**Student Voting System**

**A Project Report**

Submitted in partial fulfillment of the

Requirements for the award of the Degree of

# BACHELOR OF SCIENCE (INFORMATION TECHNOLOGY)

**By**

Sahil Chandrakant Pawar

Seat Number - \_\_\_\_\_\_\_

**Under the esteemed guidance of**

**Mrs. S. S. Ketkar**

# Assistant Professor at Information Technology Department of Gogate Jogalekar College (AUTONOMOUS), Ratnagiri



**DEPARTMENT OF INFORMATION TECHNOLOGY**

# GOGATE JOGALEKAR COLLEGE(AUTONOMOUS) , RATNAGIRI

***(Affiliated to University of Mumbai)***

**RATNAGIRI, 415612**

**MAHARASHTRA**

**2024-25**

# PROJECT PROPOSAL

PRN No. : - Roll No. : - TTA240029

1. Name of the Student: - Sahil Chandrakant Pawar .

1. Title of the Project :-Student Voting System .

1. Name of the Guide: - Mrs. S. S. Ketkar.

1. Teaching experience of :- 8 years

1. Is this your first Submission? :- Yes/No

# GOGATE JOGALEKAR COLLEGE (AUTONOMOUS), RATNAGIRI

***(Affiliated to University of Mumbai)***

**RATNAGIRI-MAHARASHTRA-415612**

## DEPARTMENT OF INFORMATION TECHNOLOGY



**CERTIFICATE**

This is to certify that the project entitled, **"** **Student Voting System "**, is bona fide work of **Sahil Chandrakant Pawar** bearing Seat. No: (**\_\_\_\_\_\_\_\_\_\_**) submitted in partial fulfillment of the requirements for the award of degree of BACHELOR OF SCIENCE in INFORMATION TECHNOLOGY from University of Mumbai.

**Internal Guide Coordinator / HOD**

**External Examiner**

**Date: College Seal**

# ABSTRACT

The "Student Voting System" is a platform designed to make the voting process in colleges faster, easier, and more secure. It replaces the traditional manual voting method with an automated system that ensures only eligible students, verified by their college ID, can vote. This helps prevent unauthorized voting and ensures fairness in the process.

One of the key features of the system is that each student can only vote once per event. The system tracks student roll numbers, ensuring no one can vote multiple times. This removes the risk of duplicate voting and improves the integrity of the election results.

After voting, students immediately receive a confirmation message on their mobile phones. This message shows the event name and the candidate they voted for, giving students confidence that their vote was recorded correctly and securely.

The system also provides a simple, easy-to-use interface for both the voting panel and the administrators. The admin can create and manage voting events, add or remove candidates, and monitor the progress of the voting in real time, making the process more efficient and well-organized.

In summary, the "Student Voting System" ensures a secure, reliable, and user-friendly voting experience for students and administrators. It aims to simplify the voting process, prevent errors, and promote transparency, making it ideal for college events and elections.

**ACKNOWLEDGEMENT**

I would like to express my sincere gratitude to several individuals for supporting me throughout this project.

I am especially thankful to Dr. M. R. Sakhalkar, Principal, and Dr. A. M. Kulkarni, Vice Principal (Science Faculty), along with Coordinator Dr. V. V. Bhide of Gogate Jogalekar College, Ratnagiri, for providing all the necessary facilities in the laboratory and library.

I wish to express my heartfelt thanks to Mrs. M. A. Sahasrabuddhe, Head of the Department of Information Technology, for granting me the opportunity to complete my project and for her valuable guidance throughout the process.

I would also like to extend my gratitude to our respected teachers for their constant support and guidance during the finalization of my project.

A special thanks goes to my guide, Mrs. S. S. Ketkar, for her enthusiasm, patience, and practical advice, which greatly contributed to the successful completion of my project.

Last but not least, I would like to thank my parents, friends, and everyone who encouraged and supported me in completing this project on time.

Name of the Student

Mr. Sahil Chandarakant Pawar

# DECLARATION

I hereby declare that the project entitled, “**Student Voting System** ” done at **Gogate Jogalekar College, Ratnagiri**, has not been in any case duplicated to submit to any other university for the award of any degree. To the best of my knowledge other than me, no one has submitted to any other university.

The project is done in partial fulfillment of the requirements for the award of degree of

BACHELOR OF SCIENCE (INFORMATION TECHNOLOGY) to be submitted as final semester project as part of our curriculum.

Mr. Sahil Chandaraknt Pawar

# TABLE OF CONTENTS

|  |  |  |
| --- | --- | --- |
| **Sr. No.** | **Content** | **Page No.** |
| 1 | **Introduction** | 10 |
| 1.1 Background | 10 |
| 1.2 Objectives | 11 |
| 1.3 Scope, Purpose, Applicability | 12 |
| 1.3.1 Scope | 12 |
| 1.3.2 Purpose | 12 |
| 1.3.3 Applicability | 12 |
| 2 | **Survey Of Technologies** | 13 |
| 2.1 Existing System | 13 |
| 2.1.1 Limitation | 13 |
| 2.2 Proposed System | 13 |
| 2.2.1 Advantages | 14 |
| 3 | **Requirement Analysis** | 15 |
| 3.1 Problem Definition | 15 |
| 3.2 Requirement Specification | 15 |
| 3.3 Planning and Scheduling - Gantt Chart | 16 |
| 3.4 Hardware and Software Requirements | 16 |
| 3.5.1 UML Diagrams | 17 |
| 4 | **System Design** | 22 |
| 5.1 Module Division | 22 |
| 5.2.1 Data Dictionary | 23 |
|  | 5.2.2 Schema Design | 24 |
| 5.3 Event Table | 27 |
| 5.4 Test Case | 28 |
| 5.5 User Interface | 31 |
| 5.6 Code | 37 |
| 5.7 Coding and Implementation – Gantt chart | 46 |
| 6 | **Conclusion** | 47 |
| 7 | **Bibliograph** | 48 |

# LIST OF DIAGRAMS

|  |  |  |
| --- | --- | --- |
| **Sr. No.** | **Diagram Name** | **Page No.** |
| 1 | Use case Diagram | 17 |
| 2 | Entity Relationship Diagram | 19 |
| 3 | Class Diagram | 20 |
| 4 | Sequence Diagram | 20 |
| 5 | Deployment Diagram | 21 |

# 1. Introduction

## 1.1 Background

The **"Student Voting System"** project was developed to create a more secure and efficient way for students to vote during college events. Traditional voting methods were prone to errors, delays, and fraud, making it difficult to ensure fairness. By using technology, this system streamlines the voting process, allowing only eligible students with verified college ID cards to participate. It simplifies identity verification and speeds up the overall voting experience.

One of the key features of the system is the **one-time voting rule**, which ensures each student can vote only once for an event by tracking their roll number. This prevents duplicate voting and maintains the integrity of the process. The system also includes a mobile confirmation message, which provides students with assurance that their vote has been correctly recorded for the right candidate.

Overall, the **"Student Voting System"** improves the voting experience by making it more transparent and reliable. The project addresses common challenges like ID verification and voting accuracy, ensuring that college events are conducted fairly and efficiently, benefiting both students and the voting panel.

## 1.2 Objectives

The main goal of the "Student Voting System" is to make voting for college events more secure and fair. The system ensures that only students with valid college ID cards can vote, which helps prevent anyone who isn’t eligible from voting. By using technology, the system makes the voting process quicker and easier for both the students and the voting panel.

One important feature is the one-time voting rule, which allows each student to vote only once for an event. The system keeps track of student roll numbers to make sure no one votes more than once. After voting, students receive a confirmation message on their phone showing the event name and the candidate they voted for, which gives them confidence that their vote was recorded correctly.

Finally, the project aims to make the voting process more accurate by reducing the mistakes that can happen with manual voting. With this digital system, votes are recorded reliably, and the process is simple and easy to use, encouraging more students to participate in voting.

## 1.3 Scope, Purpose, Applicability

.

**1.3**.1 **Scope**

The "**Student Voting System**" is designed to be used in college events where students need to vote, such as elections or competitions. It is mainly for the voting panel, admin, and students of the college. The system will handle ID verification, vote registration, and provide confirmation to students after they vote. It will also ensure that each student can vote only once for a particular event.

**1.3.2** **Purpose**

The purpose of this system is to make the voting process simple, fast, and secure. It eliminates manual voting, which can have errors, and ensures that only eligible students can vote. This system is designed to improve the accuracy of the voting process, prevent fraud, and make it easier for students to participate in college events.

**1.3.3** **Applicability**

This system can be applied to any college event that requires student voting. It is useful for managing large-scale voting where many students are involved. The system can be used by colleges or universities that want to ensure fair and transparent voting for elections, competitions, or any event that requires student participation.

# 2. Survey of Technologies

**2.1 Existing System**

In many colleges, voting is still done using manual methods, where students either write down their votes on paper or verbally communicate their choices to a voting panel, who then records them. While this system can function adequately, it has several significant drawbacks. The manual voting process is often slow, especially when dealing with a large number of students, which can lead to long wait times and frustrated participants. Furthermore, it is prone to human errors, such as missrecorded votes or lost ballots, which can compromise the accuracy of the election results. Additionally, ensuring that only eligible students vote and that each student votes only once is challenging in a manual system, making it easier for fraud to occur. There is also limited feedback to students regarding their votes, as they may not receive confirmation that their selections were recorded properly, which diminishes transparency and trust in the process. Managing large-scale voting events becomes increasingly difficult with manual methods, as organizers struggle to keep track of who has voted and who hasn’t.

**2.2 Limitations of the Existing System**

-Time-consuming : The manual process is labor-intensive and takes a lot of time, especially during large events where many students are voting.

Prone to Errors : Mistakes can easily happen in recording votes manually, leading to inaccurate results that can affect the integrity of the voting process.

Lack of Security : It is challenging to ensure that only eligible students vote and that each student casts only one vote, increasing the risk of fraud.

No Instant Feedback : Students do not receive confirmation that their vote was recorded, which reduces transparency and trust in the system.

Hard to Manage Large Events : As the number of participants increases, it becomes increasingly complex and difficult to manage the voting process efficiently.

**2.3 Proposed System**

The proposed "**Student Voting System**" is a digital platform that aims to modernize and automate the voting process for college events. By allowing students to vote electronically after verifying their identity with their college ID card, the system streamlines the entire process. The voting panel is responsible for handling the verification and casting the vote within the system. Each vote is securely recorded, and students receive a confirmation message on their mobile devices after voting, which reassures them that their selections have been accurately logged. Additionally, the system tracks student roll numbers to enforce the one-time voting rule, preventing any instances of duplicate voting. This innovative approach not only enhances security but also makes the voting experience more efficient and accessible for everyone involved.

**Advantages of the Proposed System**

Faster Voting Process: Automating the system allows for a quicker voting experience, significantly reducing wait times during events with many participants.

Fewer Errors: Electronic recording of votes minimizes human error, resulting in more accurate and reliable election outcomes.

Improved Security: The system verifies that only students with valid college IDs can vote, ensuring that each student votes only once and reinforcing the integrity of the process.

Instant Feedback: The confirmation messages provide students with immediate feedback on their votes, enhancing transparency and trust in the system.

Easy to Manage: The system is well-equipped to handle large-scale voting efficiently, allowing organizers to focus on other important aspects of the event without worrying about the voting process.

User-Friendly Interface: A digital platform can be designed to be intuitive and easy to use, encouraging greater student participation in the voting process.

Real-Time Results: The system can provide real-time results, allowing for quicker announcements of outcomes and increasing student engagement in the events.

# 3. Requirement Analysis

## 3.1 Problem Definition

The current way of voting in colleges, which uses paper ballots or students telling their votes to a panel, has many problems. It takes too long, can have mistakes, and is hard to manage when there are many students. It is difficult to make sure only eligible students vote and that they vote only once. There are also no instant confirmations, so students don’t know if their vote was recorded properly, leading to less trust in the system.

Security is another big issue. Without proper checks, some students might vote more than once, or others who aren’t eligible might vote. This makes the system unreliable, especially for large events.

A new digital system is needed to make the voting process faster, more secure, and easier to manage. It should check students' IDs to ensure they are eligible, prevent duplicate voting, and give students instant confirmation that their vote has been recorded correctly. The "Student Voting System" aims to solve all these problems by providing a better, automated solution.

