple and intuitive. A user familiar with basic app navigation skills should be able to understand all functionality provided by the portal.
- Hardware Constraints*:*
  This is intended to be an android application. The user can use this app on any android mobile phone with an internet connection. No further hardware device or interface is required.
- Software Constraints*:*
  This app would run Android 5.0 Lollipop or above (this app will be able to run on approximately 98.8% of the devices).
- User Interface Constraints*:* The portal shall be able to interface with other components according to their specifications.

SYSTEM ANALYSIS AND DESIGN

2.1 Developing model:

- Incremental Model:

Developing software is a very complex project, therefore we have a software development model to simplify our work. The whole process of developing software is called Software Development Lifecycle (SDLC). It includes a step-by-step guide for creating and maintaining the software. It includes phases starting from Planning, Analysis, Design, Implementation, Testing, and Maintenance.

For the development of our food delivery app, we have decided to use the **Incremental Model or The Iterative Model.**

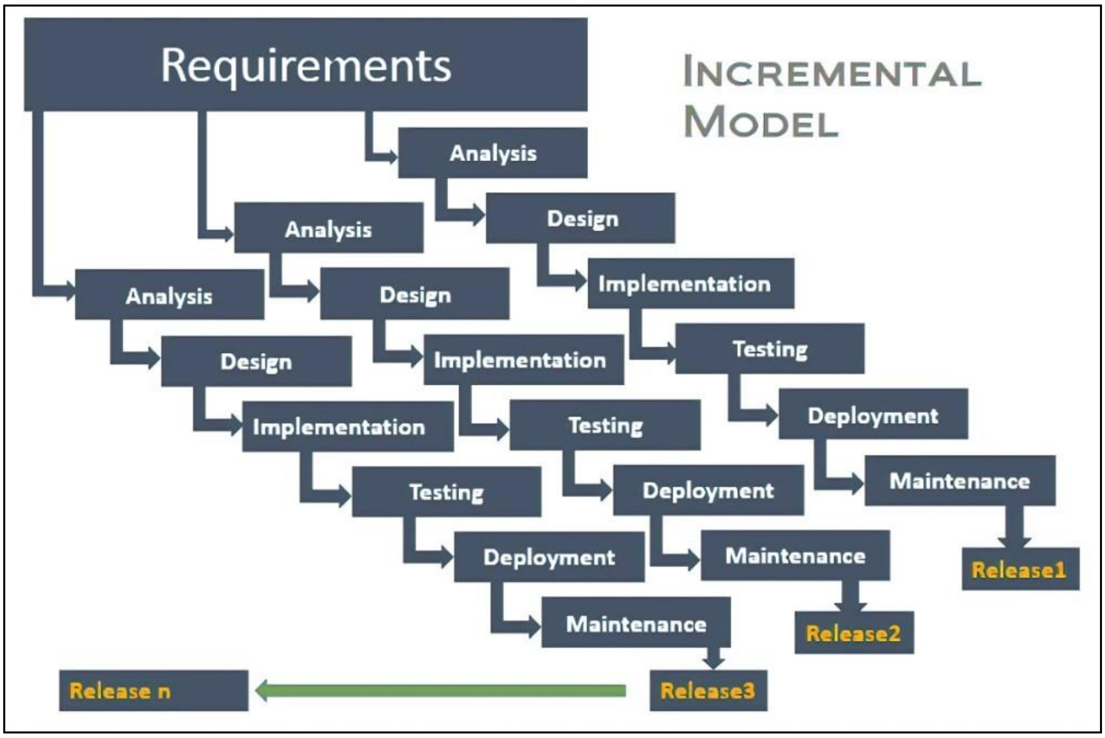


Fig 2.1 Developing Model for the project

The incremental model has the same phases that are in the waterfall model. But it is iterative. In the iterative model, the iterative process starts with a simple implementation of a small set of software requirements and iteratively enhances the evolving versions until the complete system is implemented. This model is repeated, producing a new version of the software at the end of each iteration of the model. At each iteration, design modifications are made and new functional capabilities are added. As this model supports our development requirement and allows the team to iterate models to improve upon them, it was the preferred model for our project.

2.2 Lifecycle:

The following diagrams show the detailed lifecycle of the project:

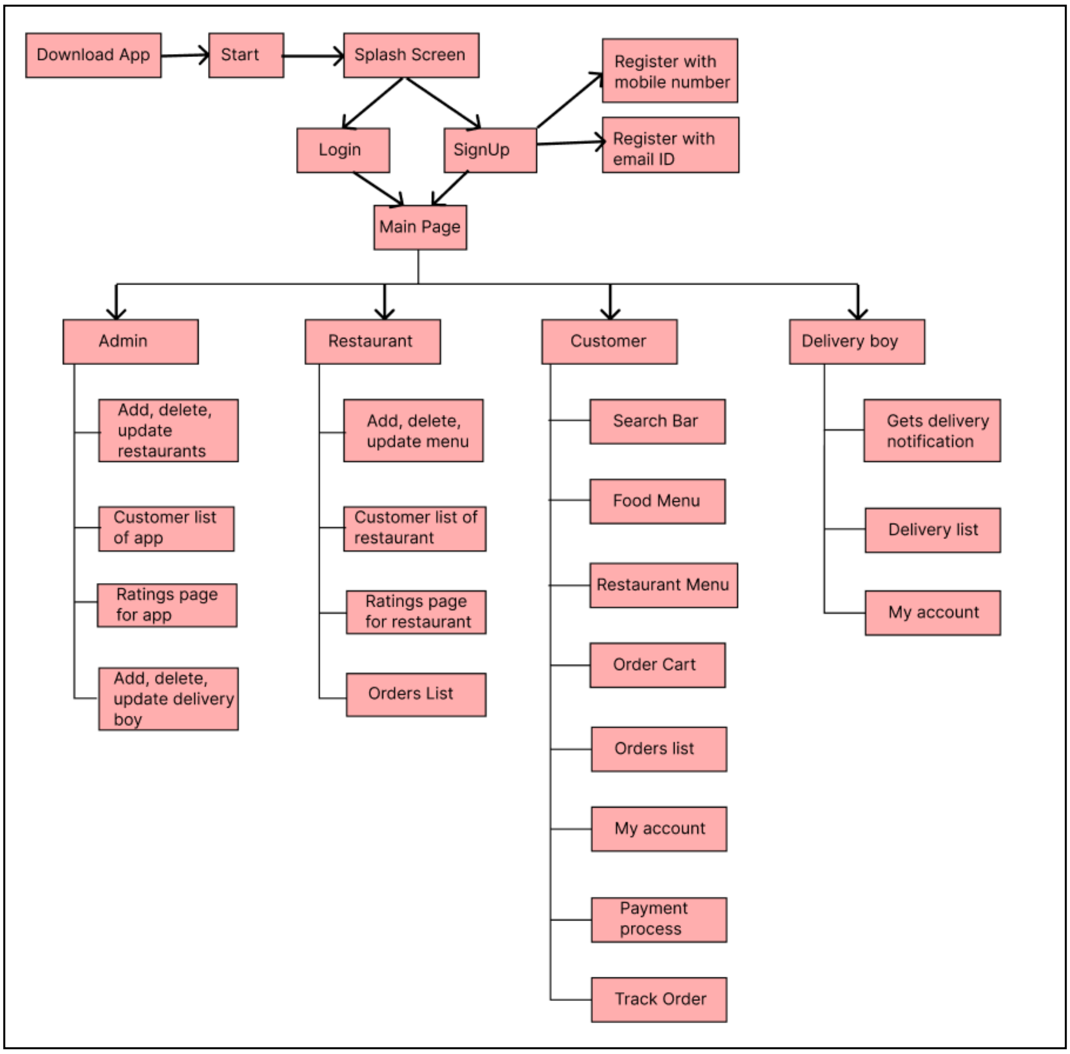
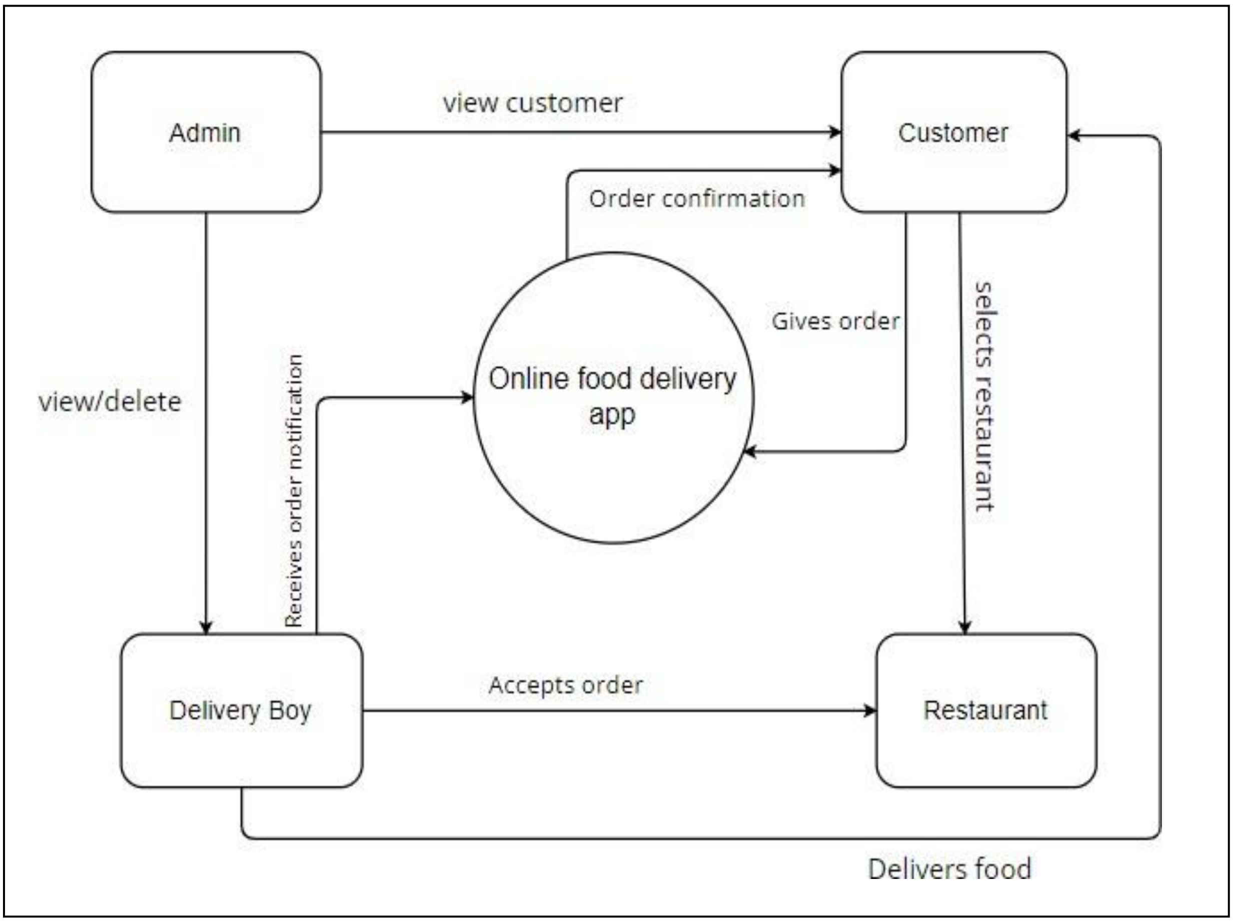


Fig 2.2. Lifecycle of the project

2.3 Context Diagram:

The Level 0 Data Flow Diagram, the **Context diagram** is the highest level in a Data Flow Diagram. It is a tool popular among Business Analysts who use it to understand the details and boundaries of the system to be designed in a project. It points out the flow of information between the system and external components.



2.3.Context Diagram for the project

In the following figure, we see the context diagram for the Online Food Delivery App. It explains how all the modules work together with the system and what are the inputs and outputs from each module.

2.4 Data Flow Diagram:

A Data Flow Diagram (DFD) is a traditional visual representation of the information flows within a system. It shows how data enters and leaves the system, what changes the information, and where data is stored.

The Data Flow Diagram has 4 components:

Process: Input-to-output transformation in a system takes place because of process function. The symbols of a process are rectangular with rounded corners, oval, rectangle, or circle.

Data Flow: it describes the information transferred between different parts of the systems. The arrow symbol is the symbol of data flow. A relatable name should be given to the flow to determine the information which is being moved.

Data Store: The data is stored in it for later use. Two horizontal lines represent the symbol of the store. The warehouse is simply not restricted to being a data file rather it can be anything like a folder with documents, an optical disc, or a filing cabinet.