## 3.2 Requirement Specification

**Functional Requirements:**

1. The system verifies student identities using college ID cards to ensure only eligible students vote.

2. Each student can vote only once per event, with real-time tracking of roll numbers.

3. Students receive instant confirmation of their vote via mobile messages.

4. Admins can create and manage events, candidates, and monitor votes in real time.

5. Votes are securely stored, preventing tampering, with real-time vote counting and error handling.

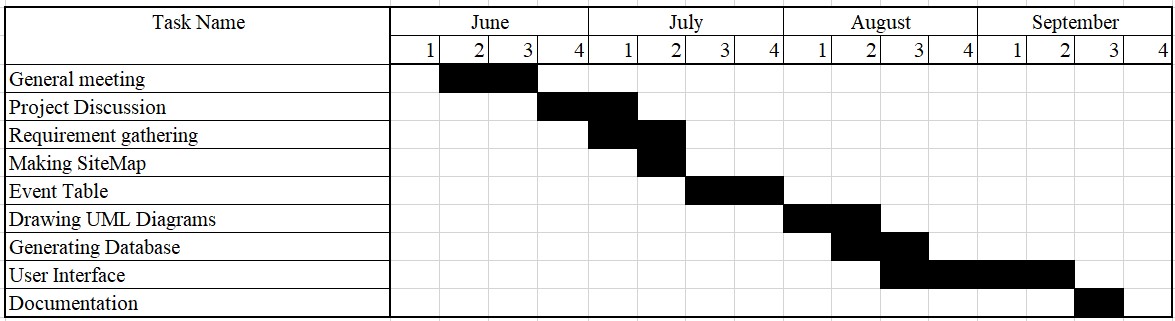
**Non-Functional Requirements:**

1. Security: The system must protect against unauthorized access, ensuring data encryption and secure storage of votes.

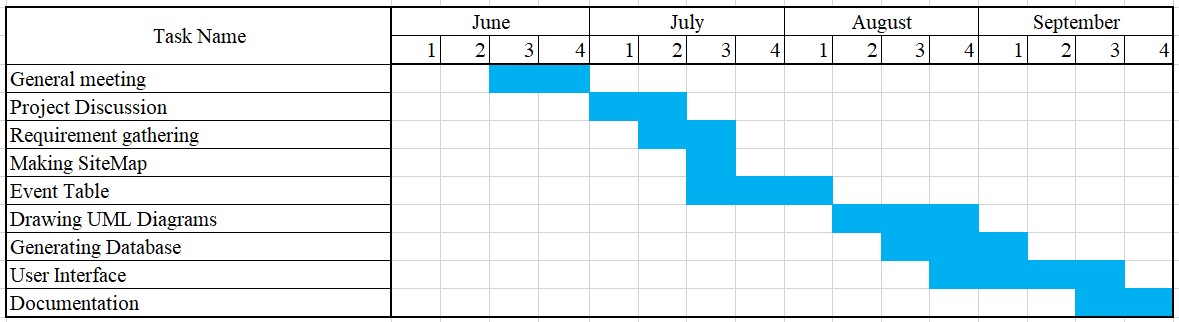
2. Reliability- The system should be highly reliable, with minimal downtime, ensuring smooth voting during events.

3. Usability- The interface should be easy to use for both students and administrators, providing a seamless voting experience.

**3.3 Planning and Scheduling – Gantt chart Planning:**



**Scheduling:**



**3.4 Software and Hardware Requirements:**

**Hardware Requirements**

Processor:- minimum Intel(R) Core(TM) i3-6100U

Speed:-2.30 GHZ

Hard Disk capacity:-1TB

Memory capacity:-minimum 2 GB and above

**Software Requirements**

Browser: Google Chrome,Firefox, etc.