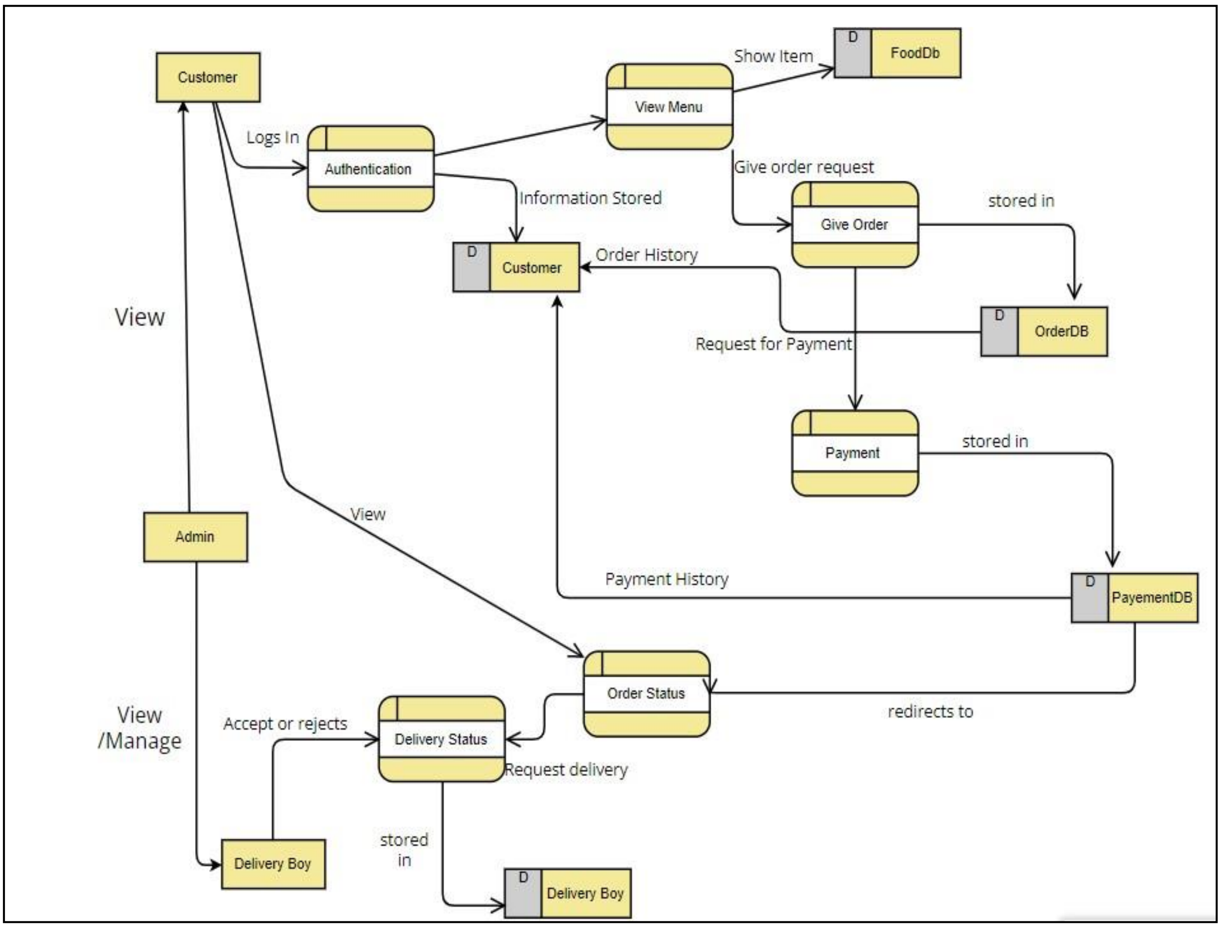


Fig 2.4. Data Flow Diagram for the project

As seen in the above diagram, the process starts at the user end, the customer logs into the system by entering credentials, this data is sent to the authentication process to confirm the credential. When authenticated successfully, the user is allowed access into the system. The user can then view his own data in profile or menus. Assuming the user selects menu, the flow would direct the user to the menus tab. Similarly, if the user clicks on profile here, they can view the Some of the data that's stored like order history, payment history, general details, etc. This information table gets it's input from other areas and processes such as order database, payment database, authentication credentials, etc.

Each data process is connected to its required process to ensure smooth flow of data between the processes that require it. They form an interconnected web flow which enables both the user, admin and any required party to monitor and track processes such as orders, payment, order status, location of the delivery person etc.

2.5 ER Diagram:

Entity relationship diagrams are a way to represent the structure and layout of a database. It is used frequently to describe the database schema. ER diagrams are very useful as they provide a good conceptual view of any database, regardless of the underlying hardware and software. An ERD is a model that identifies the concepts or entities that exist in a system and the relationships between those entities.

There are three main objects in an ER Diagram:

1. Entities
2. Attributes
3. Relationships

The below ER diagram for the project shows various attributes related to the entity included in the project:

1) The underlined attributes are the primary keys for that particular entity

2) And the dotted circle attributes are the foreign attributes.

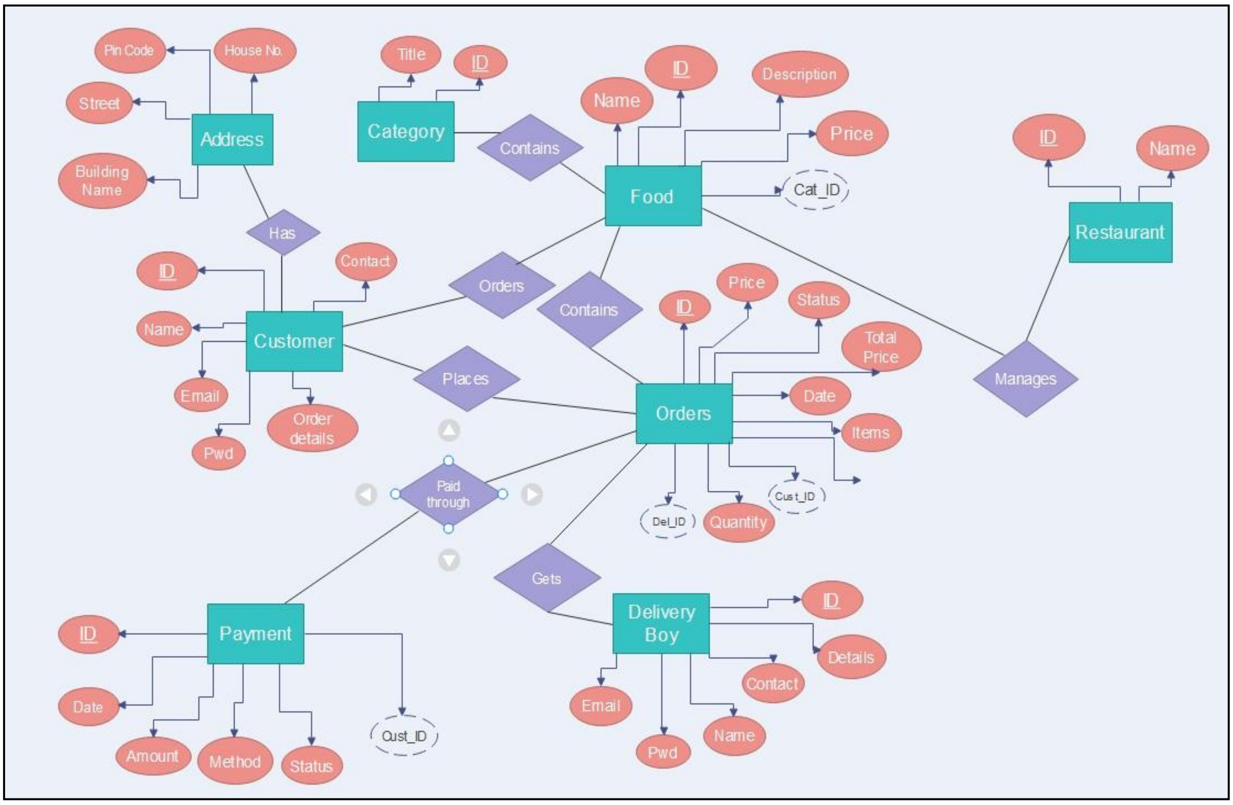


Fig 2.5. ER Diagram for the project

In the above diagram it contains of the 8 entities where each of them shows how the implementation actually is working. In the food entity it consists of the name, id, price, etc which are the contents of the food entity. As a major step in the Food Ordering System, it determines the included data when a customer orders or transacts. Customer order details data examples include name, address, and phone number. This process takes consumer meal orders, quantities, and payment information. Further after the ordering the food it navigates to the payment entity where it provides id, amount, and status of the food.

2.6 Use Case Diagram:

A use case diagram is a visual representation of how a user might interact with a program. It depicts the system's supposed structure and behaviour. It represents how an entity from the external environment can interact with a part of the system. **An actor** in a use case diagram is

any entity that performs a role in one given system. This could be a person, organization, or an external system and is usually drawn like a skeleton. The below diagrams depict the use case diagram for an online food ordering system.

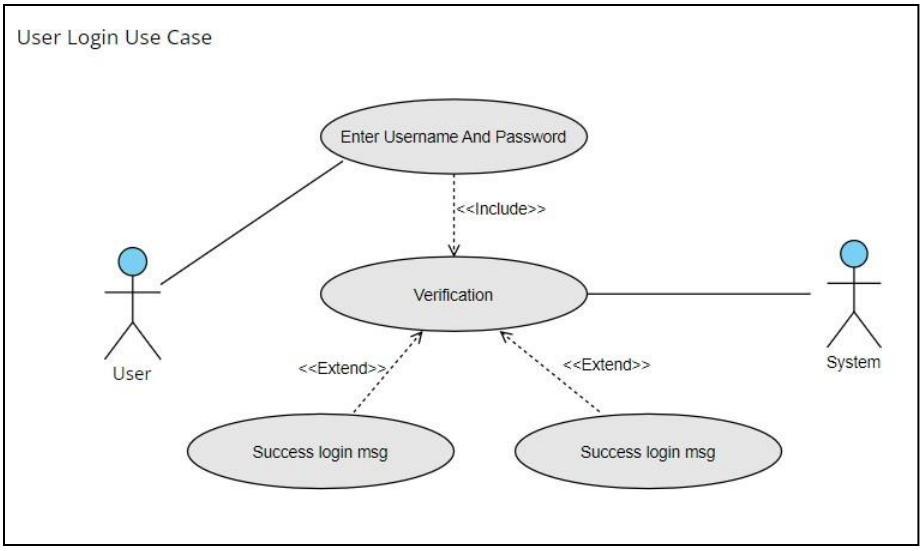


Fig 2.6. User Login Use-case diagram

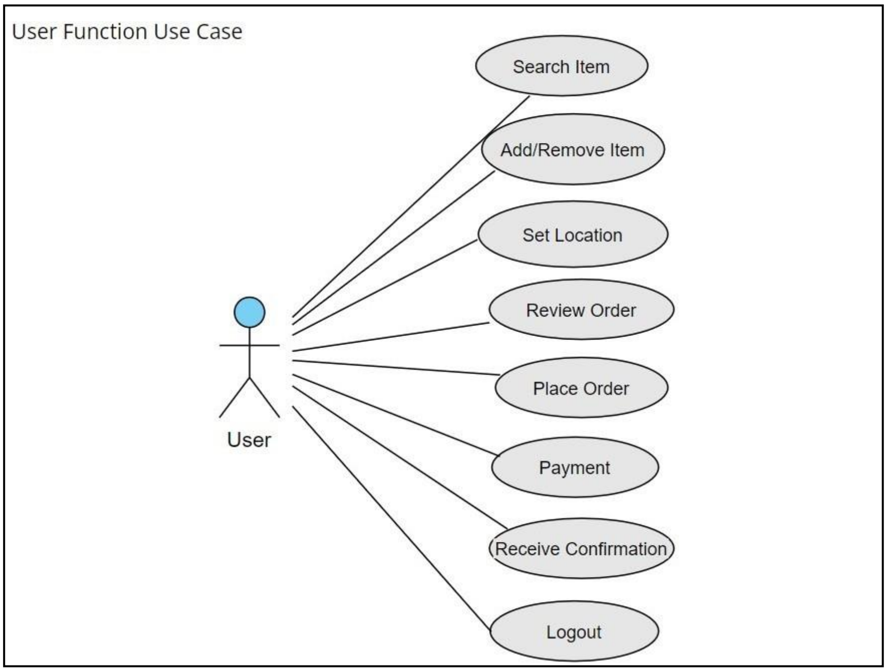


Fig 2.7. User Function Use-case Diagram

- User login and function: This is the first page where the user must proceed to log in to access the application after registration where it will require the user's name and password and after successful login can move to the application. Once the user enters the home page, he/she can search for the food items and add them to the cart. Th e user can also view the selected items and after setting the location can proceed to place.

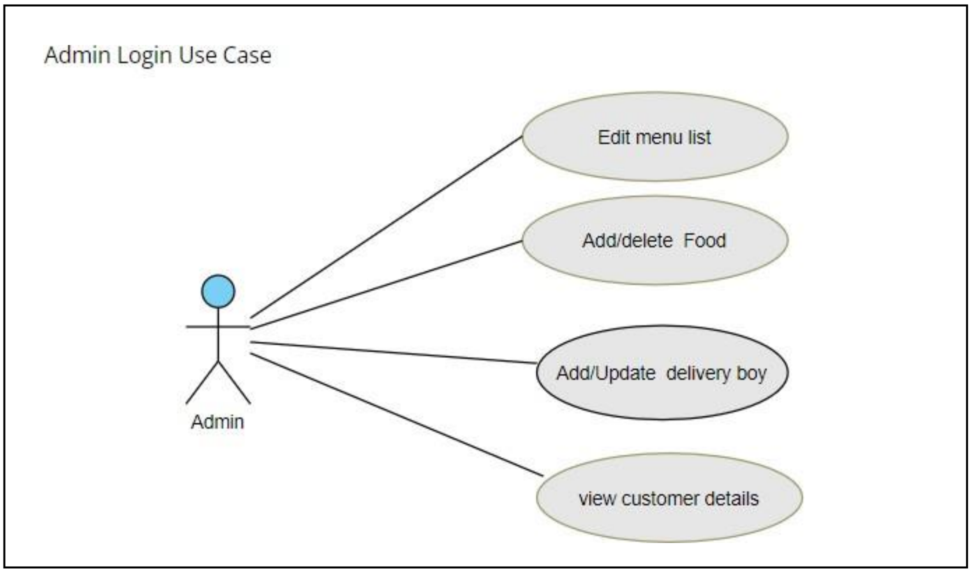


Fig 2.8. Admin Login Use-Case Diagram

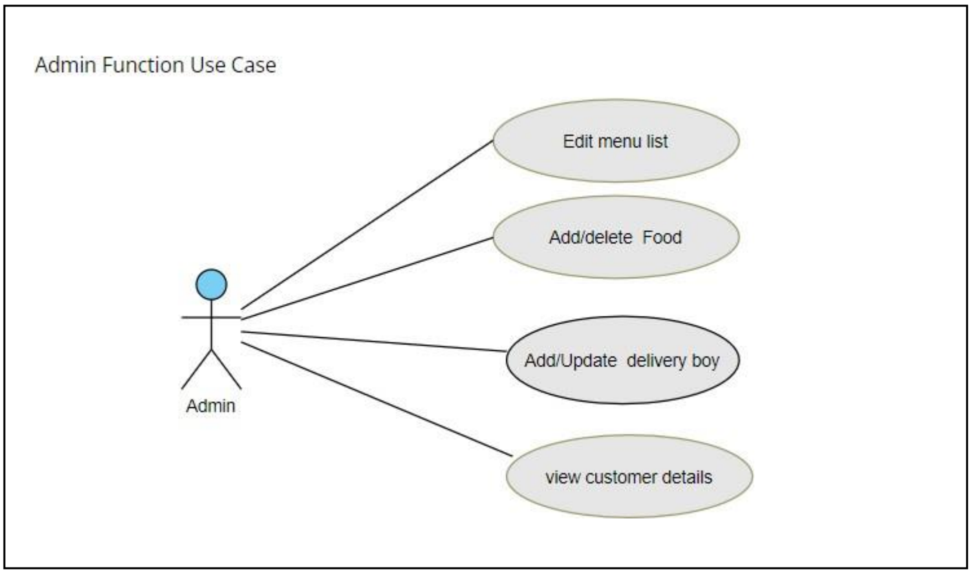


Fig 2.9. Admin function Use-Case diagram

- Admin login and function: The admin after successful login can navigate to the home page and view the user’s order, payment, and user details, as well as edit the menu list, and add and delete delivery boy.