Server : Xampp

Operating System : Windows 10/11

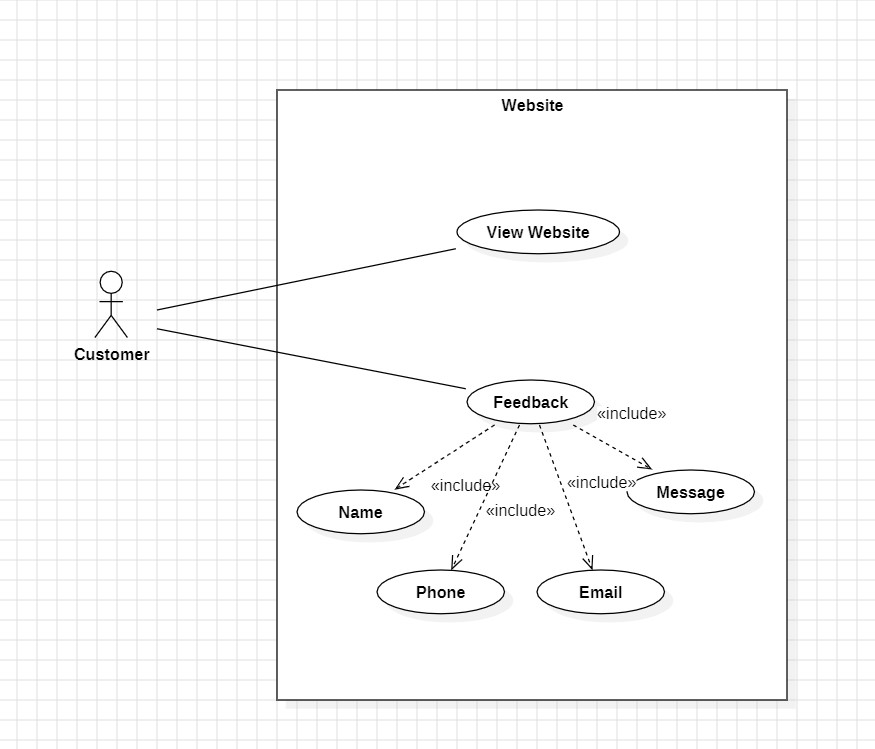
Back end : MySQL, Php

Front end : HTML, CSS, Javascript

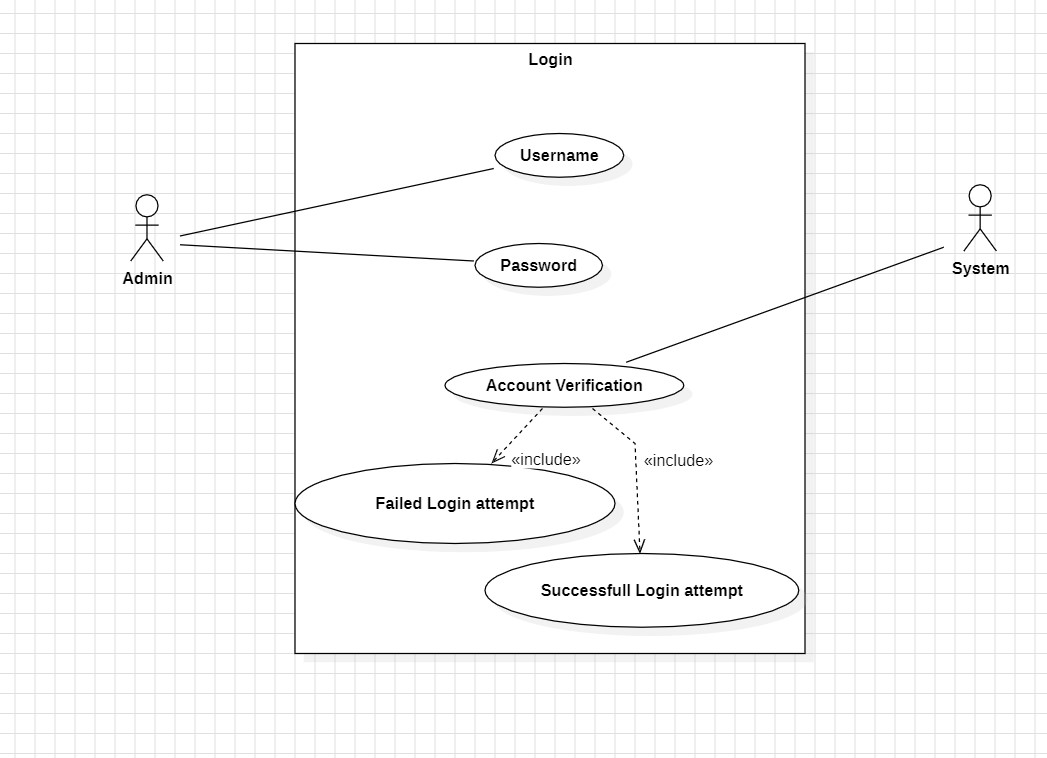
## 3.5 UML Diagrams

1. **Use-Case Diagram:**

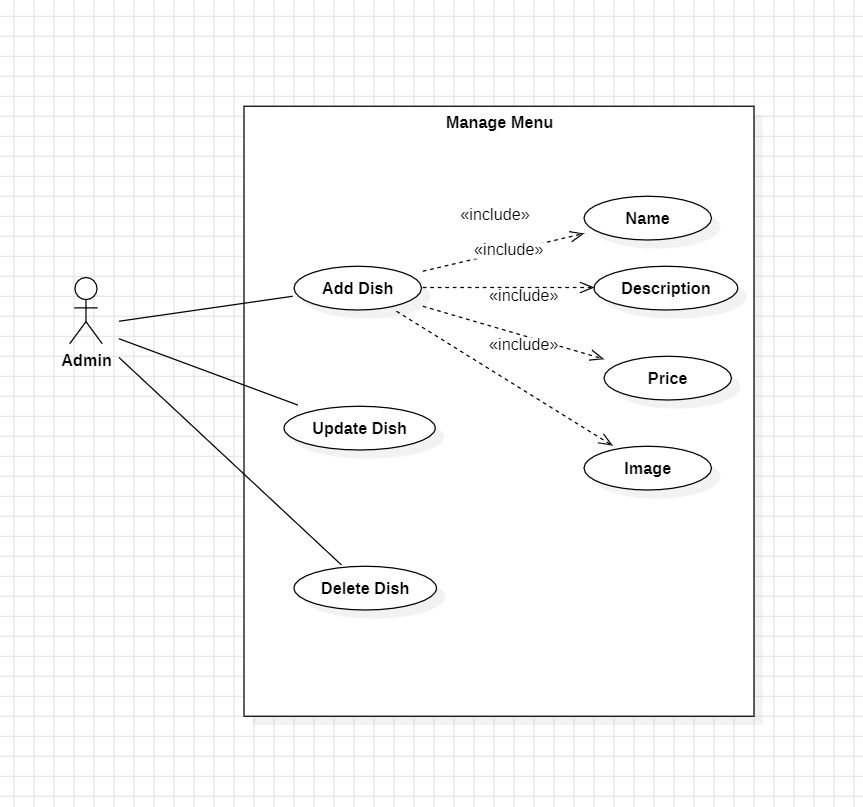
Website



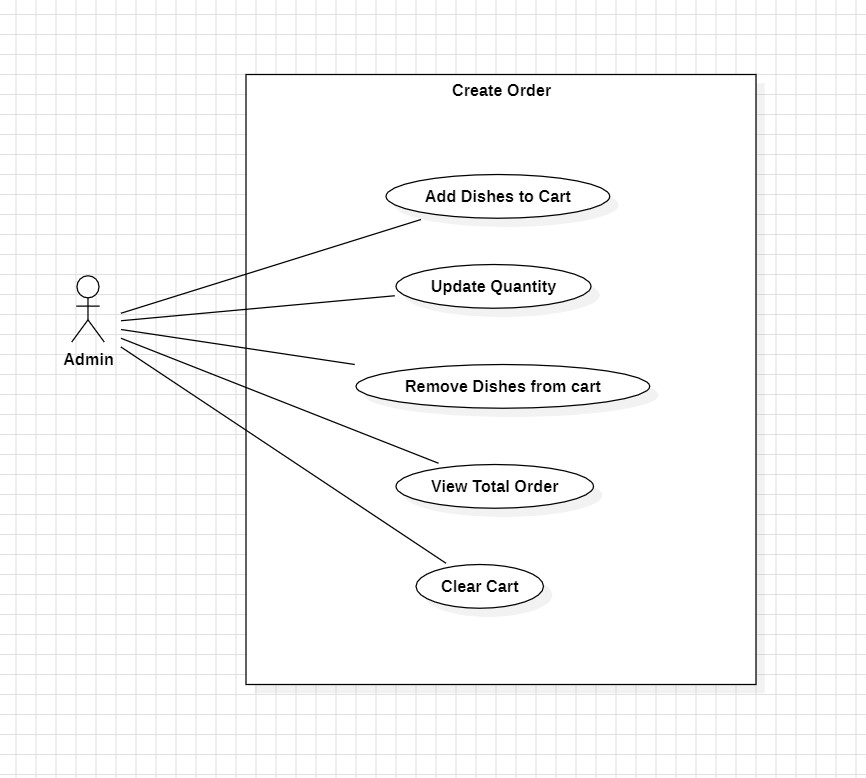
Admin Panel



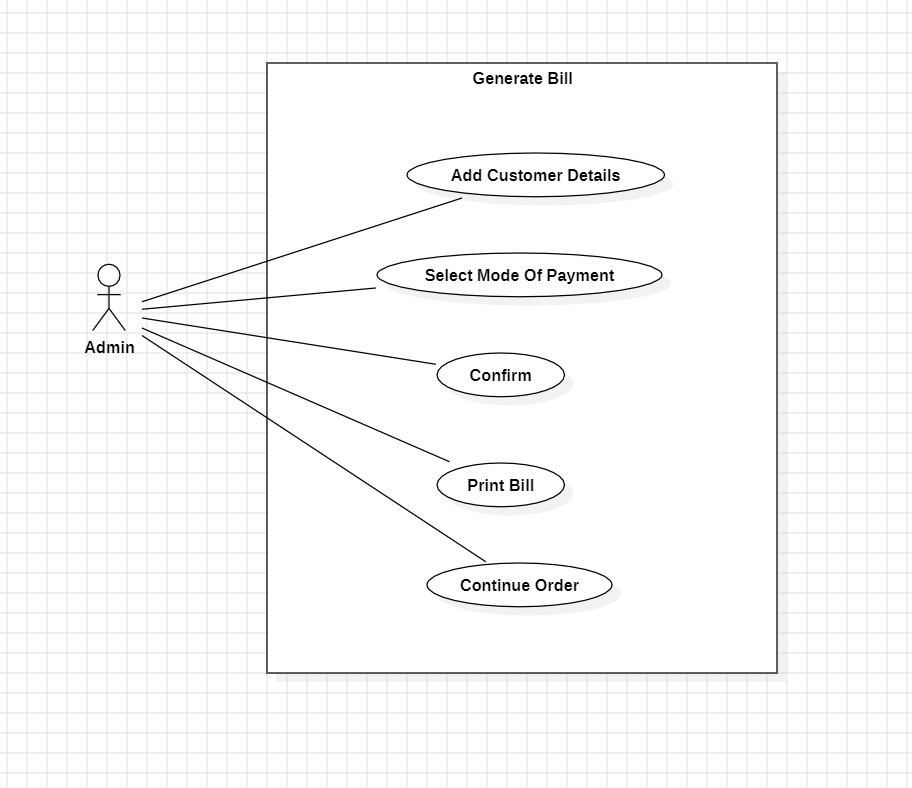
Manage Menu



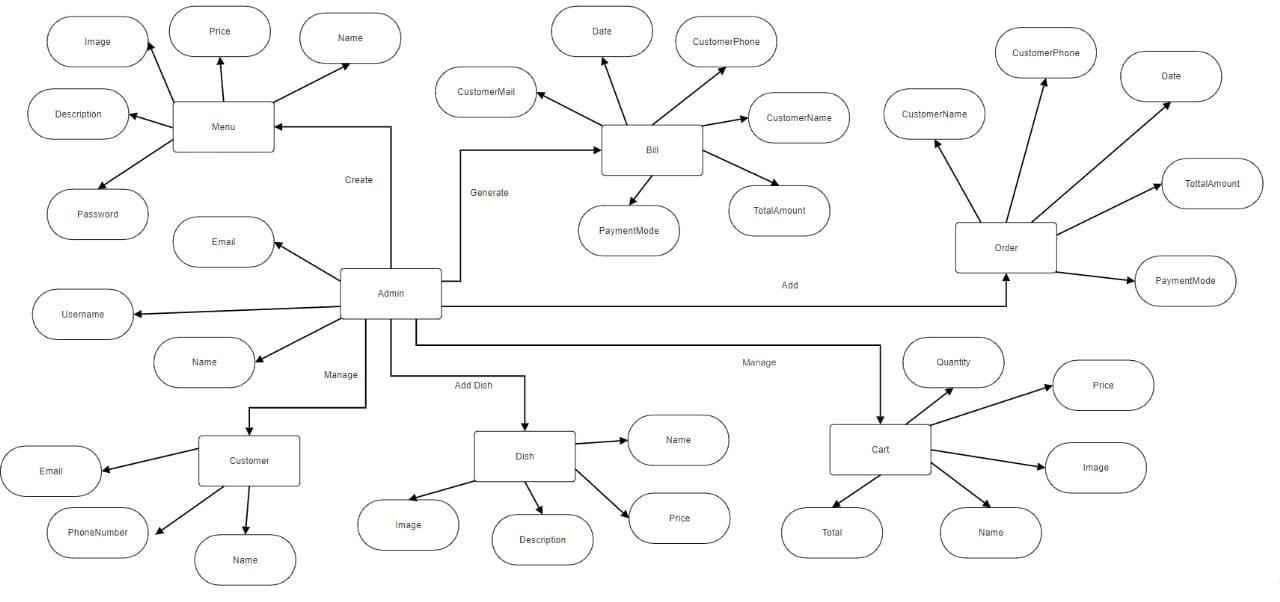
Create Order



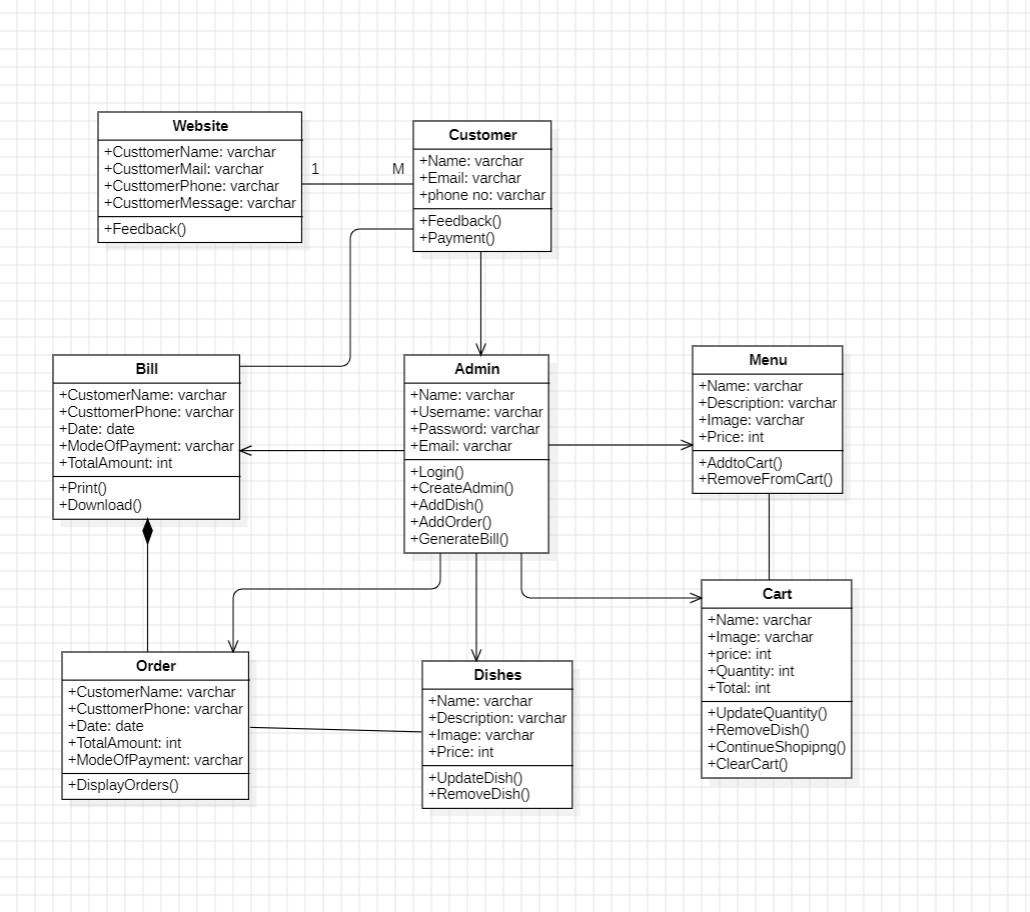
Generate bill



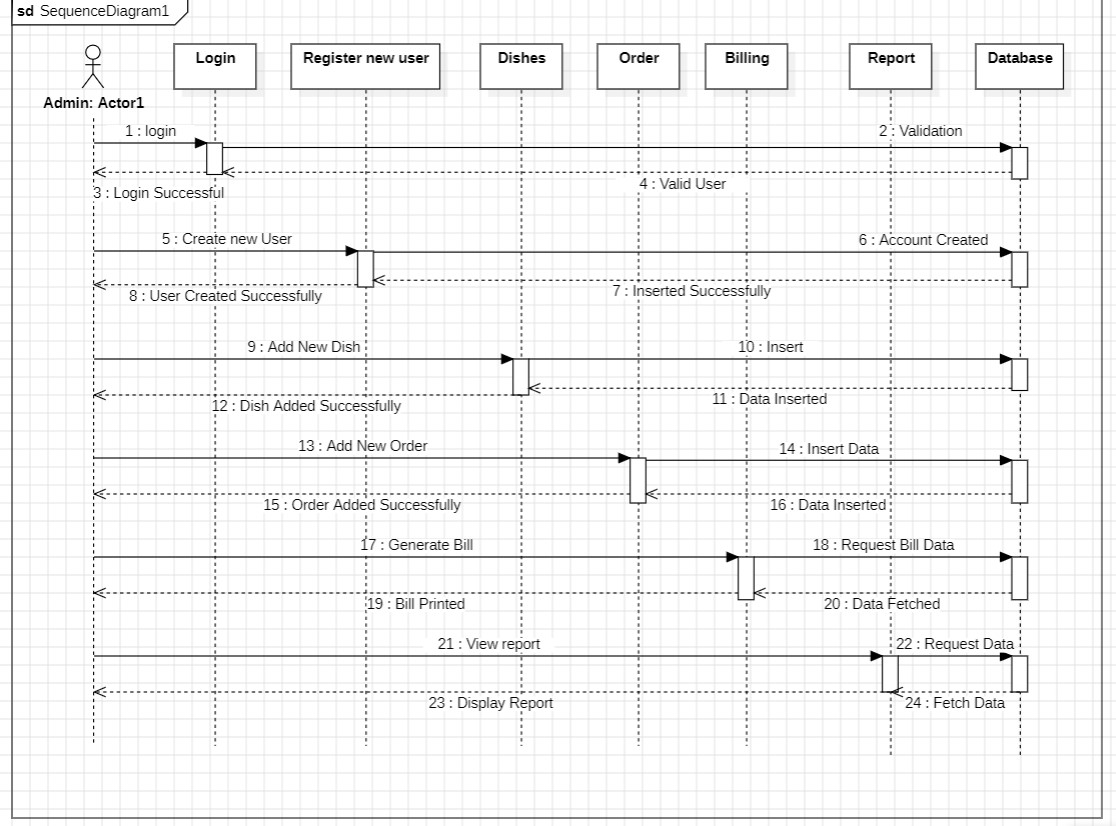
1. **ER Diagram:**



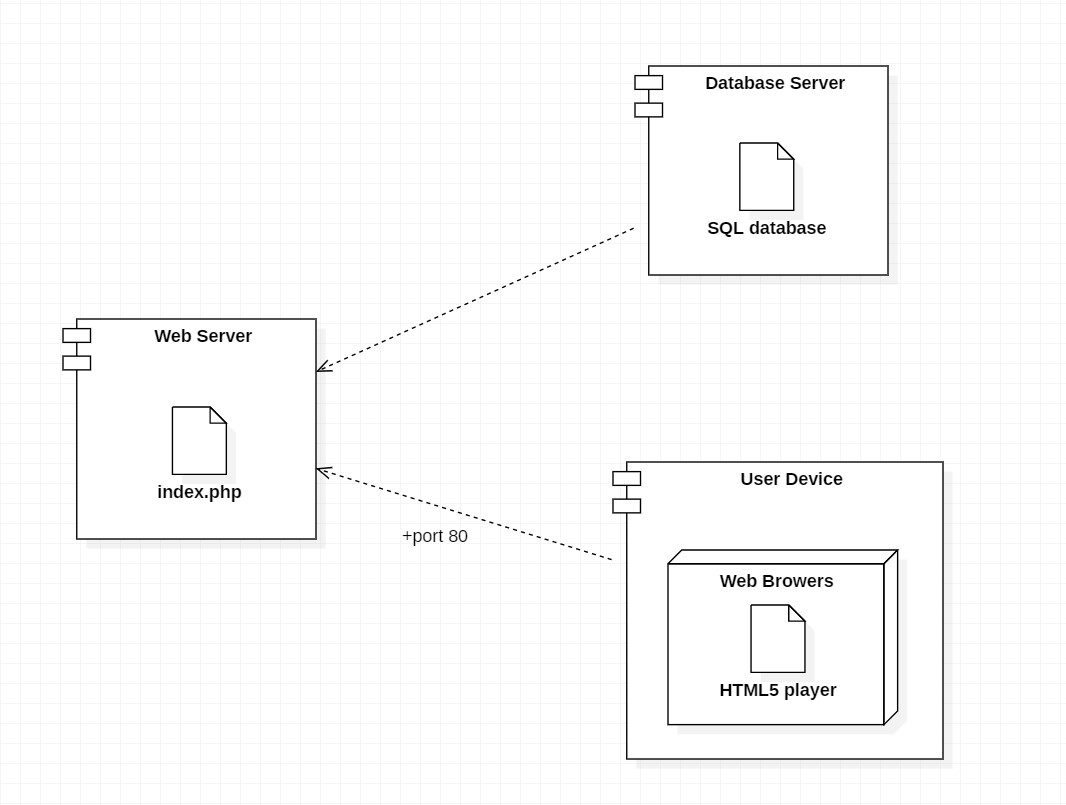
1. **Class Diagram:**



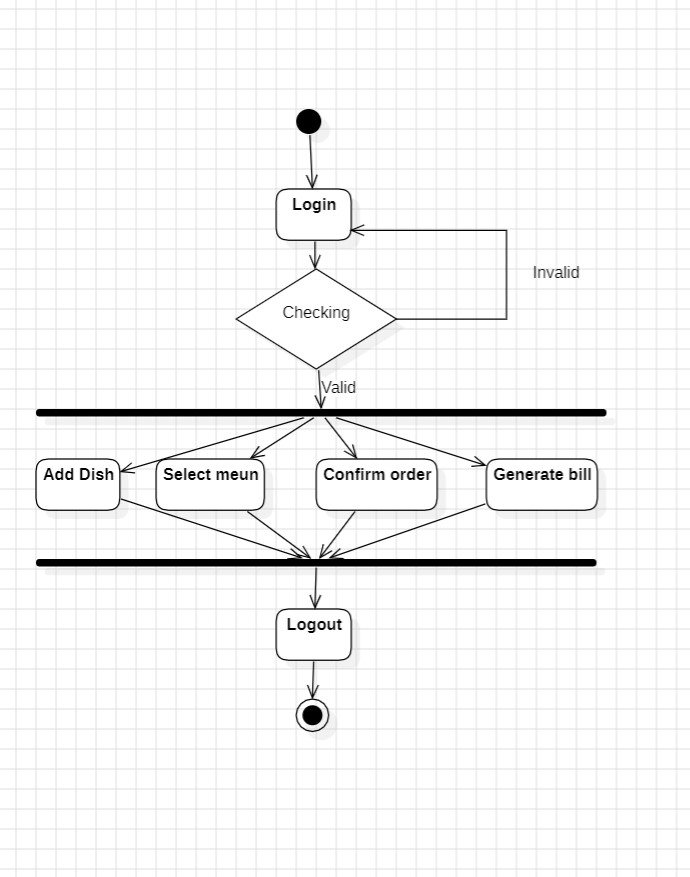
1. **Sequence Diagram:**



1. **Deployment Diagram:**



1. **Activity Diagram:**



# 4. SYSTEM DESIGN

## 4.1 Module Division

The system is divided into several modules based on functionality. They are as follows:

**User Interface Modules**

b. Voting Panel Interface: This module provides the voting panel with an interface to verify student details, input votes, and manage the overall voting process during the event.

c. Admin Panel Interface: This module provides admins with tools to create and manage events, candidates, view real-time voting results, and generate reports.

**Authentication and Authorization Module:**

This module handles the verification of student identity using college ID cards. It ensures that only eligible students are allowed to vote and that each student can vote only once per event.

**Vote Casting Module:**

This module allows the voting panel to cast votes on behalf of students after verifying their details. It securely records votes and ensures that each student’s vote is stored correctly and uniquely tied to their roll number.

**Confirmation and Notification Module:**

This module sends instant confirmation messages to students after they have voted, displaying the event name and the candidate they voted for. It ensures transparency and provides proof of the student’s vote.

**Voting Management Module:**

This module enables admins to create and manage events, add or remove candidates, set event times, and control access to voting. It also tracks voting activity and generates results at the end of the event.

**Security and Data Integrity Module:**

This module ensures that all voting data is securely stored and protected from unauthorized access. It includes encryption and integrity checks to prevent tampering with votes.

**Real-Time Vote Counting and Reporting Module:**

This module provides real-time vote counting and allows admins to view up-to-date results during and after the voting process. It generates detailed reports that can be used for analysis and decision-making.

**Database Management Module**

This module manages the storage, retrieval, and updating of all data related to students, voting events, candidates, and votes. It ensures that all data is correctly stored and accessible when needed, while also maintaining database integrity and performance.

## 4.2 Data Dictionary

This contains all data definitions for cross-referencing and for managing and controlling access to the information repository / database. It provides a very thorough interface description that is independent of the model itself. Changes made to a model may be applied to the data dictionary to determine if the changes have affected the model’s interface to other systems. Data dictionaries do not contain any actual data from the database, only book keeping information for managing it. Without a data dictionary, however, a database management system cannot access data from the database.

## 4.3 Schema Design

1. Table Name :- tbl\_admin

|  |  |  |  |
| --- | --- | --- | --- |
| **Column Name** | **Data Type** | **Index** | **Description** |
| id | int (100) | Not Null(primary key) | Admin id |
| full\_name | varchar (100) | Not Null | Admin full name |
| username | varchar (100) | Not Null | Admin username |
| email | varchar (100) | Not Null | Admin Email id |
| password | varchar (100) | Not Null | Admin password |

1. Table Name :- tbl\_cart

|  |  |  |  |
| --- | --- | --- | --- |
| **Column Name** | **Data Type** | **Index** | **Description** |
| id | int (100) | Not Null (primary key) | Cart id |
| name | varchar (100) | Not Null | Dish name |
| price | varchar (100) | Not Null | Dish price |
| image | varchar (100) | Not Null | Dish image name |
| quantity | int (100) | Not Null | Dish quantity |

1. Table Name :- tbl\_feedback

|  |  |  |  |
| --- | --- | --- | --- |
| **Column Name** | **Data Type** | **Index** | **Description** |
| id | int (10) | Not Null (primary key) | Customer id |
| cname | varchar (200) | Not Null | Customer name |
| cpmail | varchar (100) | Not Null | Customer Email id |
| cphone | varchar (100) | Not Null | Customer phone number |
| cmessage | varchar (100) | Not Null | Customer feedback |

1. Table Name :- tbl\_food

|  |  |  |  |
| --- | --- | --- | --- |
| **Column Name** | **Data Type** | **Index** | **Description** |
| id | int (10) | Not Null (primary key) | Dish id |
| title | varchar (200) | Not Null | Dish name |
| price | varchar (100) | Not Null | Dish price |
| image\_name | varchar (100) | Not Null | Dish image name |
| description | varchar (100) | Not Null | Dish description |
| ord\_time | int (100) | Not Null | Number of  times a dish is ordered |

1. Table Name :- tbl\_order

|  |  |  |  |
| --- | --- | --- | --- |
| **Column Name** | **Data Type** | **Index** | **Description** |
| id | int (10) | Not Null (primary key) | Order id |
| name | varchar (200) | Not Null | Customer name |
| number | varchar (100) | Not Null | Customer mobile number |
| email | varchar (100) | Not Null | Customer Email id |
| method | varchar (100) | Not Null | Payment method |
| total\_products | varchar (100) | Not Null | Total number of dishes ordered |
| total\_price | varchar (100) | Not Null | Total price of dishes |
| table\_no | int (100) | Not Null | Table Number of restaurant |
| date | date | Not Null | Order date |

**4.4 Event Table**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Sr.No | Event | Trigger | Source | Activity | Response | Destination |
| 1 | Login | User Want to login | System | User enters user name & password | 1 . Login successful  2 . Login failed | System |
| 2 | Vote Option | User want vote to event | System | Select vote option  1 .Vote for event  2 . GS Voting  3 . CR Voting | Option selected | System |
| 3 | Vote to Event | User want vote to candidate | System | User select candidate to vote | 1 . Vote inserted into vote table  2.SMS displayed on student mobile no | System |
| 4 | Admin login | Admin want to login | Admin | Admin enters username & password | 1 . Login successful  2 . Login failed | Admin |
| 5 | Manage Event | Admin want to  1 .Add event  2 .Delete event  3 .Edit event  4 . Add candidate  5 . Delete candidate  6 . Edit candidate | System | Admin fill event & candidate details | Data fetched successfilly | Admin |
| 6 | Vote Count | Event displyed | System | Admin select event to view result | Vote Count displyed | Admin |

## 

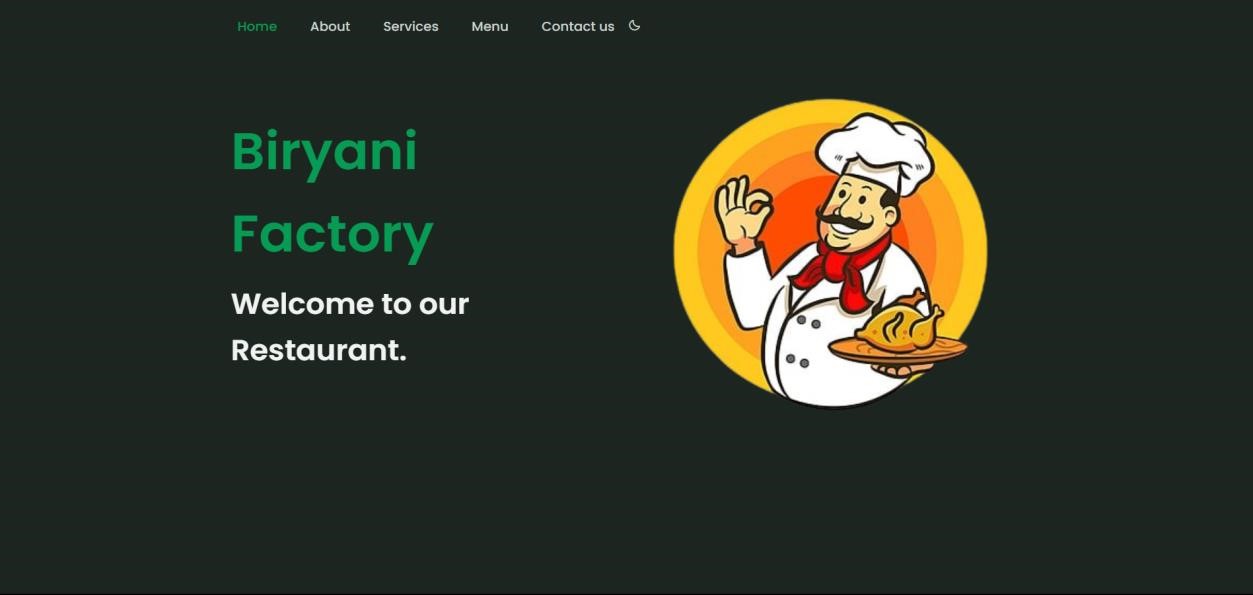
## 4.5 Test Cases

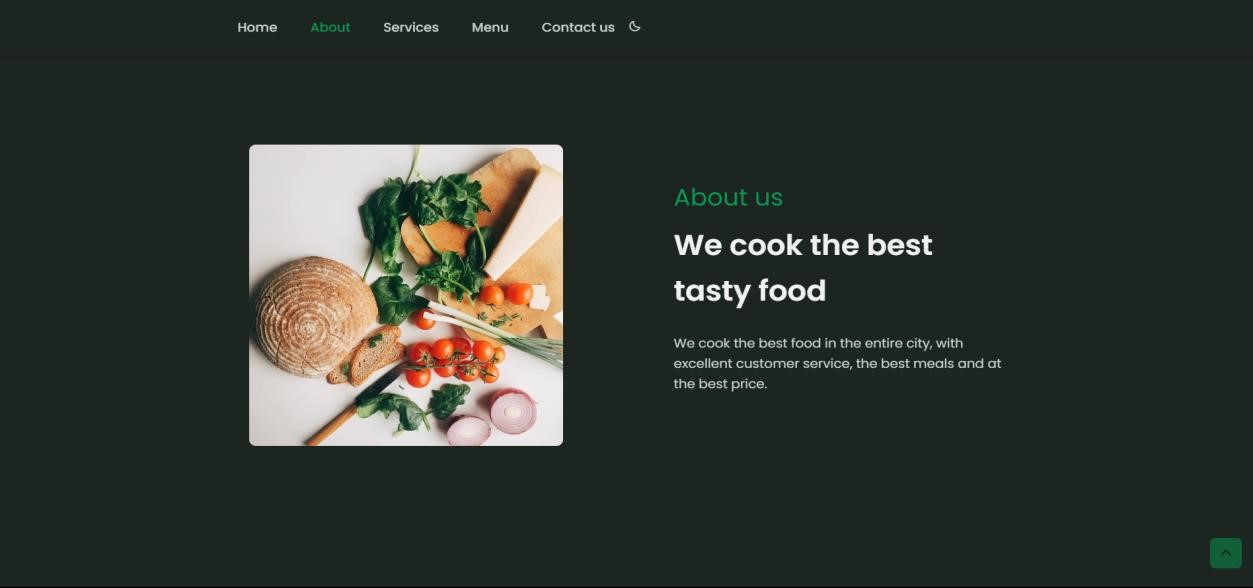
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Test Case Id** | **Condition** | **Test Data** | **Expected**  **Outcome** | **Actual**  **Outcome** | **Pass/Fail** |
| 1 | Login | Empty Username and Password  Field | Required message display | Required message is displayed | pass |
| Username left empty and password entered | Required message display | Required message is displayed | pass |
| Username entered and password  field left blank | Required message display | Required message is displayed | pass |
| Wrong username and correct password | Required message display | Error message is displayed | pass |
| Correct username and wrong password | Error message display | Error message is displayed | pass |
| Correct username and password | Redirected to admin panel | Redirected to admin panel | pass |