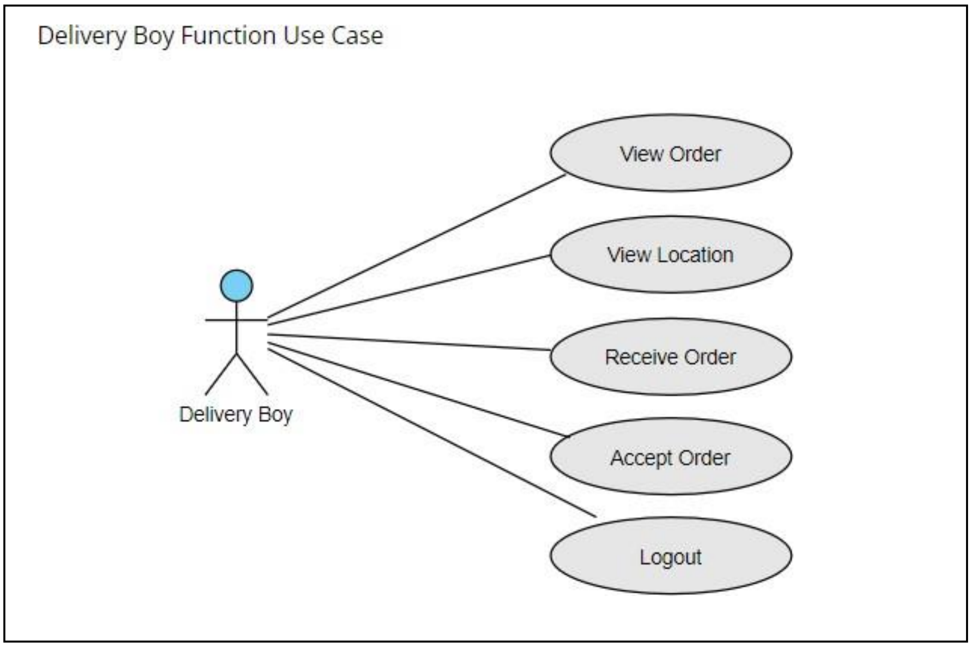


Fig 2.10. Delivery Boy Login Use-case diagram

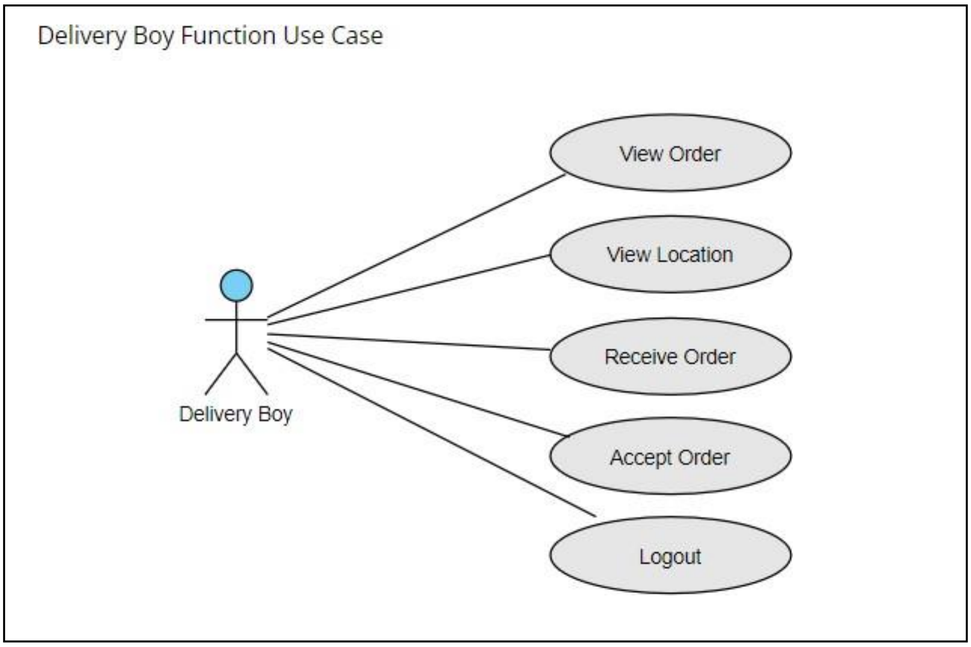


Fig 2.11. Delivery Boy Function Use-case diagram

- Delivery boy login and function*:* Once the order is been placed by the user the delivery boy gets the notification and after the login process, he can view the details of the location and payment of the order.

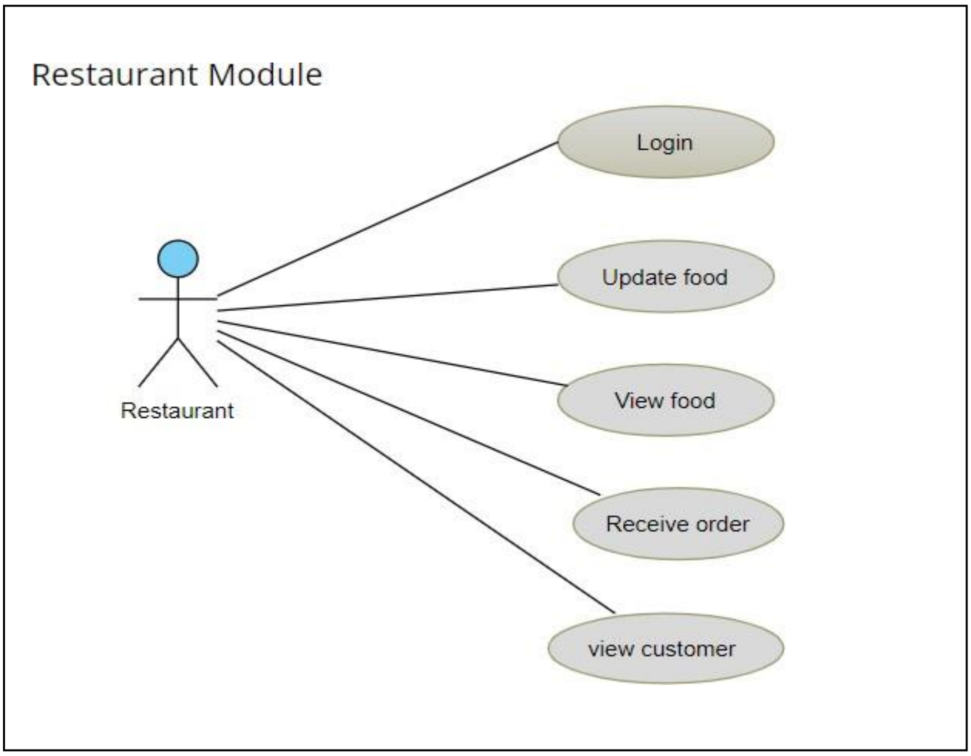


Fig 2.12. Restaurant module use-case diagram

- A restaurant module can easily log in to the system and can update the information regarding the food items, view the food items, receive the orders view the customer details. The details regarding the restaurant can be viewed in this module

2.7 Activity Diagram

The activity diagram is another important diagram in UML to describe the dynamic aspects of the system. An activity diagram is a flowchart to represent the flow from one activity to another activity. The activity can be described as an operation of the system.

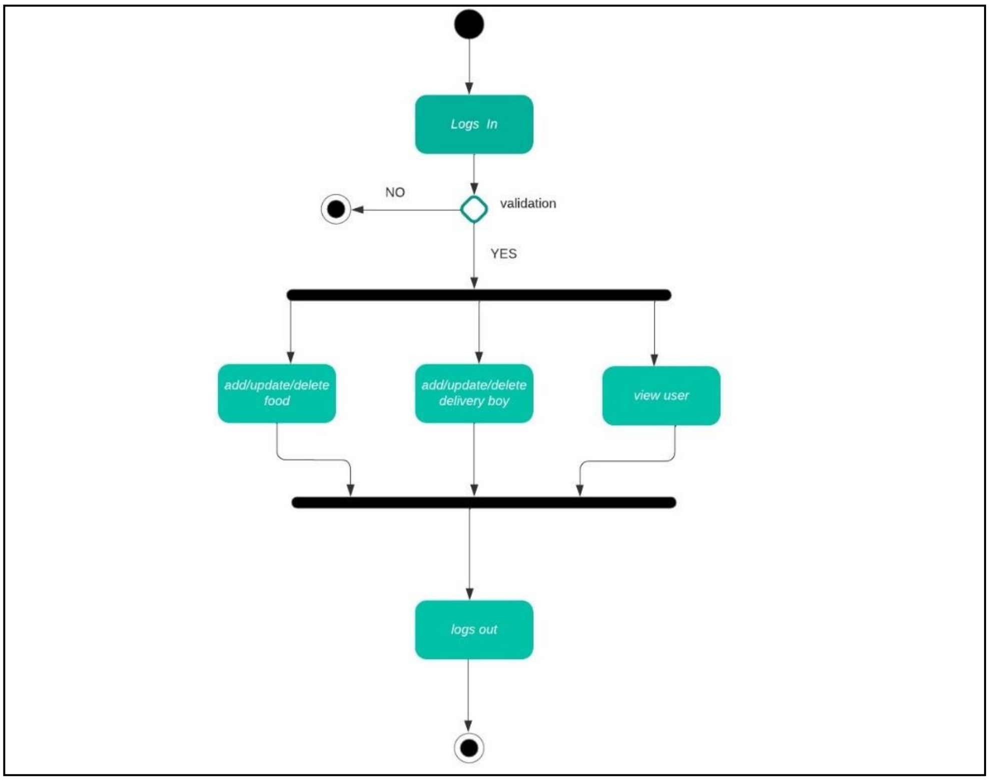


Fig 2.13. Activity diagram of the project

The first activity diagram is for the admin purpose, and shows the following steps:

- First, the admin logs in, and the validation of the password takes place, if the password is valid then the admin logs in otherwise, he is redirected to the same login page.
- After logging in, the admin can perform add, update, and delete functionalities for the food menu and its dishes

- Also, he can add, update and delete delivery boy-related information
- And can view user/customer-related information.
- Then he can log out if he wants.

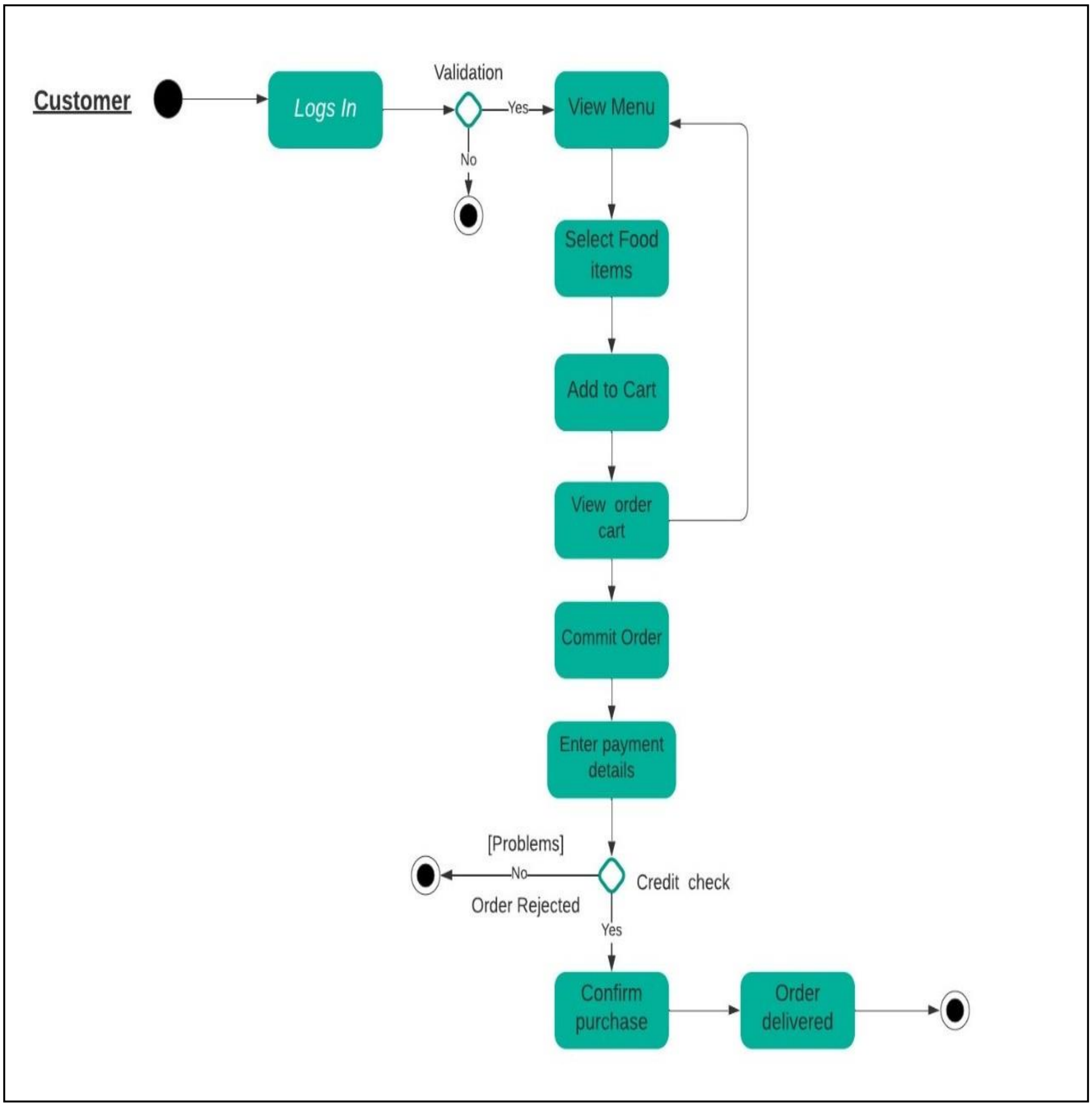


Fig 2.14. Customer Activity Diagram

The second activity diagram is for the admin purpose, and shows the following steps:

- The customer can register with the required information, and then logs in. the validation takes place, if the password is valid then the user logs in otherwise, he's redirected to the same login page.
- After logging in, a customer can do multiple functionalities from searching the food menu or dishes to ordering what's already in the cart.