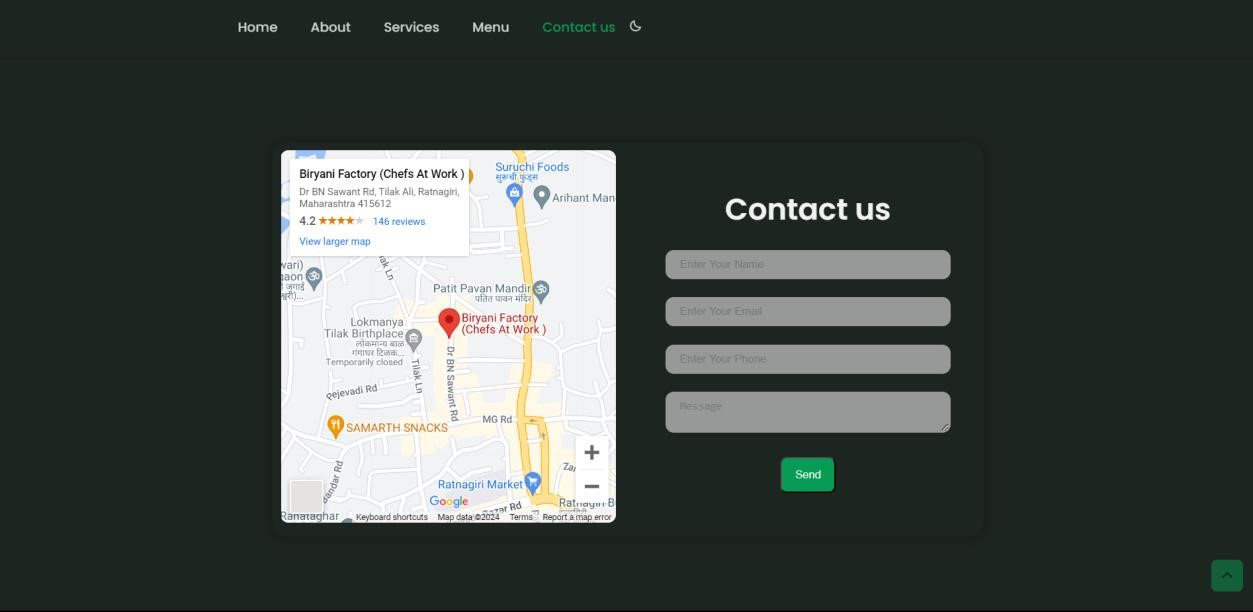
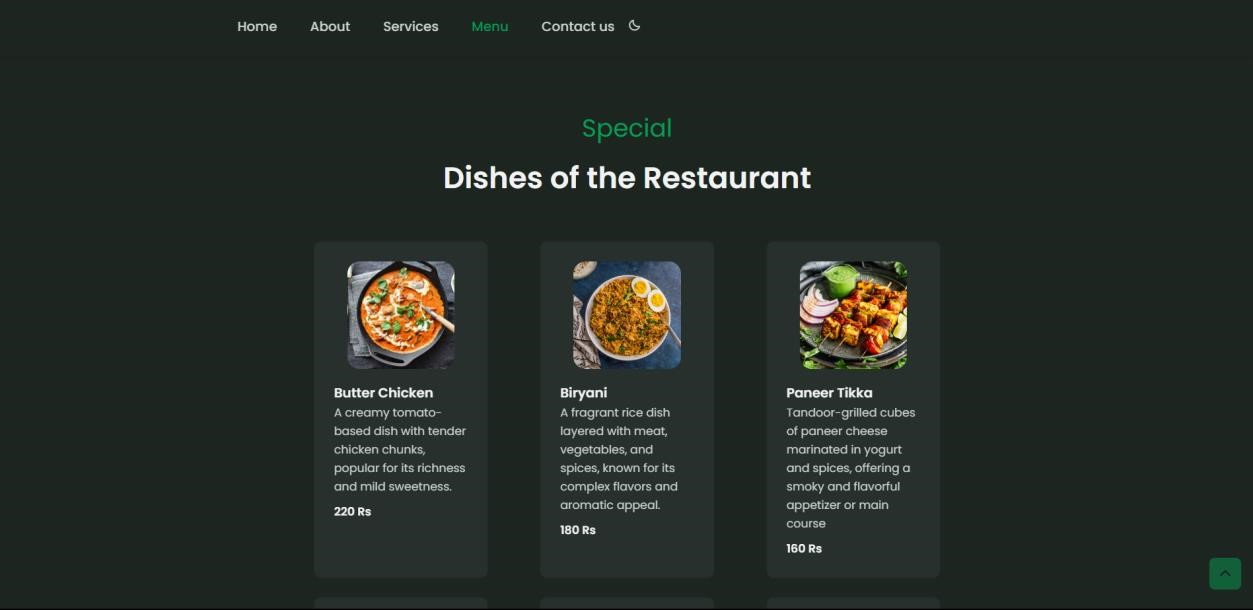
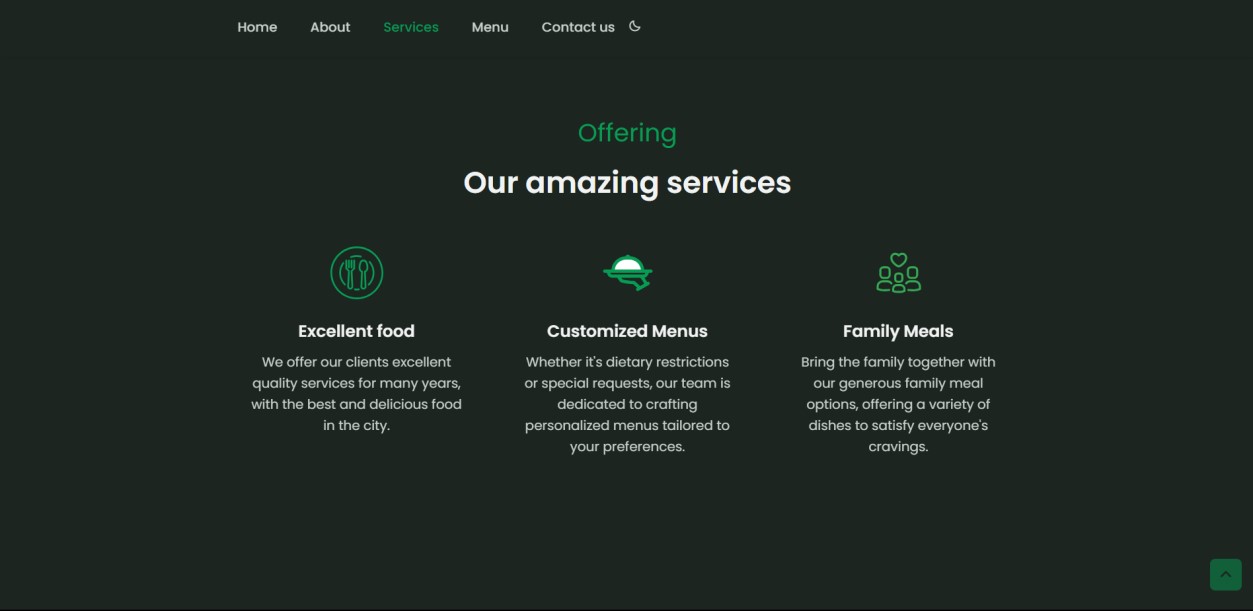
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 2 | Add Admin | If any field or fields are empty | Required message should display | Required message is displayed | pass |
| If incorrect data enterd in all the fields | Error message displaying match the required format | Error message is displayed | pass |
| If correct data enterd in all the fields | Admin added successfully | Admin added successfully | pass |
| 3 | Update  Admin | If any field or fields are empty | Required message should display | Required message is displayed | pass |
| If incorrect data enterd in any of the fields | Error message displaying match the required format | Error message is displayed | pass |
| If correct data enterd in all the fields | Admin  Updated  successfully | Admin  Updated  successfully | pass |
| 4 | Add Dish | If any field or fields are empty | Required message should display | Required message is displayed | pass |
| If incorrect data enterd in any of the fields | Error message displaying match the required format | Error message is displayed | pass |
| If correct data enterd in all the fields | Dish added successfully | Dish added successfully | Pass |
| If dish name enterd already exist | Dish already added | Dish already added | pass |

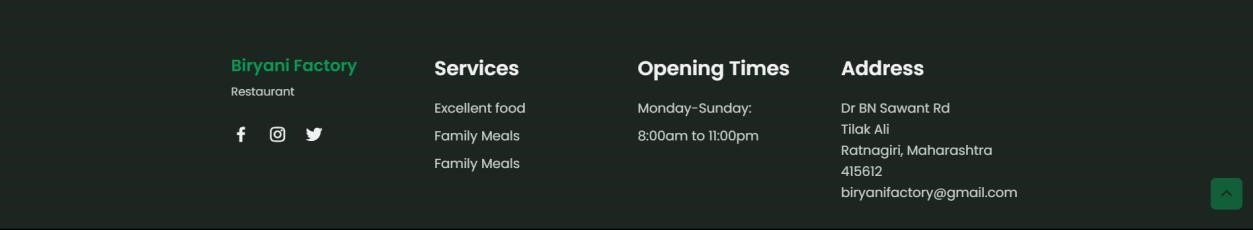
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 5 | Update Dish | If any field or fields are empty | Required message should display | Required message is displayed | pass |
| If incorrect data enterd in any of the fields | Error message displaying match the required format | Error message is displayed | pass |
| If correct data enterd in all the fields | Dish update successfully | Dish update successfully | pass |
| 6 | Add Order | No dishes selected | Cart is empty | Cart is empty | pass |
| Add dishes to cart | Dish added to cart | Dish added to cart | pass |
| If dish added to cart is selected again | Alert message "Dish already added to cart" | Alert message is displayed | pass |
| 7 | Manage Cart | Manage cart as per customer requirement | Cart update successfully | Cart update successfully | pass |
| 8 | Proceed to  Checkout | If any field or fields are empty | Required message should display | required message is displayed | pass |
| If incorrect data enterd in any of the fields | Error message displaying match the required format | Error message is displayed | pass |
| If correct data enterd in all the fields | Confirm order | Order confirmed | pass |
| 9 | Feedback | If any field or fields are empty | Required message should display | Required message is displayed | pass |
|  |  | If incorrect data enterd in any of the fields | Error message displaying match the required format | Error message is displayed | pass |
| If correct data enterd in all the fields | Feedback submited on admin panel | Feedback submited on admin panel | pass |