- After adding to the cart, the user can view cart items, place the order and make the payment. If the payment gets confirmed then only the order gets confirmed otherwise the user is asked to wait till any further notifications.
- Once payment is confirmed an order is placed, the user can track the order.

The third activity diagram is for the delivery boy's purpose, and shows the following steps:

- First, the delivery boy logs in, and the validation of the password takes place, if the password is valid then the delivery boy logs in otherwise, he’s redirected to the same login page.
- Then he’s able to see the various orders list from which he can accept

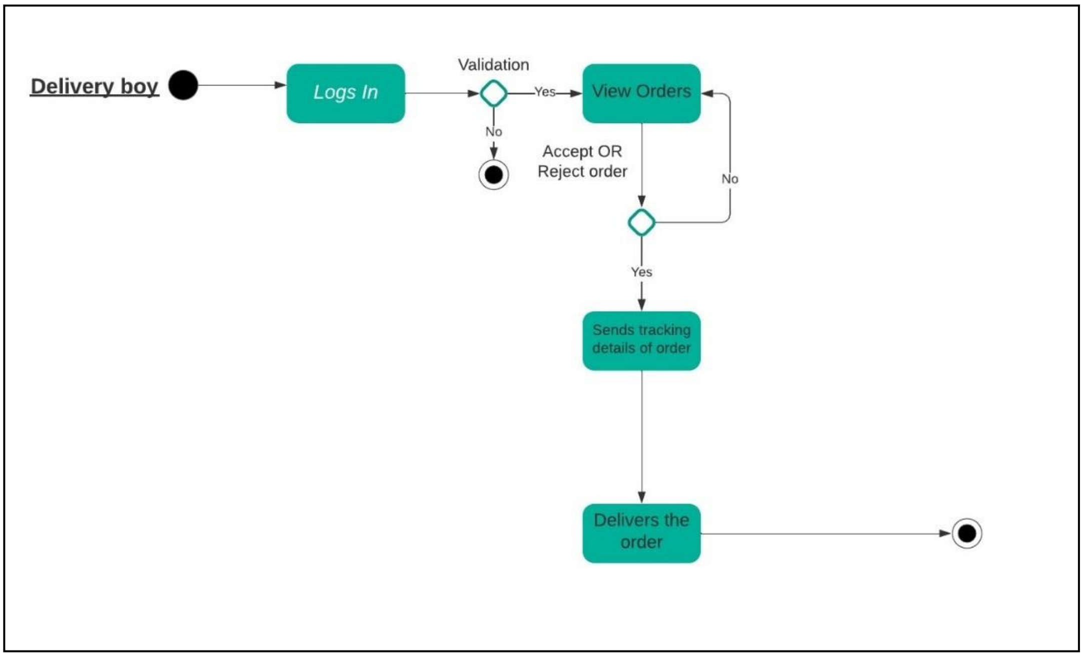


Fig 2.15. Delivery boy activity diagram

- Then proceeds to give the tracking details to the customer.
- Once the delivery is completed, he sends the feedback form to the customer.

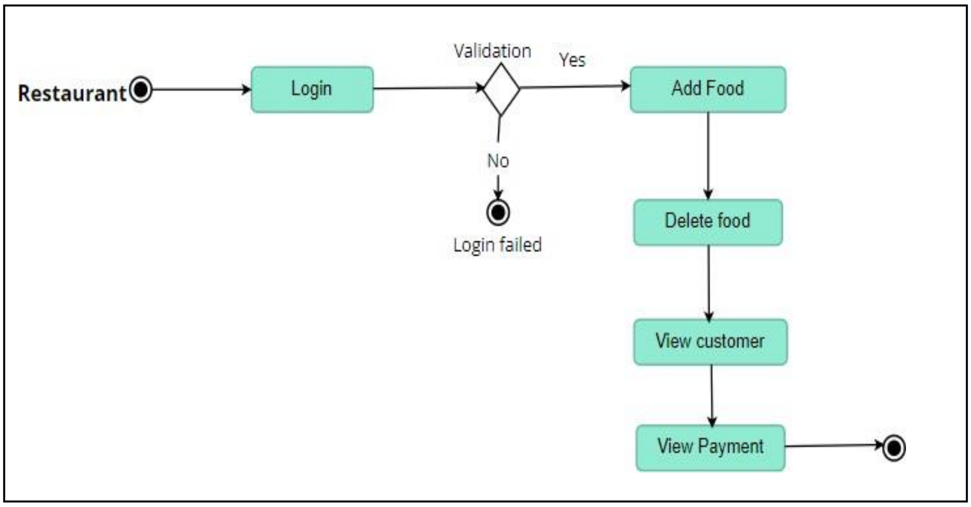


Fig 2.16. Restaurant activity diagram

The fourth activity diagram is for the restaurant module where more than one restaurant can log in, which shows the following steps:

- Firstly, the restaurant can log in and after successful validation can modify the changes.
- It can add, and delete food items from the cart.
- The restaurants can also view the information of the customer and their payment details.

2.8 Component Diagram

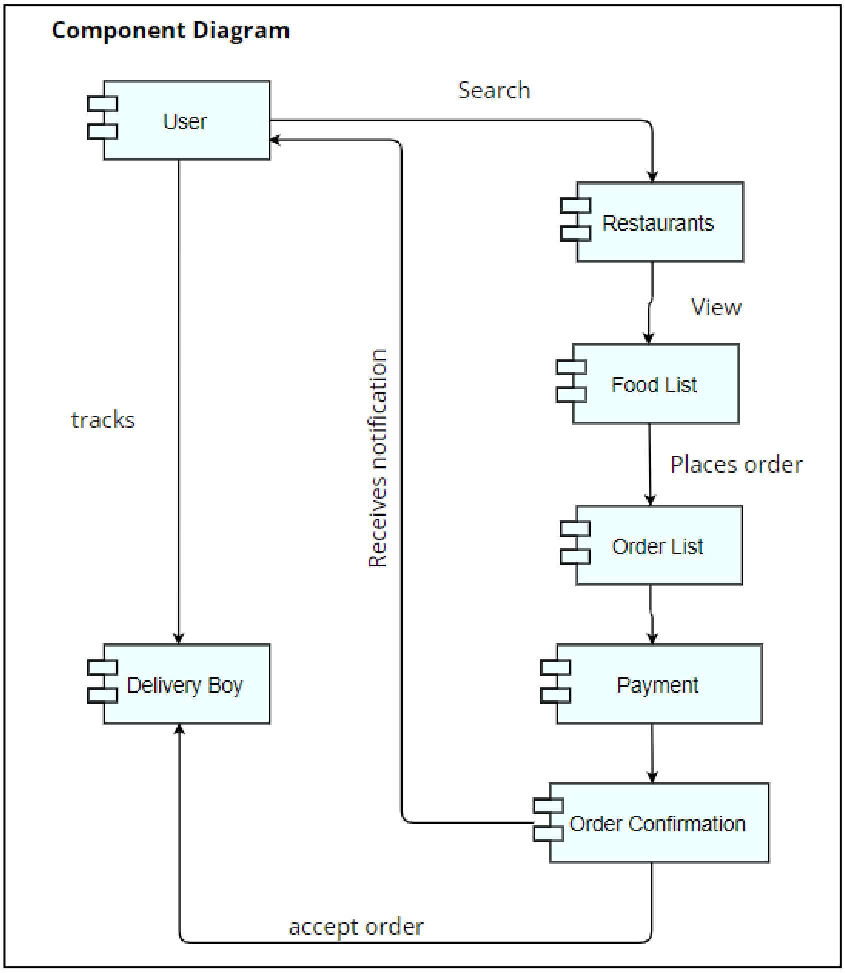


Fig 2.17 Component Diagram for the project

The component diagram is a special purpose diagram, which is used to visualize the static implementation view of a system. It represents the physical components of a system, or we can say it portrays the organization of the components inside a system. The components, such as libraries, files, executables, etc. are first needed to be organized before the implementation. The above diagram shows how the components coordinate with each other.

The above diagrams explain as follows:

- Firstly, the user orders the food selects it from the food list or the menu list, and adds it to the cart which consists of all the order list, orders the food and finally proceed to pay for the order.
- Once the payment is done the user gets the notification that the order is confirmed and at the same, the delivery boy receives the notification for delivering the order after the

acceptance of the order by the delivery boy the user can track the current status of their order.

2.9 Data Structure Diagram:

The class diagram shows how your system or subsystem is put together. Moreover, you can use class diagrams to model the objects that make up the system, show how the objects relate to each other, and explain what the objects do and what services they offer. It is the best way to show a system's structure in detail, including its parts, how they work, and how they relate to each other.

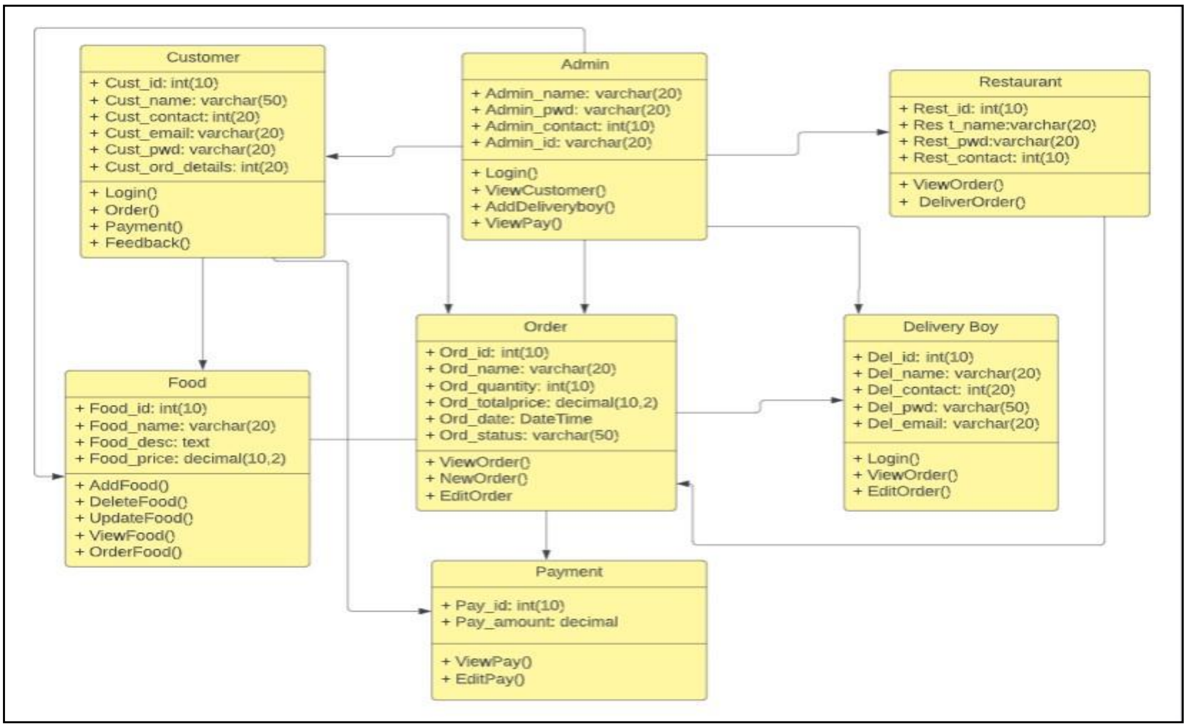


Fig 2.18 Data Structure/Class Diagram for the project

Online Food Ordering System describes the structure of system **classes**, their **attributes**, **methods**, and the relationships among objects. The customer orders online pay and then chooses whether to have their food delivered or picked up at your store. The ordering process is simple: your customer uses their mobile phone to browse the online menu.

- This diagram contains the classes for the application customer, admin, food, order, payment, and delivery boy.

- The attributes define the class information that must be filled in the application with the data type that is used in the database.
- Firstly, the admin can manage the details and view the details of the customer as well as the delivery boy and the food items that is been ordered by the customer.
- Whereas the customer is related to the food, order, and at the last proceed to payment. And lastly, the delivery boy accepts the order to deliver it.

2.10 Deployment Diagram

Deployment diagrams are used to visualize the topology of the physical components of a system, where the software components are deployed. They are used to describe the static deployment view of a system. It consists of nodes and their relationships. Using it you can understand how the system will be physically deployed on the hardware.

A UML deployment diagram is used by the application to show how the software should be deployed. It makes clear how links (nodes) talk to each other, which helps the project work the way it was designed to.