## 4.6 User Interface

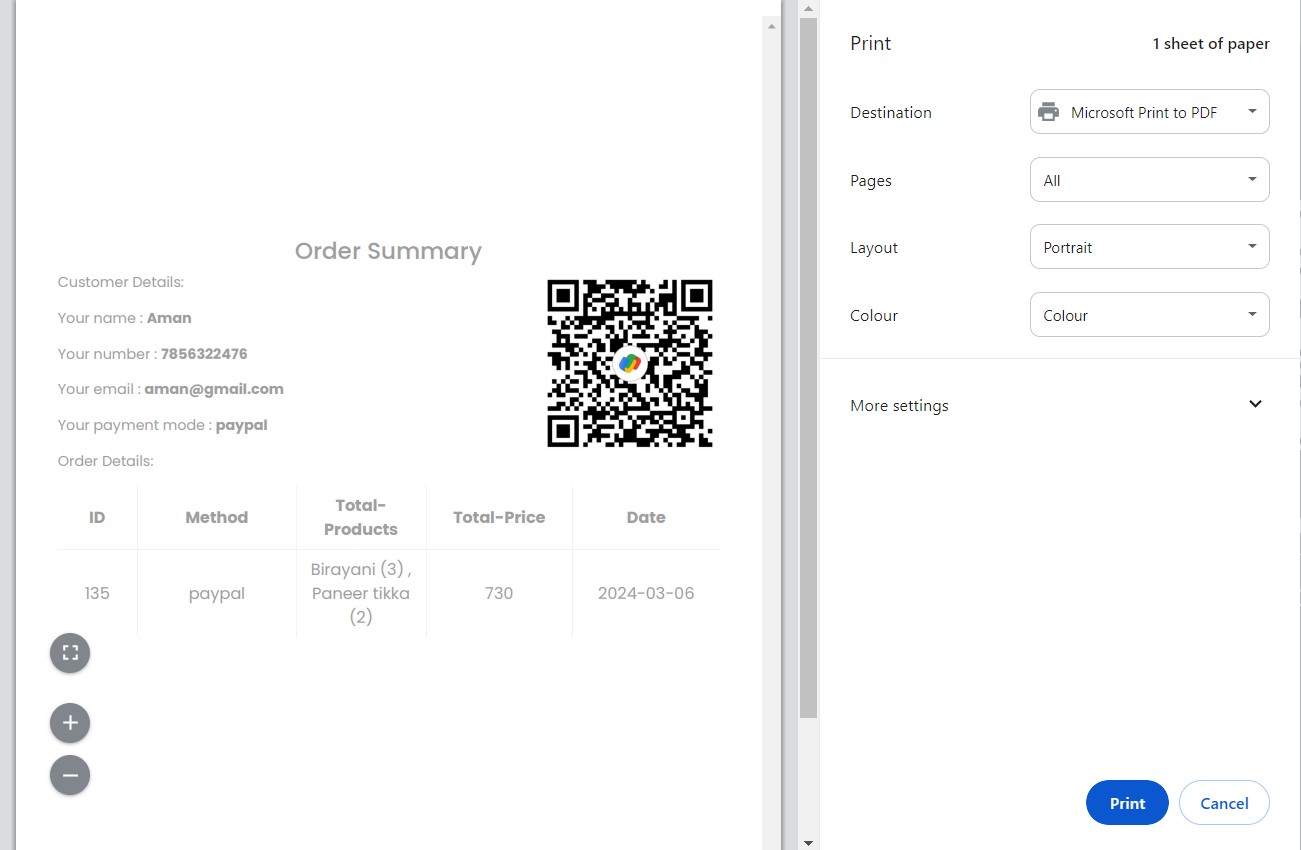
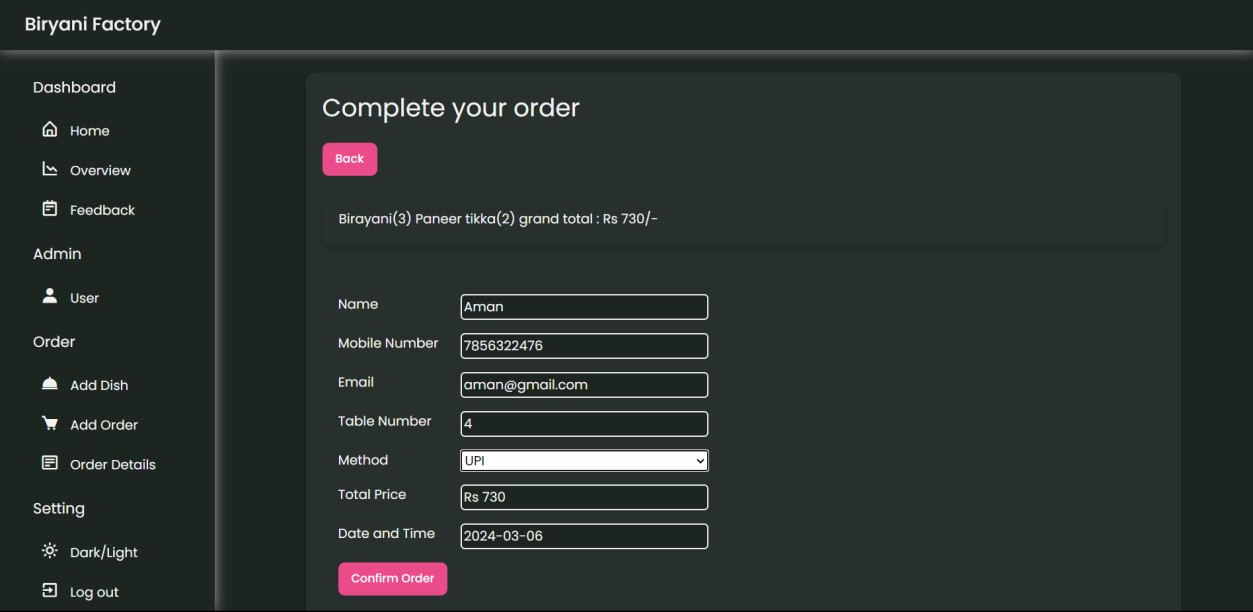
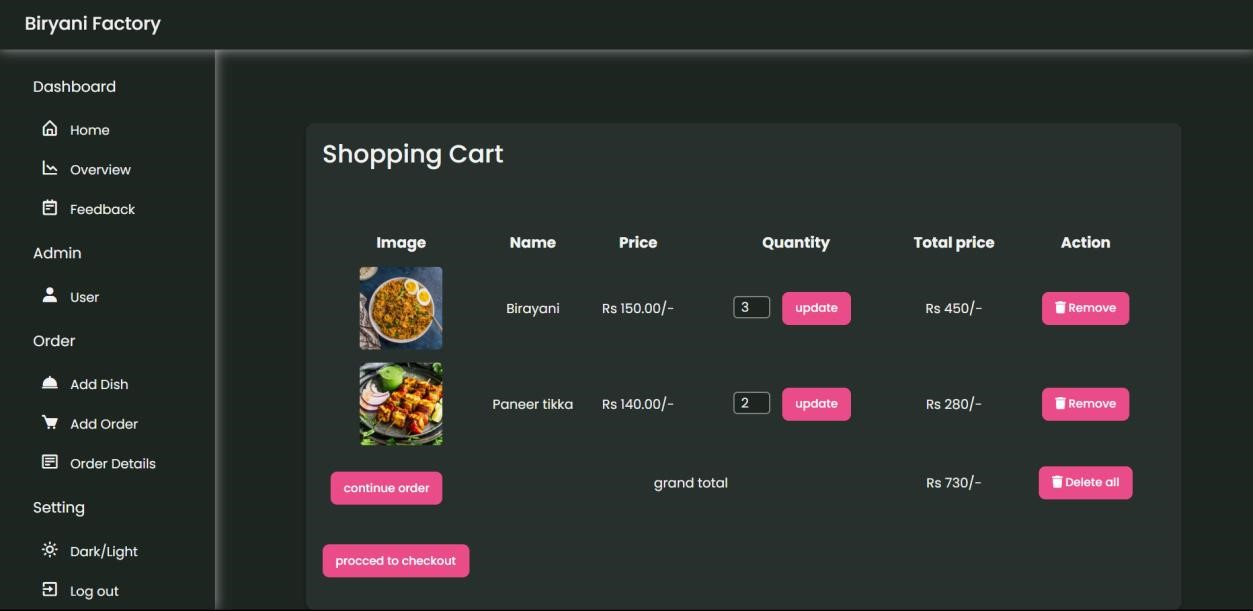
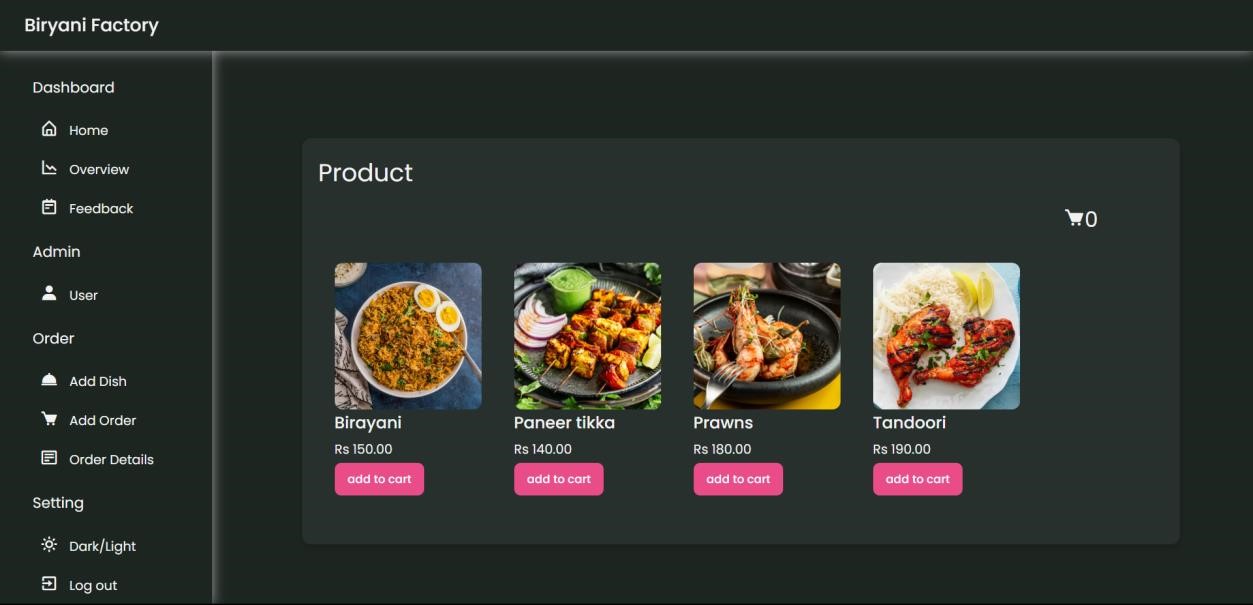
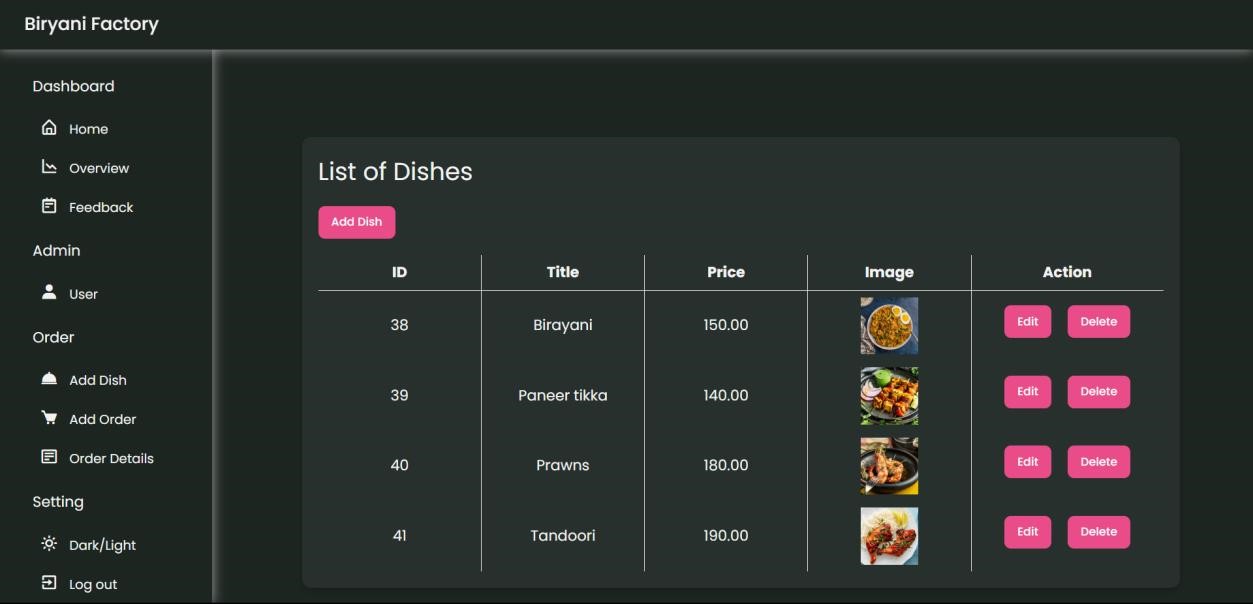
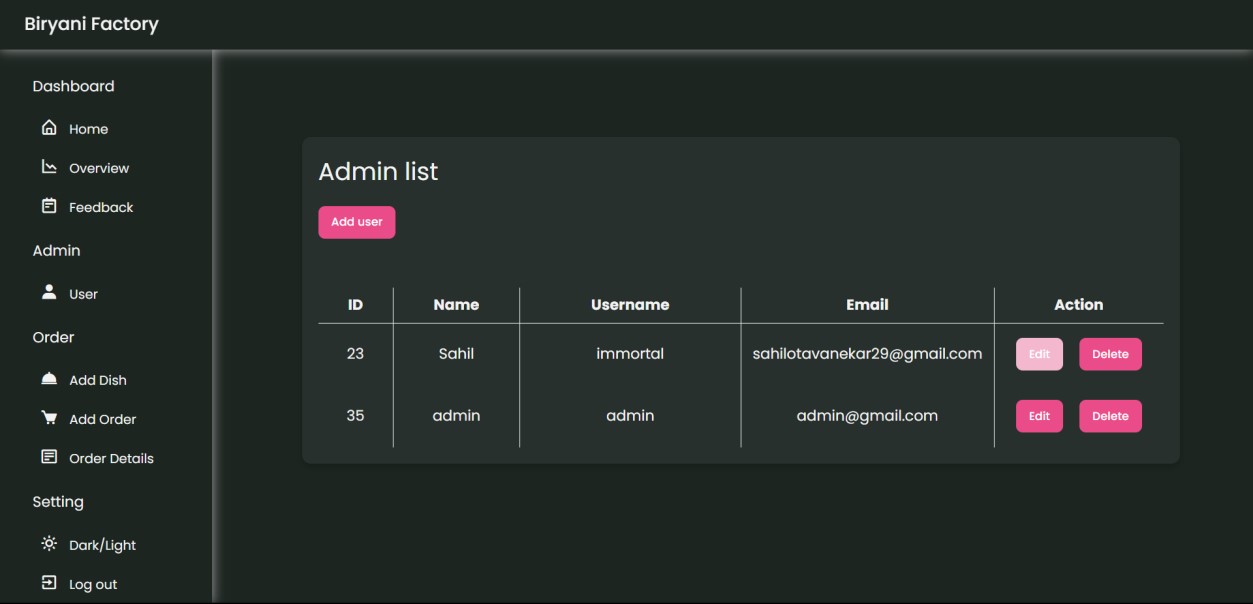
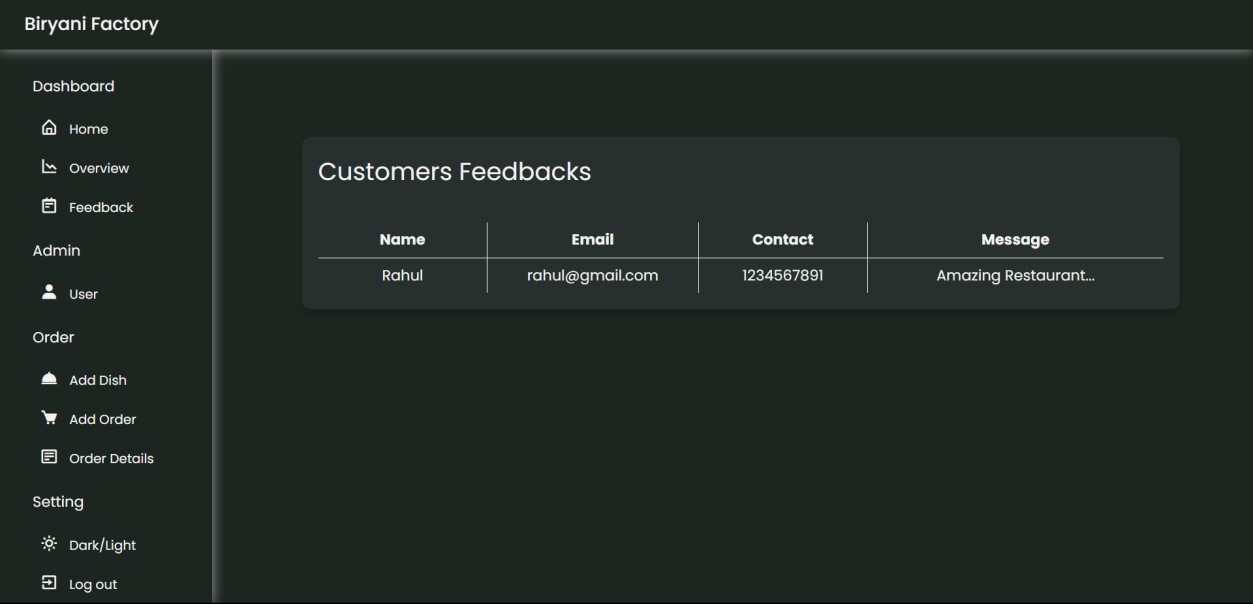
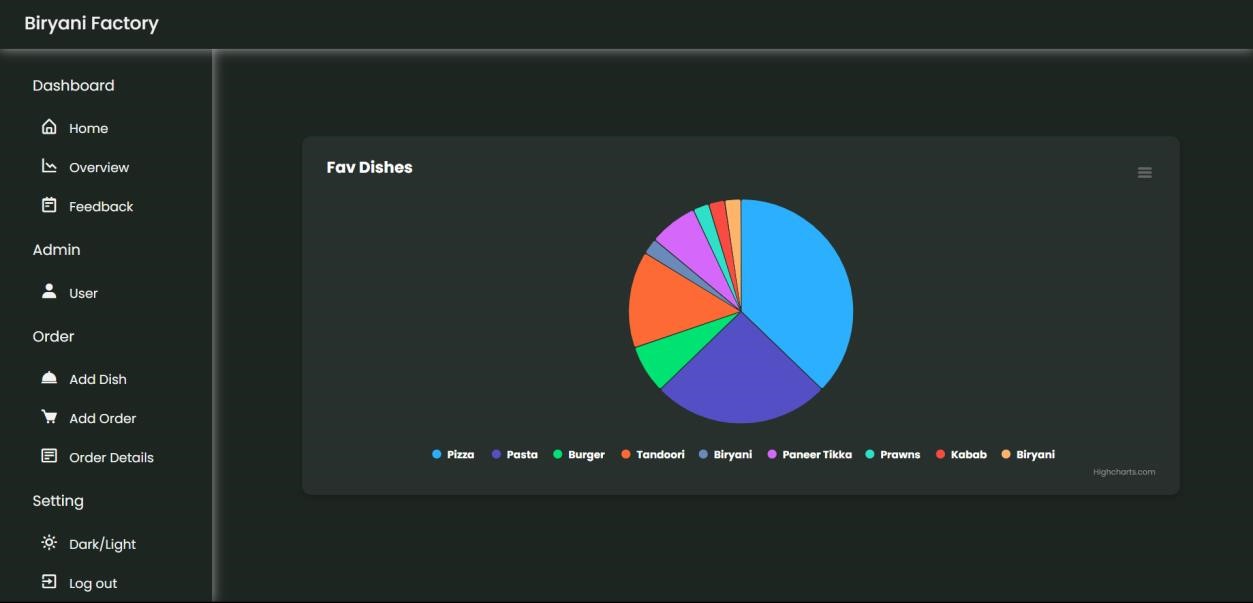
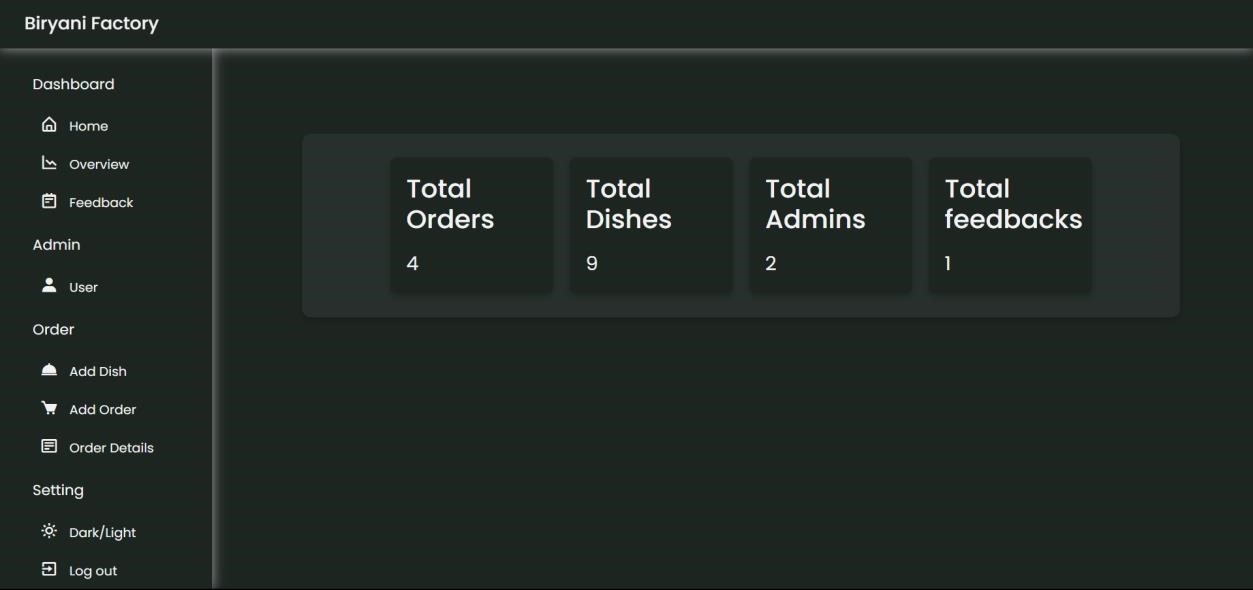
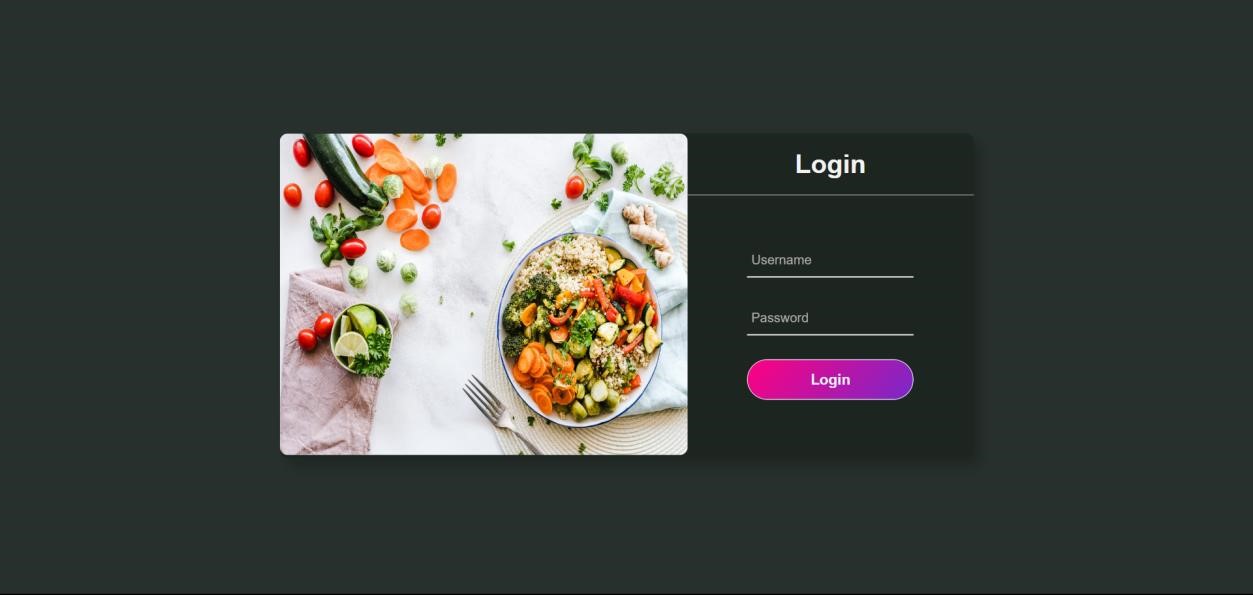


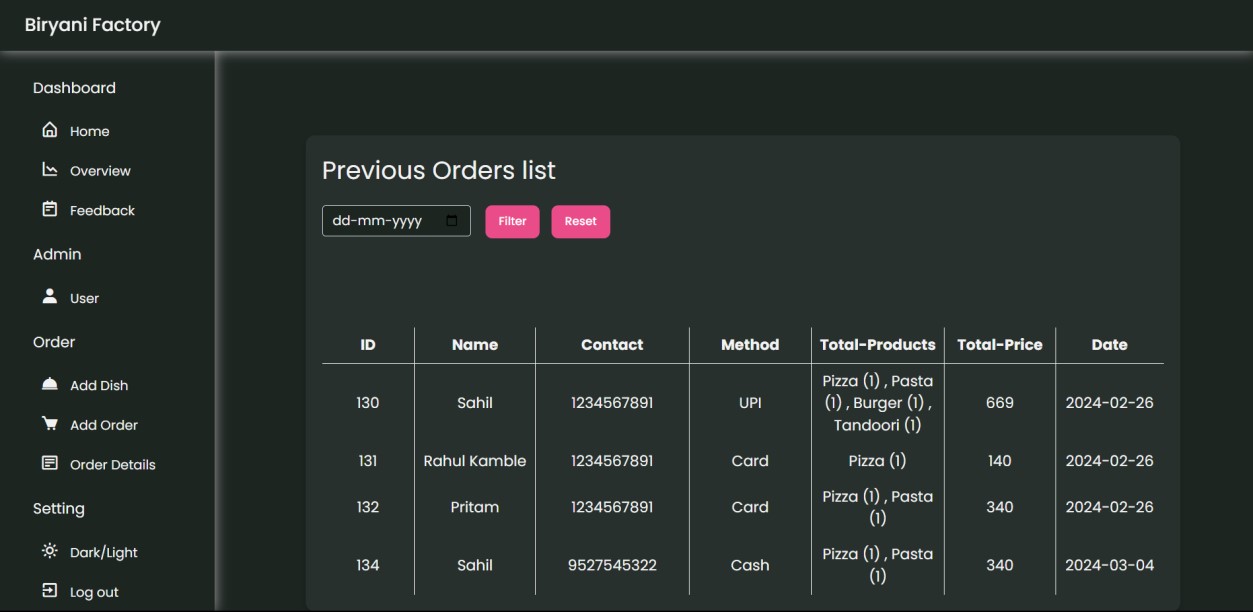






Admin panel





## 4.6 Code

**Website**

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<!--========== BOX ICONS ==========-->

<link href='https://cdn.jsdelivr.net/npm/boxicons@2.0.5/css/boxicons.min.css' rel='stylesheet'>

<!--========== CSS ==========-->

<link rel="stylesheet" href="assets/css/styles.css">

<title>Biryani Factory</title>

</head>

<body>

<!--========== SCROLL TOP ==========-->

<a href="#" class="scrolltop" id="scroll-top">

<i class='bx bx-chevron-up scrolltop\_\_icon'></i>

</a>

<!--========== HEADER ==========-->

<header class="l-header" id="header">

<nav class="nav bd-container">

<div class="nav\_\_menu" id="nav-menu">

<ul class="nav\_\_list">

<li class="nav\_item"><a href="#home" class="nav\_link activelink">Home</a></li>

<li class="nav\_item"><a href="#about" class="nav\_link">About</a></li>

<li class="nav\_item"><a href="#services" class="nav\_link">Services</a></li>

<li class="nav\_item"><a href="#menu" class="nav\_link">Menu</a></li>

<li class="nav\_item"><a href="#contact" class="nav\_link">Contact us</a></li>

<li><i class='bx bx-moon change-theme' id="theme-button"></i></li>

</ul>

</div>

<div class="nav\_\_toggle" id="nav-toggle">

<i class='bx bx-menu'></i>

</div>

</nav>

</header>

<main class="l-main">

<!--========== HOME ==========-->

<section class="home" id="home">

<div class="home\_\_container bd-container bd-grid">

<div class="home\_\_data">

<h1 class="home\_\_title">Biryani Factory</h1>

<h2 class="home\_\_subtitle">Welcome to our <br> Restaurant.</h2>

</div>

<img src="assets/img/chef-removebg-preview.png" alt="" class="home\_\_img">

</div>

</section>

<!--========== ABOUT ==========-->

<section class="about section bd-container" id="about">

<div class="about\_\_container bd-grid">

<div class="about\_\_data">

<span class="section-subtitle about\_\_initial">About us</span>

<h2 class="section-title about\_\_initial">We cook the best <br> tasty food</h2> <p class="about\_\_description">We cook the best food in the entire city, with excellent customer

service, the best meals and at the best price.</p> </div>

<img src="assets/img/about.jpg" alt="" class="about\_\_img"> </div>

</section>

<!--========== SERVICES ==========-->

<section class="services section bd-container" id="services">

<span class="section-subtitle">Offering</span>

<h2 class="section-title">Our amazing services</h2>

<div class="services\_\_container bd-grid">

<div class="services\_\_content">

<svg class="services\_\_img" xmlns="http://www.w3.org/2000/svg">

</svg>

<h3 class="services\_\_title">Excellent food</h3>

<p class="services\_\_description">We offer our clients excellent quality services for many years,

with the best and delicious food in the city.</p> </div>

<div class="services\_\_content">

<svg class="services\_\_img" width="800px" height="800px" viewBox="0 0 32.00 32.00"