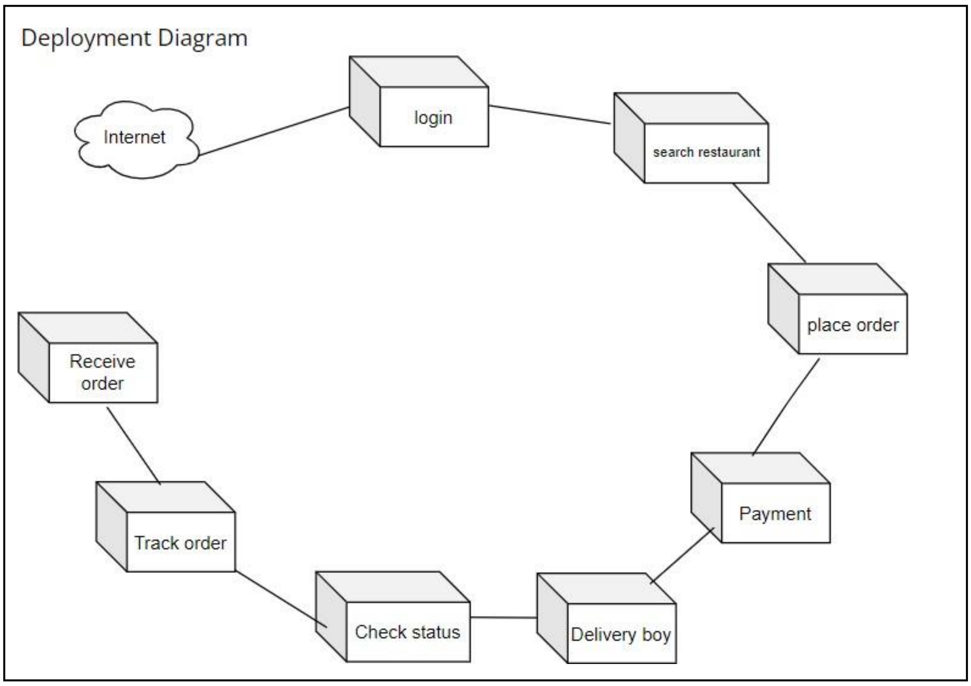


Fig 2.19. Deployment Diagram for the project

The above deployment diagram for the online food delivery application shows how software and hardware work together to make sure the online food ordering works right. The requirement of the internet helps the application work smoothly and finally, these data are stored in the database.

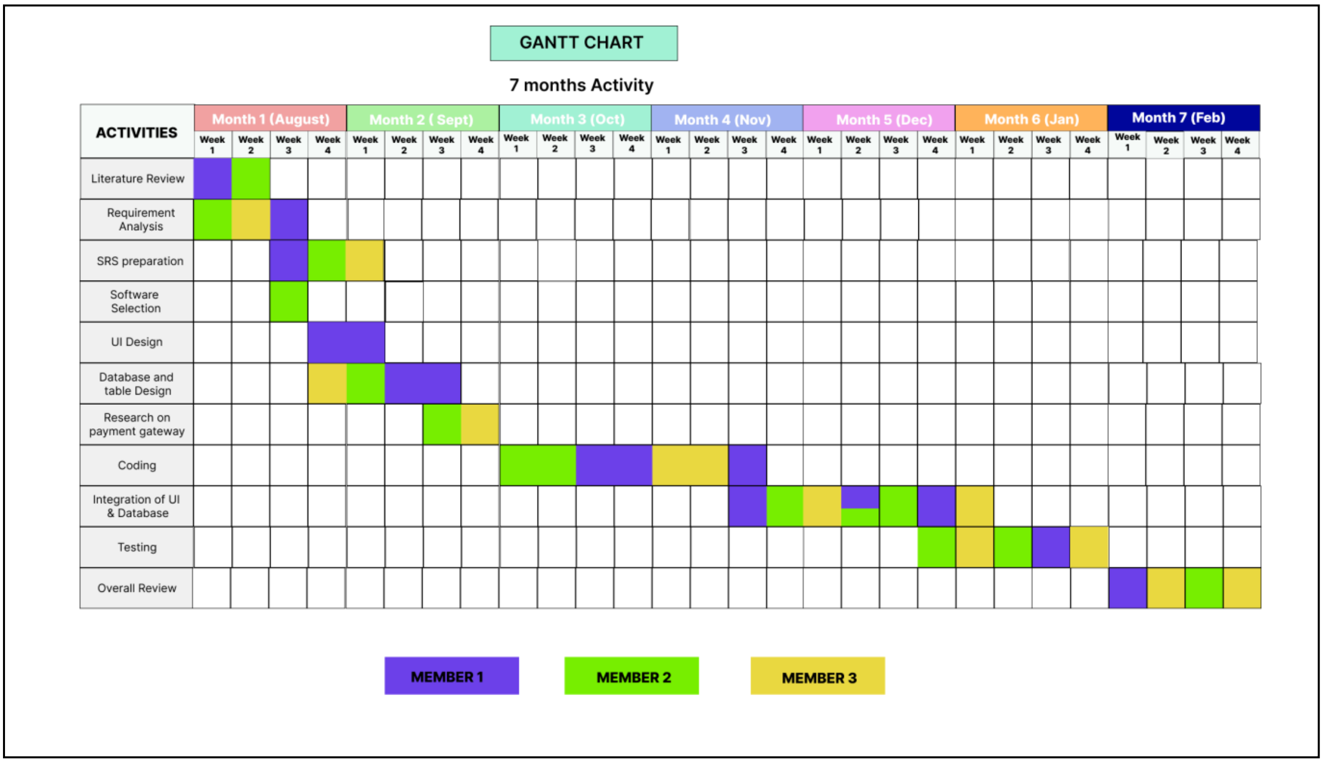
3

SYSTEM PLANNING AND ANALYSIS

3.1 Gantt Chart

A Gantt chart, commonly used in project management, is one of the most popular and useful ways of showing activities (tasks or events) displayed against time. On the left of the chart is a list of the activities and along the top is a suitable time scale. Each activity is represented by a bar; the position and length of the bar reflect the start date, duration, and end date of the activity. This allows you to see at a glance:

- What the various activities are
- When each activity begins and ends
- How long each activity is scheduled to last
- Where activities overlap with other activities, and by how much
- The start and end date of the whole project



GANTT CHART

7 months Activity

ACTIVITIES	Month 1 (August)				Month 2 (Sept)				Month 3 (Oct)				Month 4 (Nov)				Month 5 (Dec)				Month 6 (Jan)				Month 7 (Feb)			
	Week 1	Week 2	Week 3	Week 4	Week 1	Week 2	Week 3	Week 4	Week 1	Week 2	Week 3	Week 4	Week 1	Week 2	Week 3	Week 4	Week 1	Week 2	Week 3	Week 4	Week 1	Week 2	Week 3	Week 4	Week 1	Week 2	Week 3	Week 4
Literature Review	🟪	🟩																										
Requirement Analysis	🟩	🟨	🟪																									
SRS preparation			🟪	🟩	🟨																							
Software Selection			🟩																									
UI Design				🟪	🟪																							
Database and table Design				🟨	🟩	🟪	🟪																					
Research on payment gateway							🟩	🟨																				
Coding									🟩	🟩	🟪	🟪	🟨	🟨	🟪													
Integration of UI & Database															🟪	🟩	🟨	🟪 🟩	🟩	🟪	🟨							
Testing																				🟩	🟨	🟩	🟪	🟨				
Overall Review																									🟪	🟨	🟩	🟨

🟪 MEMBER 1

🟩 MEMBER 2

🟨 MEMBER 3

Fig 3.1 Gantt Chart for the project

In the first month which was August, we analysed the requirements and did the literature review. This activity went on for three weeks, next we proceeded to create the UI design over the span of the next 2 weeks. Initially we decided to go with the structured database, however later we decided to switch to Firebase, this decision was made as the result of the functionality firebase offers that allows one to implement various technologies into one project and also gives the ability to somewhat future proof the application. Payment being a critical and sensitive part of any system need to be efficient, secure, and available. To ensure these standards we met, we researched various payment gateways which could integrated which our application, after testing out a few services we decided to use razor pay due to its security features and it also had test mode option available, the test mode enables the team to test if the payment is processing and secure. After this stage was complete, we started with coding in the month of October, for this purpose we used Android Studio as it is the go to software for a project of this scale and chose Java language to program it as the development team has had prior experience with it. We integrated our application with Firebase as it is real-time. As we are using Incremental model for Software development, after completing every module we proceeded to test it. We performed black-box testing, we checked if there are any deviation from the output.

3.2 Project Activity Diagram

A Project Activity Diagram is a chart that shows the arrangement of tasks within a project. It can also contain other information such as the duration and connection between each task. This is an essential part of project management since you will be able to streamline the execution process.

Aside from the ones mentioned above, there are other benefits that you can get from using a Project network chart. With that being said, we listed down some of the benefits of using this diagram.

- Time-efficiency

This diagram shows the estimated time needed for a task to finish. Having a clear idea about the time it takes to complete a task, gives the project manager a clear view of the next steps needed to proceed with the task.

- Control over the task sequence

As a project manager, you are responsible for making sure that all steps are done in chronological order. This is where project network diagram examples come in. By using this tool, you can ensure that all necessary tasks are done without any complications to avoid any future issues.

- Task time management

There is no absolute way to avoid minor hiccups in any project. However, there are ways to minimize the effects of these hiccups is to know when to do the mending even before the actual problem arrives. Using a project network diagram and agile framework can help you predict upcoming issues, so you can immediately decide if you need more or less time for a single task.

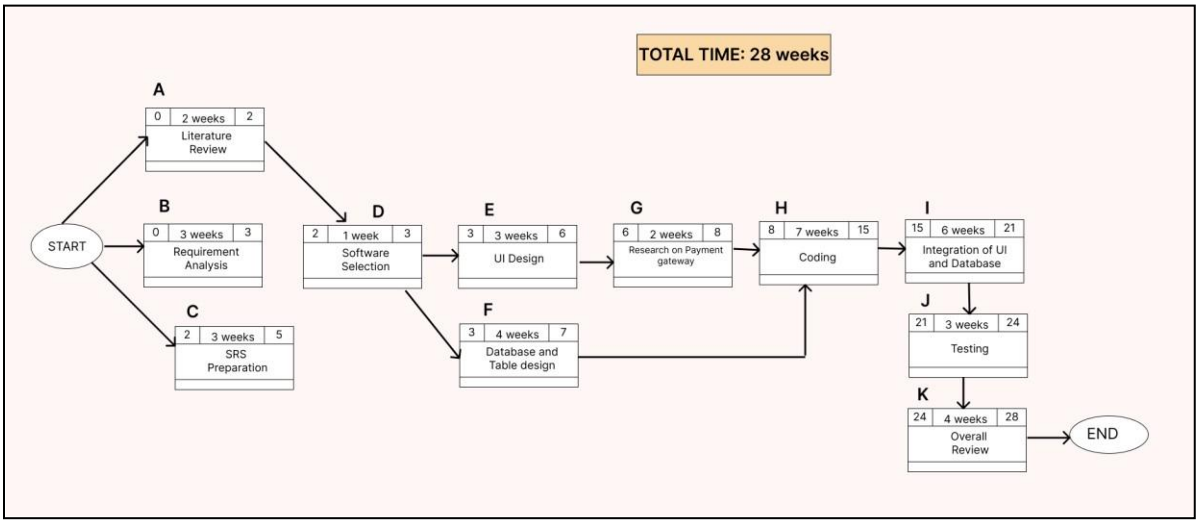


Fig 3.2. Activity diagram for the project

Activity	Description	Predecessor
0	Kick-off meeting (Start)	
A	Literature Review	**-**
B	Requirement Analysis	**-**
C	SRS Preparation	A, B
D	Software Selection	A
E	UI Design	D
F	Database And Table Design	D, E
G	Research on Chatbot and Recommendation System	F
H	Coding	G
I	Integration of UI and Database	H
J	Testing	I
K	Overall Review	J

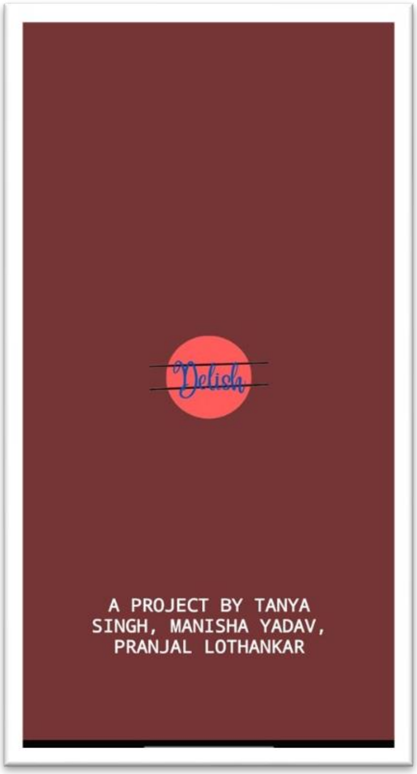
4

SYSTEM IMPLEMENTATION

➢ User Module

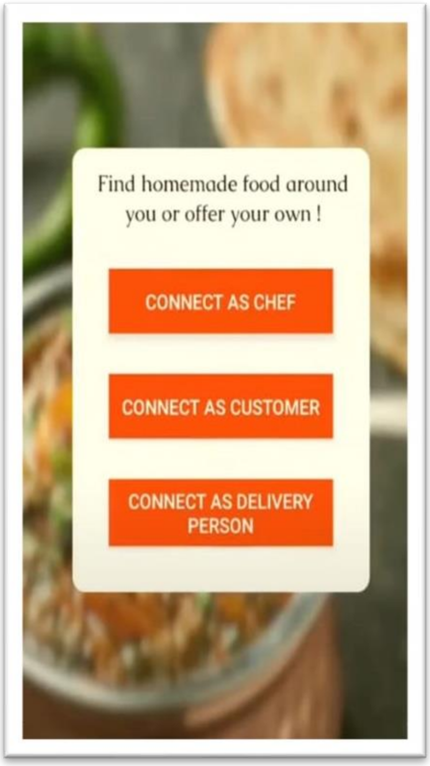
Splash Screen:

Android Splash Screen is **the first screen visible to the user when the application's launched**. Splash screen is one of the most vital screens in the application since it is the user'sfirst experience with the application.



A PROJECT BY TANYA SINGH, MANISHA YADAV, PRANJAL LOTHANKAR

Fig 4.1: Splash Screen



Find homemade food around you or offer your own !

Fig 4.2: User Connection Page

Registration Page:

When the user is has chosen the customer option, they will be guided to the login page, if user has not registered before they can link on registration link then they must enter details such as name, email, phone number and address.

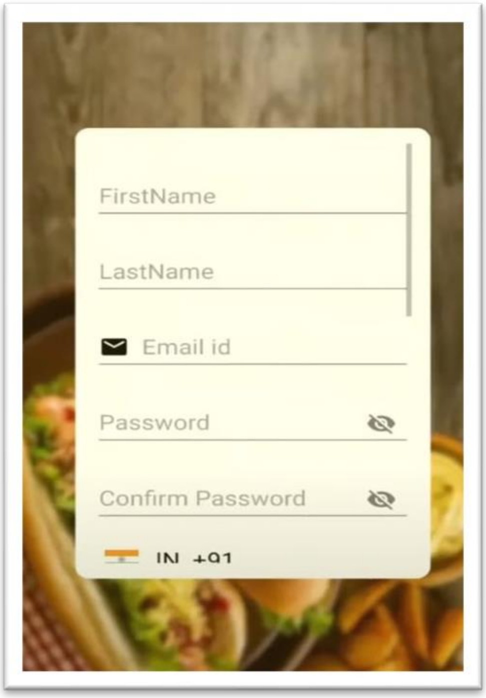
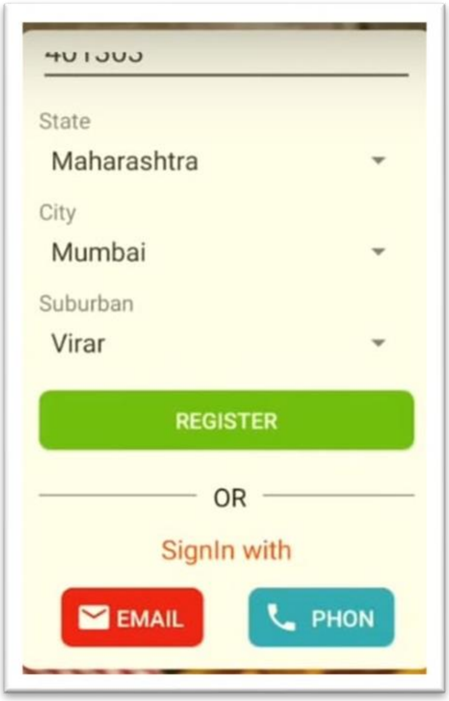


Fig 4.3: Registration Page (1)



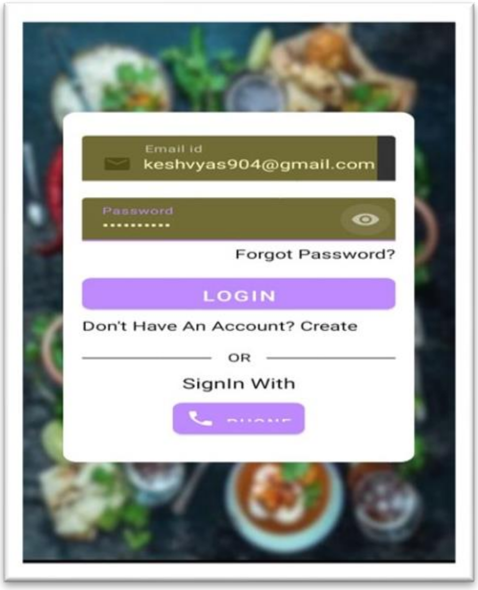
401303

State
Maharashtra

City
Mumbai

Suburban
Virar

Fig 4.4: Registration Page (2)



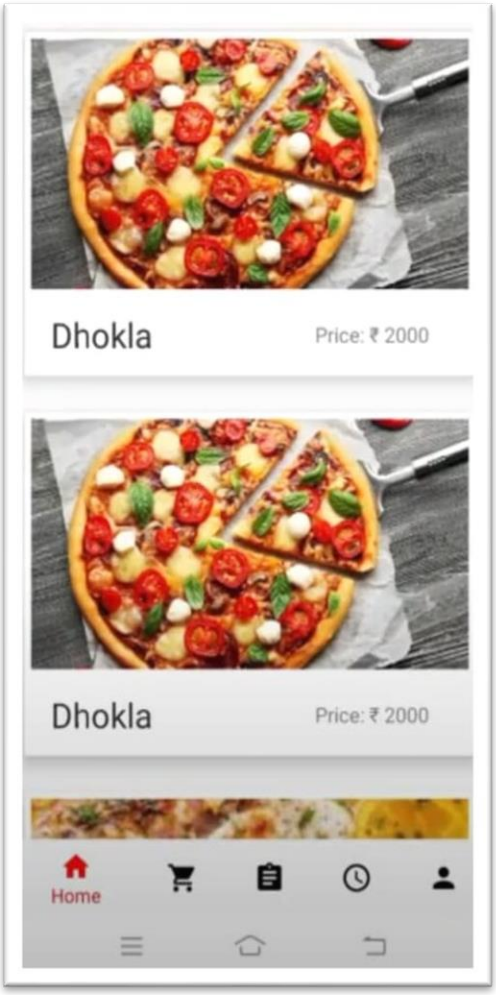
Email id
keshvyas904@gmail.com

Password
••••••••••

Fig 4.5: Login

Dashboard:

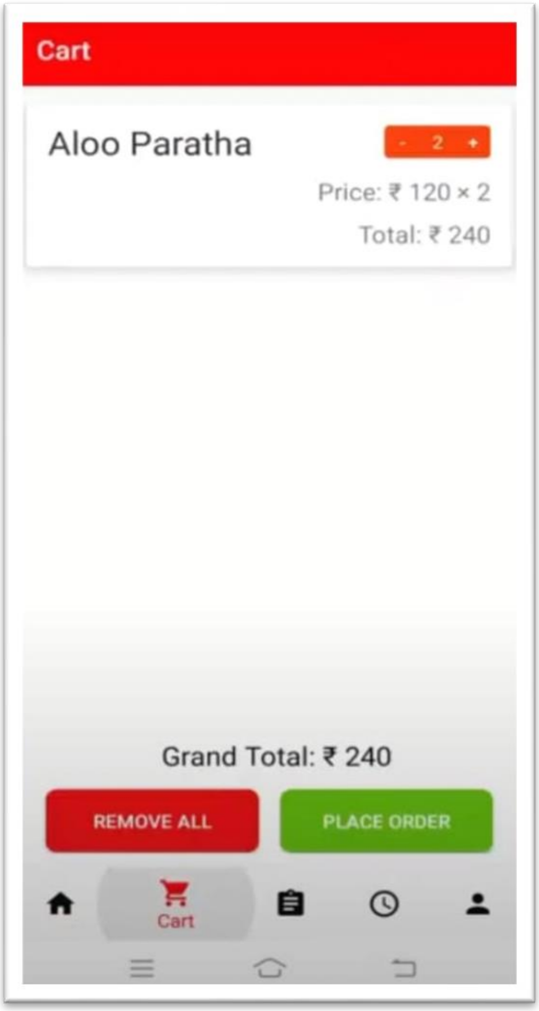
After login page user can see the dashboard, there he can browse the food items either by selecting a restaurant or by selecting a food category. Each restaurant has their own menu from which users can choose.



Dhokla Price: ₹ 2000

Dhokla Price: ₹ 2000

Fig .4.6: Customer’s Home Page



Cart

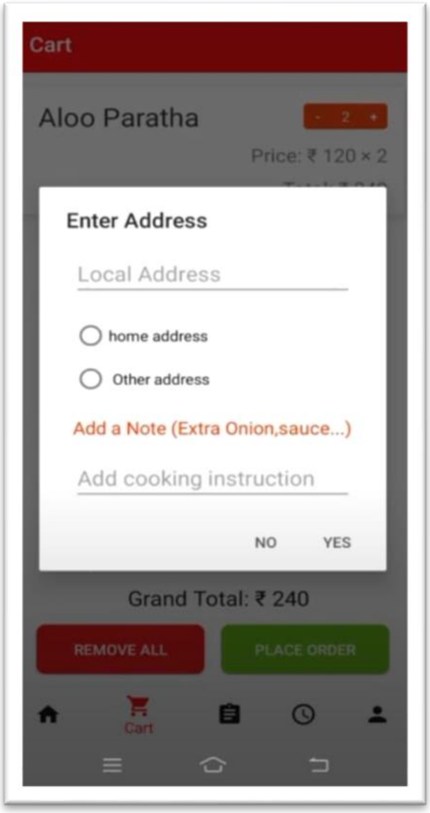
Aloo Paratha - 2 +

Price: ₹ 120 × 2

Total: ₹ 240

Grand Total: ₹ 240

Fig 4.7: Food Ordering Page (Customer)



Cart

Aloo Paratha

Price: ₹ 120 × 2

Enter Address

Local Address

○ home address

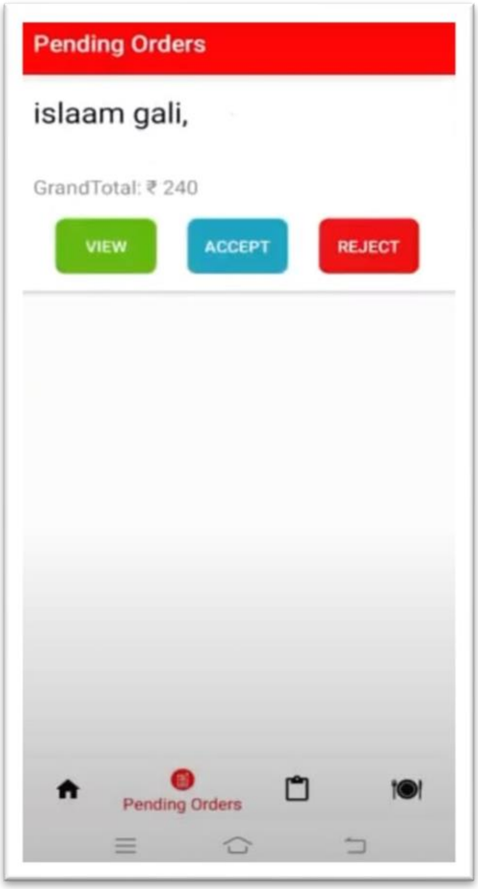
○ Other address

Add a Note (Extra Onion,sauce...)

Add cooking instruction

Grand Total: ₹ 240

Fig 4.8: Addressing Page

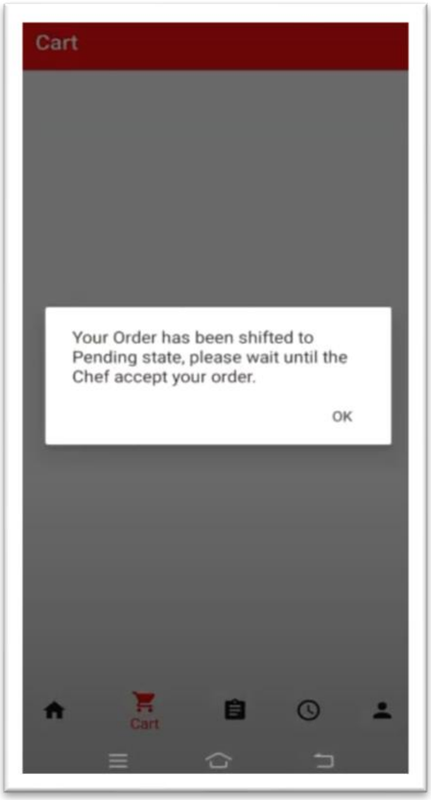


Pending Orders

islaam gali,

GrandTotal: ₹ 240

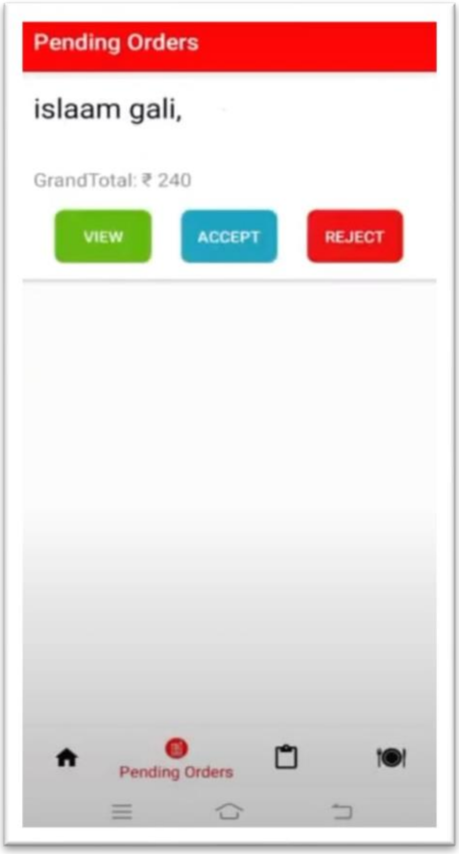
Fig 4.9: User Connection Page



Cart

Your Order has been shifted to Pending state, please wait until the Chef accept your order.