</svg>

<h3 class="services\_\_title">Customized Menus</h3>

<p class="services\_\_description"> Whether it's dietary restrictions or special requests, our team is

dedicated to crafting personalized menus tailored to your preferences.</p> </div>

<div class="services\_\_content">

<svg class="services\_\_img" xmlns="http://www.w3.org/2000/svg" viewBox="0

0 25 25" id="Family">

</svg>

<h3 class="services\_\_title">Family Meals</h3>

<p class="services\_\_description">Bring the family together with our generous family meal options,

offering a variety of dishes to satisfy everyone's cravings.</p> </div>

</div>

</section>

<!--========== MENU ==========-->

<section class="menu section bd-container" id="menu">

<span class="section-subtitle">Special</span>

<h2 class="section-title">Dishes of the Restaurant</h2>

<div class="menu\_\_container bd-grid">

<div class="menu\_\_content">

<img src="assets/img/butter-chicken.jpg" alt="" class="menu\_\_img">

<h3 class="menu\_\_name">Butter Chicken</h3>

<span class="menu\_\_detail">A creamy tomato-based dish with tender chicken chunks, popular for its richness and mild sweetness. </span> <span class="menu\_\_preci">220 Rs</span>

</div>

<div class="menu\_\_content">

<img src="assets/img/Chicken-Biryani.jpeg" alt="" class="menu\_\_img">

<h3 class="menu\_\_name">Biryani</h3>

<span class="menu\_\_detail"> A fragrant rice dish layered with meat, vegetables, and spices, known

for its complex flavors and aromatic appeal. </span>

<span class="menu\_\_preci">180 Rs</span>

</div>

<div class="menu\_\_content">

<img src="assets/img/Paneer-Tikka.jpg" alt="" class="menu\_\_img">

<h3 class="menu\_\_name">Paneer Tikka</h3>

<span class="menu\_\_detail">Tandoor-grilled cubes of paneer cheese marinated in yogurt and spices,

offering a smoky and flavorful appetizer or main course</span>

<span class="menu\_\_preci">160 Rs</span>

</div>

<div class="menu\_\_content">

<img src="assets/img/Chicken-Tandoori.jpg" alt="" class="menu\_\_img">

<h3 class="menu\_\_name">Chicken Tandoori</h3>

<span class="menu\_\_detail">Chicken tandoori is marinated in yogurt and spices, and then cooked in a tandoor, a cylindrical clay oven.

</span>

<span class="menu\_\_preci">170 Rs</span>

</div>

<div class="menu\_\_content">

<img src="assets/img/Indo-Chinese-Fried-Rice.jpg" alt="" class="menu\_\_img">

<h3 class="menu\_\_name">Fried Rice</h3>

<span class="menu\_\_detail">

Fried rice is a dish made with cooked rice that has been stir-fried in a wok or pan with various ingredients like vegetables, eggs, meat, or seafood.

</span>

<span class="menu\_\_preci">110 Rs</span>

</div>

<div class="menu\_\_content">

<img src="assets/img/Paneer-butter-masala.jpg" alt="" class="menu\_\_img">

<h3 class="menu\_\_name">Paneer Butter Masala </h3>

<span class="menu\_\_detail">This is a rich and creamy curry featuring soft paneer cubes in a buttery tomato-based gravy</span>

<span class="menu\_\_preci">160 Rs</span>

</div>

</div>

</section>

<!--========== CONTACT US ==========-->

<section class="contact section bd-container" id="contact">

<section class="abt-container">

<div class="contact-box">

<div class="left">

<!--Google Map-->

</div>

<div class="right">

<h2 class="cnt section-title">Contact us</h2>

<form action="../code.php" method="POST">

<input type="text" class="field" name="fname" placeholder="Enter Your

Name" required pattern="[a-zA-Z ]{1,30}" title="Only characters allowed">

<input type="email" class="field" name="fmail" placeholder="Enter Your Email" required >

<input type="tel" pattern="^[789]\d{9}$" class="field" name="fphone"

placeholder="Enter Your Phone" required title="Invalid number">

<textarea placeholder="Message" name="fmessage" class="field" required pattern="[a-zA-Z ]{1,60}" title="Only characters allowed"></textarea>

<button name="fsubmit" class="button">Send</button>

</form>

</div>

</div>

</section>

</section>

</main>

<!--========== FOOTER ==========-->

<footer class="footer section bd-container">

<div class="footer\_\_container bd-grid">

<div class="footer\_\_content">

<a href="#" class="footer\_\_logo">Biryani Factory</a>

<span class="footer\_\_description">Restaurant</span>

<div>

<a href="#" class="footer\_\_social"><i class='bx bxl-facebook'></i></a>

<a href="#" class="footer\_\_social"><i class='bx bxl-instagram'></i></a> <a href="#" class="footer\_\_social"><i class='bx bxl-twitter'></i></a>

</div>

</div>

<div class="footer\_\_content">

<h3 class="footer\_\_title">Services</h3>

<ul>

<li><a href="#" class="footer\_\_link">Excellent food</a></li>

<li><a href="#" class="footer\_\_link">Family Meals</a></li>

<li><a href="#" class="footer\_\_link">Family Meals</a></li>

</ul>

</div>

<div class="footer\_\_content">

<h3 class="footer\_\_title">Opening Times</h3>

<ul>

<li><a href="#" class="footer\_\_link">Monday-Sunday: </a></li>

<li><a href="#" class="footer\_\_link">8:00am to 11:00pm </a></li>

</ul>

</div>

<div class="footer\_\_content">

<h3 class="footer\_\_title">Address</h3>

<ul>

<li>Dr BN Sawant Rd</li>

<li>Tilak Ali</li>

<li>Ratnagiri, Maharashtra 415612</li>

<li>biryanifactory@gmail.com</li>

</ul>

</div>

</div>

</footer>

<!--========== SCROLL REVEAL ==========-->

<script src="https://unpkg.com/scrollreveal"></script>

<!--========== MAIN JS ==========-->

<script src="assets/js/main.js"></script>

</body>

</html>

**Admin Panel**

**index.php**

<?php

require'config/function.php';

?>

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="utf-8" />

<meta name="viewport" content="width=device-width, initial-scale=1" />

<title>Sidebar</title>

<!-- Boxicons CSS -->

<link href="https://unpkg.com/boxicons@2.1.4/css/boxicons.min.css" rel="stylesheet" /> <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/css/bootstrap.min.css" rel="stylesheet" >

<!-- Style CSS -->

<link rel="stylesheet" href="assets\css\style.css" />

<link rel="stylesheet" href="assets\css\highcharts.css" />

<script src="https://code.highcharts.com/highcharts.js"></script>

<script src="https://code.highcharts.com/modules/exporting.js"></script>

<script src="https://code.highcharts.com/modules/accessibility.js"></script>

</head>

<body>

<?php include('sidebar.php'); ?>

</body>

</html>

**sidebar.php**

<!-- navbar -->

<nav class="navbar" id="nav">

<div class="logo\_item">

Biryani Factory

</div>

</nav>

<!-- sidebar -->

<nav class="sidebar">

<div class="menu\_content">

<ul class="menu\_items">

<div class="menu\_title menu\_dahsboard"></div>

<li class="item">

<a href="home.php" class="nav\_link">

<span class="navlink\_icon">

<i class="bx bx-home-alt"></i>

</span>

<span class="navlink">Home</span>

</a>

</li>

<li class="item">

<a href="overview2.php" class="nav\_link">

<span class="navlink\_icon">

<i class="bx bx-line-chart-down"></i>

</span>

<span class="navlink">Overview</span>

</a>

</li>

<li class="item">

<a href="cfeedback.php" class="nav\_link">

<span class="navlink\_icon">

<i class="bx bx-notepad"></i>

</span>

<span class="navlink">Feedback</span>

</a>

</li>

</ul>

<ul class="menu\_items">

<div class="menu\_title menu\_admin"></div>

<li class="item">

<a href="users1.php" class="nav\_link">

<span class="navlink\_icon">

<i class="bx bxs-user"></i>

</span>

<span class="navlink">User</span>

</a>

</li>

</ul>

<ul class="menu\_items">

<div class="menu\_title menu\_order"></div>

<li class="item">

<a href="dish-display.php" class="nav\_link">

<span class="navlink\_icon">

<i class="bx bxs-dish"></i>

</span>

<span class="navlink">Add Dish</span>

</a>

</li>

<li class="item">

<a href="add-order.php" class="nav\_link">

<span class="navlink\_icon">

<i class="bx bxs-cart"></i>

</span>

<span class="navlink">Add Order</span>

</a>

</li>

<li class="item">

<a href="order-history.php" class="nav\_link">

<span class="navlink\_icon">

<i class='bx bx-detail'></i>

</span>

<span class="navlink">Order Details</span>

</a>

</li>

</ul>

<ul class="menu\_items">

<div class="menu\_title menu\_setting"></div>

<li class="item" id="darkLight" onclick="removeAnimation()">

<a href="#" class="nav\_link">

<span class="navlink\_icon">

<i class="bx bx-sun" ></i>

</span>

<span class="navlink">Dark/Light</span>

</a>

</li>

<li class="item">

<a href="logout.php" class="nav\_link">

<span class="navlink\_icon">

<i class="bx bx-exit"></i>

</span>

<span class="navlink">Log out</span>

</a>

</li>

</ul>

</div>

</nav>

**dbcon.php**

<?php

session\_start();

define('SITEURL','http://localhost/food-order/');

$servername="localhost";

$username="root";

$password= "";

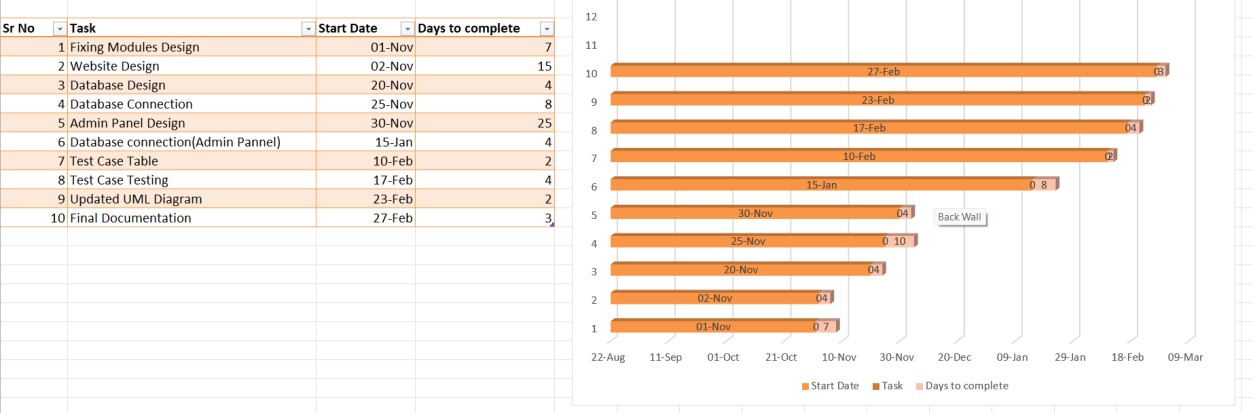
$db="food-order";

$conn = mysqli\_connect($servername,$username,$password,$db); if(!$conn){ die("Connection Failed: ".mysqli\_connect\_error());

}

?>

## 4.8 Coding and Implementation – Gantt Chart



# 5. Conclusion

It has been a great pleasure for me to work on this exciting and challenging project. This experience was actually an astounding entryway for me and I explore every single period of work. This project proved good for me as it provided me knowledge of not only programming language but also about all the handling

procedures related to “ Restaurant Management ”.

The struggle and challenges faced by me during this project development was life learning and will surely help me in future. There is always a scope for improvement and enhancement. During the development of this project, standards are followed for easy maintainability and extensibility.

|  |  |
| --- | --- |
|              | **6. Bibliography**  [www.youtube.com](http://www.youtube.com/) [www.w3school.com](http://www.w3school.com/) [www.stackoverflow.com](http://www.stackoverflow.com/) [www.boxicons.com](http://www.boxicons.com/) [www.getbootstrap.com](http://www.getbootstrap.com/) [www.scrollrevealjs.org](http://www.scrollrevealjs.org/) [www.highcharts.com](http://www.highcharts.com/)  48 |