Fig 4.10: Order Pending Page

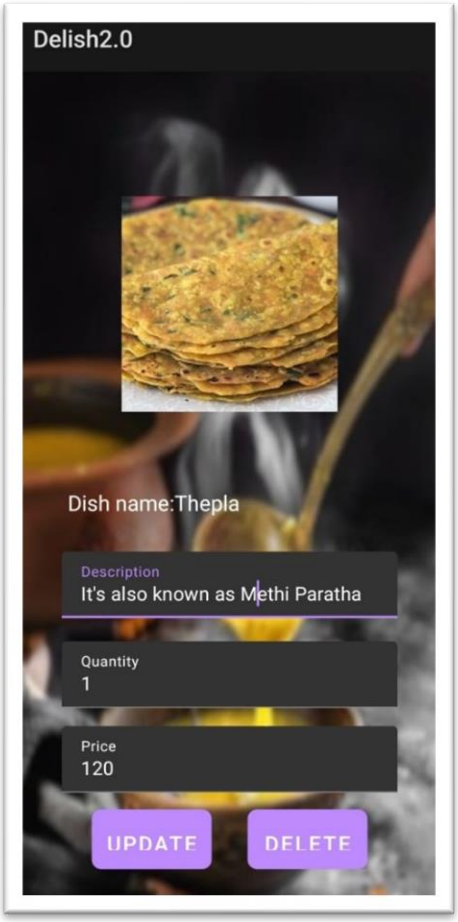


Pending Orders

islaam gali,

GrandTotal: ₹ 240

Fig 4.11: Home Page of Rider



Delish2.0

Dish name:Thepla

Description
It's also known as Methi Paratha

Quantity
1

Price
120

Fig 4.12: Modification Page

5

COST-BENEFIT ANALYSIS

5.1 Cost Estimation

Software Cost Estimation is a process to predict and estimate the approximate cost of the software project before the development starts. It is one of the vital processes to start development for software by considering all internal & external cost factors. Cost estimation is a tool to estimate the planning, budgeting, and resource utilization for software projects. Before cost estimation for a software project, we will have to know what are the actual requirements for a project, what is the complexity of those requirements, and other cost driver factors that affect the development.

The cost factors of the project can be included with the responses of three factors:

i. Effort: The amount of effort required to complete the development of software projects in terms of Man-Months (MM).

ii. Development duration: The time taken to complete the development of the project which is the total development time.

iii. Resources: The number of Manpower required for a software project in terms of time to complete.

Various factors affect the cost of app development. When considering the food delivery application cost of development is affected by various factors like the platform, features to be included, and most importantly, the region of food delivery.

For our project, we have estimated the following function points and the total function points are:

Function Point	**Count**		**Simple**	**Average**	**Complex**		
Number of External Inputs (EI)	13	*	3	*4*	6	=	**52**
Number of External Output (EO)	6	*	4	*5*	7	=	**30**
Number of external inquiries (EQ)	3	*	3	*4*	6	=	**12**
Number of internal files (ILF)	7	*	7	*10*	15	=	**70**
Number of external interfaces (EIF)	3	*	5	*7*	10	=	**21**
Count Total						=	**185**

Table 5.1 Function Point Analysis

Here that weighing factor will be simple, average, or complex for a measurement parameter type.

The Function Point (FP) is thus calculated with the following formula

FP = Count-total * [0.65 + 0.01 * ∑(fi)]

= Count-total * CAF

where Count-total is obtained from the above Table.

CAF = [0.65 + 0.01 *∑(fi)]

and ∑(fi) is the sum of all 14 questionnaires and shows the complexity adjustment value/ factor-CAF (where i ranges from 1 to 14).

Complexity Adjustment Factors (CAF):

SR NO.	Questions or Factors	Range
1	Data Communication	4
2	Distributed Data Processing	4
3	Performance	3
4	Heavily used configuration	3
5	Transaction Role	4
6	Online Data Entry	4
7	End-user Efficiency	4
8	Online updates	3
9	Complex Processing	3
10	Reusability	3
11	Installation Ease	3
12	Operational Ease	4
13	Multiple Sites	2
14	Facilitate Change	4

Table 5.2 Complexity adjustment factors

Estimated Scale= 3

Fi = 14*3=42

CAF = [0.65 + 0.01 *∑(fi)]

= [0.65+0.01*42]

= 1.07

Now, **Function Point = 185 * 1.07 = 197.95**

Cost Estimation

Cost-benefit analysis (CBA) is a technique used to compare the total costs of a programme /projectwith its benefits, using a common metric. This enables the calculation of the net cost or benefit associated with the programme.

COCOMO stands for Constructive Cost Estimation Model. It is one of the most generally used software estimation models in the world. It predicts the efforts and schedule of a software productbased on the size of the software.

A basic COCOMO model of the system based on ORGANIC MODE will be used for the costestimation.

A software project is said to be an ORGANIC type if the team size required is adequately small, the problem is well understood and the team members have a nominal experienceregarding the problem.

BASIC COCOMO can be used for quick and slightly rough calculations of Software Costs. Its accuracy is somewhat restricted due to the absence of sufficient factor considerations.

- **STEPS TO CALCULATE COST ESTIMATION**
  - ❖ The basic initial estimate for development effort is represented as: -

Effort=a*(KLOC)b PM Tdev = c*(efforts)d Months

KLOC is the estimated size of the software product indicate In Thousand Lines of Code,

a, b, c, d are constants for each group of software products,

Tdev is the estimated time to develop the software, expressed in months,

Effort is the total effort required to develop the software product, expressed in person months (PMs).

Estimated LOC = 2500,

KLOC=2.5

Constants for the project: a=2.4, b=1.05

c=2.5, d=0.38

- **Estimation of development effort**

Effort = a*(KLOC)b

Effort = 2.4*(2.5)$^{1.05}$

= 6.28 PM

- **Estimation of development time**

Tdev = 2.5*(6.28)$^{0.38}$

= 7.0 Months

- **Estimation of Development Cost:**

Total Time Required to Develop the project = 5.0 Months

= 7.0 *30

= 210 days

Cost per day: 100 Number of developers:3

Total cost =7.0*30*100 =21,000

Total cost = <u>Rs. 21,000</u>

SYSTEM TESTING

6.1 Methodology used for testing

In this project we have used the following methodologies for testing purposes:

- White Box Testing: A software testing method in which the internal structure/design/implementation of the item being tested is known to the tester.
  - So here we the group members have tested the code and checked if the codes are running properly as they should.
  - Each file of the project was tested by us.
  - Also checking the inputs and outputs.
  - We have also tested the database of firebase to verify the nodes
  - Online Food Delivery Application uses the 'Equivalence Partitioning' technique for the purpose of unit testing of the project. Equivalence Partitioning is a software testing technique that divides the input data of a software unit into partitions of data from which test cases can be derived. In principle, test cases are designed to cover each partition at least once. This technique tries to define test cases that uncover classes of errors, thereby reducing the total number of test cases that must be developed.

- Black Box Testing:

Black box testing is a type of software testing in which the functionality of the software is not known. The testing is done without the internal knowledge of the products.

- Manual Testing:

A method of testing whereby software is tested manually (by a human). Manual testing is a crucial aspect of the software testing process, and it plays an essential role in ensuring the quality of the food delivery app.

Manual testing is the process of verifying the functionality of an application manually, without the use of any automated tools or scripts. In the case of the food delivery app, manual testing is carried out by our team acting as testers by simulating the various scenarios that users may encounter when using the app. The goal of manual testing is to identify any defects or issues in the app's functionality and user interface, as well as to ensure that the app meets the user's expectations and requirements.

6.2 Data Used for Test Cases:

The proper selection of the data is very important. If the test data is not appropriate or representative of the data to be provided by the user, the result of the test case becomes unreliable.

- Using Artificial Test Data - Live data is difficult to obtain and sometimes is insufficient to conduct extensive testing. It does not test all possible combination or formats. Therefore, artificial test data was used at the time of unit testing. Artificial test data was created solely for test purposes which provide extreme values for testing the limit of the system.

6.3 Test cases

SR No.	Test Case	Steps	Expected	Result
1.	Registration	Fill in the name, email, and password field	Data gets stored in the database and customers get a successful message.	Successful
2.	Login	Fill the credentials	The credentials get verified and the user gets logged in	Successful
3.	Usability	Clicking buttons and various navigations	All buttons should function correctly	Successful
4.	Add to cart button	Click on add to cart button	The product gets added to the cart and pops a successful message	Successful
5.	Buy now button	Click on the buy now button	The user gets redirected to the payment process page	Successful
6.	Add Address button	Click on add address button	The user gets redirected to create an address page	Successful
7.	Payment button	Click on continue payment button	The user gets redirected payment detailed page	Successful
8.	Check out button	Click on check out button	The user will be directed to the Razorpay payment page	Successful

9.	Quantity button of products	Click on the (+) and (–) quantity buttons	Increase in the quantity of the particular product	Successful
10.	Navigation (Category, new products, popular products)	Clicking on the menu to navigate	All the menus should be clickable	Successful
11.	Logout button	Clicking on the logout button	The user will be redirected out of the application	Successful
12.	Security		Receiving mail after sign	

Table 5.3 Test Cases

7

SYSTEM MAINTENANCE AND EVALUATION

7.1 System Maintenance

System maintenance can be considered on the following basis:

- **Managing Push Notifications**

Enabling push notifications is one of the effective ways to keep the users engaged with the application while resulting in improved user engagement. However, if you need to integrate some additional push notification features into your food delivery app, hire mobile app maintenance service experts. This will help you manage and personalize push notifications based on users' engagement and future business goals.

- **Bug Fixing & Updates**

None of the businesses will launch a mobile app with bugs and errors. But, with time, users may find some error in the features and functionalities of a food delivery application that needs to be updated with time. In this situation, it becomes essential to address those issues, and hence businesses need to perform mobile app maintenance.

- **New Content Upgradation**

Initiating the food delivery website development process doesn't end the job there. A lot more needs to be performed to keep the app running. This activity involves publishing new content as well. To ensure that the users get updated information from your food delivery app, update the content timely. It could be in the form of text, images, videos, and more. It will help you increase user engagement and earn huge business profits and sales.

7.2 System Evaluation

Similarly, the evaluation can be done as per the working of the food delivery application.

1. Consistency

Pick The Food keeps its consistency by using the same font type, color, layout, menu, and menu icon on each page. It applies to website applications and mobile applications.

2. Provide Universal Usability

Pick The Food interface is made user-friendly so it eases users to use the application. It can be seen in the Back button provided in both the customer side and courier side application which is using the universal icon and is similar to the user.

3. Design the Final Dialog

Pick The Food application is made to ease users in understanding their actions while using it. For example, in the customer application, there is a final dialog confirmation in the setting interface while users try to edit their profile.

4. Reversal Action Allowance

Pick The Food application has integrated the simple reversal action. For example, there is a Back button in almost all interfaces for both the customer side and courier side applications.

8

USER MANUAL

With our Online Food Delivery Application, you do not have to call the restaurant to order food, you can simply browse through the menu on the app and place an order directly. You can also make payments online and can track the status of the order. The app also has a GPS by which the users can track their delivery and the details of the delivery man. This app is also beneficial for the owner as they can expand their businesses and gain customers from a wider area and create a brand image for themselves. This application also allows the restaurant owners to add the food items themselves. They can also modify their restaurant menu as they can update or delete the food items.

Here is a quick guide to get started with our application:

- When you first open our application, you can see the splash screen, this screen will only visible for few seconds then you navigated to the next page.

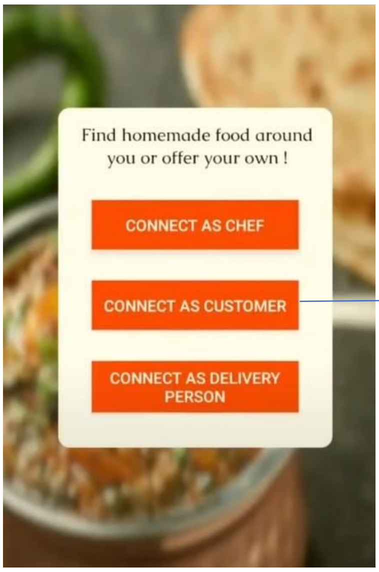
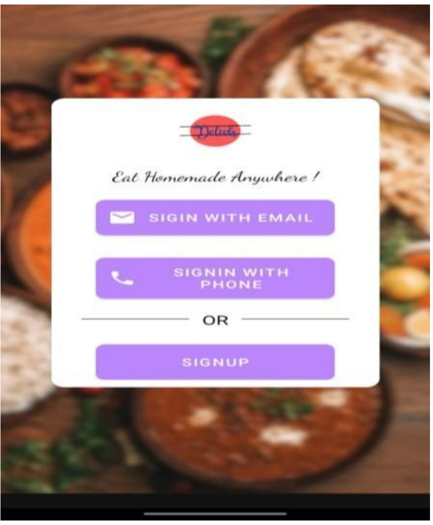

Find homemade food around you or offer your own !
CONNECT AS CHEF
CONNECT AS CUSTOMER
CONNECT AS DELIVERY PERSON

Fig 8.1: Sign-up Option

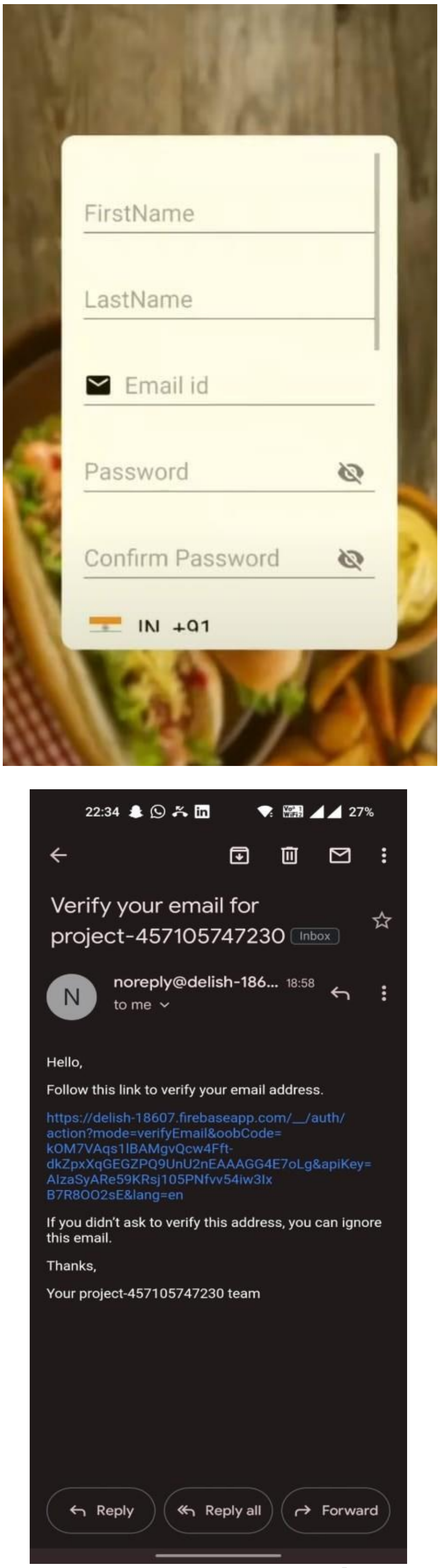


Sign-in Option

- Next, you are given three sign-in options, you can choose to login either as customer, restaurant owner or as a delivery person. Kindly chose any one these to proceed further.

Guide for Customer:

- If you have chosen the customer option, you can sign in using your valid email and password which you have registered with before. If you are new you can click Sign Up option, to create an account.



Verify your email for project-457105747230

noreply@delish-186... 18:58

to me

Hello,

Follow this link to verify your email address.

https://delish-18607.firebaseapp.com/__/auth/
action?mode=verifyEmail&oobCode=
kOM7VAqs1lBAMgvQcw4Fft-
dkZpxXqGEGZPQ9UnU2nEAAAGG4E7oLg&apiKey=
AIzaSyARe59KRsj105PNfvv54iw3Ix
B7R8OO2sE&lang=en

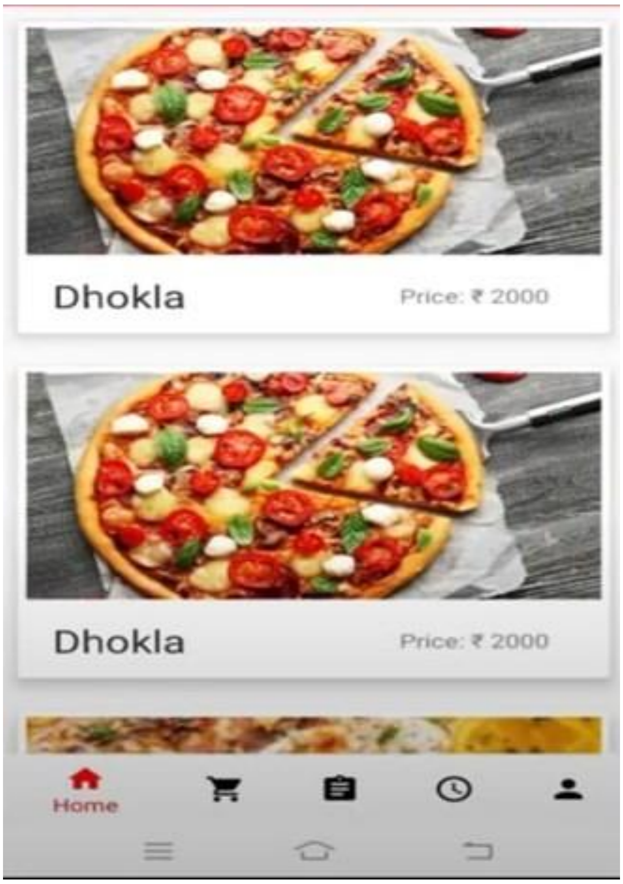
If you didn't ask to verify this address, you can ignore this email.

Thanks,

Your project-457105747230 team

Fig 8.2: User Guide

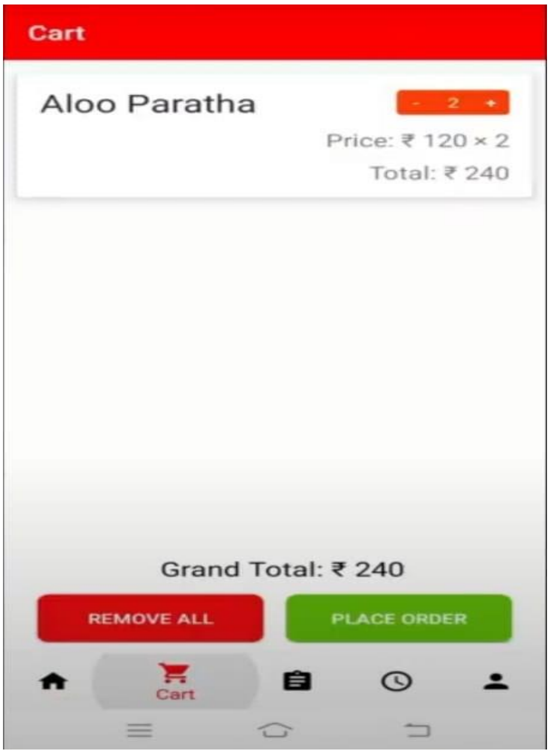
- If you want to sign up, you must enter your name, email address, phone number and your address to register yourself. Once you have entered click on the Sign-Up Button.
- You will receive an email from the app asking you to click on link to verify your email address, this is important step without this you would not be able to sign-in later.
- After the register to the login page and enter the email and password and click on login. After your email and password has been validated you will be led to dashboard page.
- On the dashboard page you can see the list of restaurants and food items by category. You select the food item by choosing a restaurant or you can select food by category.
- You can view the food details by click on food item from the menu, it will show details about the food including price and can click on add to cart button to add food to your cart.


Dhokla Price: ₹ 2000

Dhokla Price: ₹ 2000

Fig 8.2: User Guide

- To view the cart, you can simply click on the cart icon visible on the dashboard, by clicking on the cart you can see all the food items you have selected. You can click on place order.



Cart

Aloo Paratha

Price: ₹ 120 × 2

Total: ₹ 240

Grand Total: ₹ 240

- You can then see your order status.

Guide for Restaurant Owner:

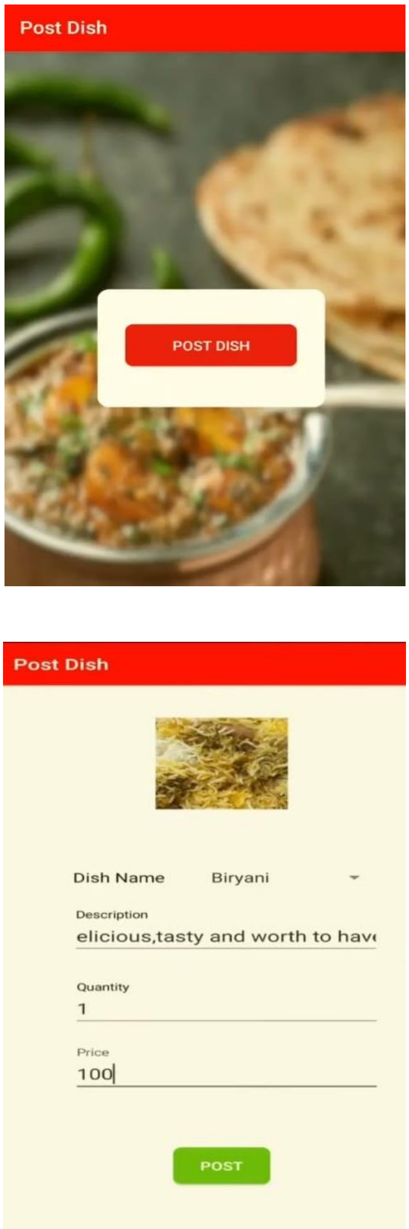
• If you select the option as restaurant owner for the login purpose then you have to enter your mobile number and email and then you will receive a message that your account has been created.

• Using the email id and the password then further you can login to the application.

• Here the restaurant owner can add the restaurant and within it add multiple food items based on its categories. They can also differentiate the food items based on veg and non-veg.

• The restaurant owner also has options to add items, delete and update them. They can mention the price of the items as per their decision.

• The restaurant owner can also add their location to the application.



Dish Name Biryani

Description
elicious,tasty and worth to hav

Quantity
1

Price
100

Fig 8.3: Restaurant owner Guide

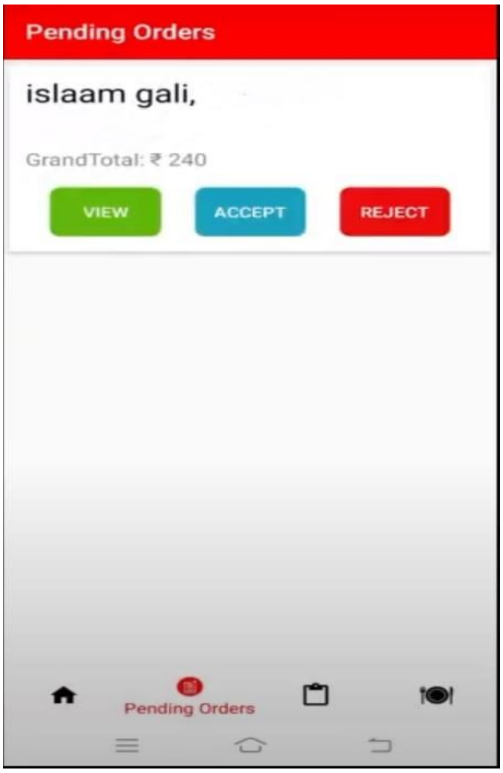
Multiple images can be added to the menu page to provide attractive screen for the customers.

• Similarly multiple restaurants can be added which provides various selecting options for the customers.

Guide for Delivery Boy:

- If you select the third option that is the delivery boy, then you must enter your mobile number and email id for creating the account in case you are new to the application.
- Further after successfully login to the application the delivery boy can see the all the list of the pending orders.

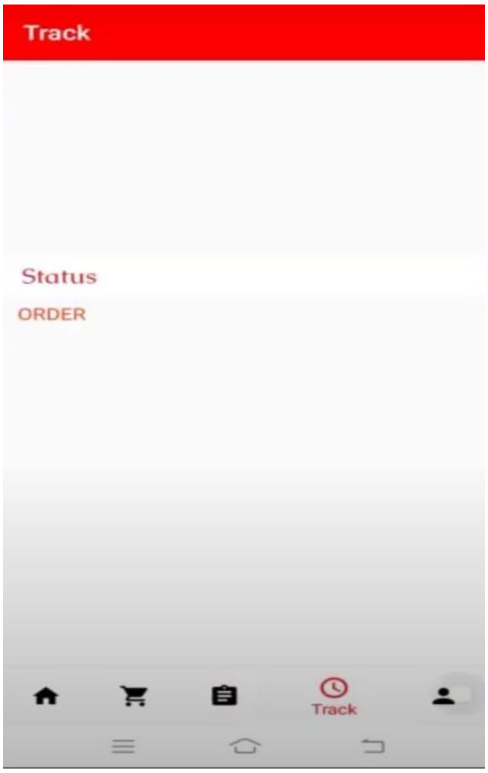
- It will show three options such as view, accept and reject. He can also view the location of the users for delivering the food.



Pending Orders

islaam gali,

GrandTotal: ₹ 240



Track

Status

ORDER

Fig 8.4 Delivery Boy Guide

- If the reject option has been chosen then it will delete that order from the list.
- If the delivery boy selects the view option, he can view the list of the orders and then accept among it.
- Also, he can view the total price of the food items. As per the mode of payment the delivery boy can collect the payment.
- After the accepting the order, the user can track or view the location of the delivery boy.
- The delivery boy can also log out from the page whenever required and stay updated with the orders received.

9

CONCLUSION AND SCOPE FOR FUTURE WORK

9.1 Future work

In recent times, the market for food ordering applications has grown exponentially. As a result of the comfort and connivence offered by food ordering applications more and more people are switching to this modern digital method from the classic methods.

As the demand grows, it is necessary to improve upon the applications in the future to enhance its functionality and make them better, the following are a few methods:

- A recommendation system based on past orders to suggest user restaurants or dishes they may like
- Can have an option to “order my daily menu”. So that the customer will not have to invest his time to add his regular dishes to the cart.
- Similarly, the option to order certain items automatically or at pre-set intervals can be added.
- Functionality to order groceries can be implemented.
- Functionality to order food and pick it up yourself from the restaurant.
- It can have a feature to upload the image of the dish to find that particular dish in the menu easily.
- The ability to search for specific items with photos or images can be implemented with the help of AI&ML
- It can also have the feature to search by voice.
- Integrating the application with AI assistants such as Google or Alexa to be able to place orders or receive updates through these AI assistants.

9.1 Conclusion

In conclusion, the food delivery app documentation presented here provides a comprehensive guide to the features and functionality of the app, as well as the development process and user experience. By following this documentation, users can easily understand how to use the app and its various features, from browsing restaurants and menus to placing orders and tracking delivery status. The documentation also provides useful information for developers and other stakeholders, including an overview of the app's architecture, data models, and UI. Overall, this documentation serves as a valuable resource for anyone involved in the development, testing, or use of the food delivery app, and it will help ensure a smooth and enjoyable experience for all users.

The documentation covers every aspect of the app, from the user interface and navigation to the backend server and database structure. By following the guidelines outlined in this documentation, developers will be able to create a high-quality and user-friendly food delivery app that meets the needs and expectations of our customers. We believe that this documentation will be a valuable resource for all developers involved in the project and provide a seamless and convenient food delivery experience to our users.

REFERENCES

[1] https://www.researchgate.net/publication/350430883_ONLINE_FOOD_ORDERING_SYSTEM

[2] Online food ordering app documentation, Online food ordering app documentation (slideshare.net)

[3] Software Engineering | Calculation of Function Point (FP) - GeeksforGeeks

[4] RESTAURANT ORDERING SYSTEM By Wong Siew Jiuan, fyp_IA_2019_WSJ_1506513.pdf (utar.edu.my)

[5] Software Engineering | Extended Function Point (EFP) Metrics - javatpoint

[6] Project Management Network Diagram Complete Guide | EdrawMax (edrawsoft.com)

[7] What is a Gantt Chart? Gantt Chart Software, Information, and History

[8] User Manual for Delivery App, User Manual for Delivery App (fos365.com)

[9] What is Test Methodology? (With 7 Methodologies) | BrowserStack

[10] Template_1-copy.pdf, Cloud Tutorial

[11] Software Testing | Basics - GeeksforGeeks