CP-III Project Report on

# MoneyMingle

**U. V. Patel College of Engineering**



|  |  |
| --- | --- |
| **Guide:** | **Prepared By:** |
| Prof. Hiten Sadani | Yesha Chauhan(21012012008) |
|  |
|  |
|  |

### B. Tech Semester-VII (Information Technology)

July-August, 2023

Submitted to,

Department of Computer Engineering/Information Technology

U.V. Patel College of Engineering Ganpat University, Kherva - 384 012

**U.V. PATEL COLLEGE OF**

**ENGINEERING**



# C E R T I F I C A T E

## TO WHOM SO EVER IT MAY CONCERN

This is to certify that Ms. Yesha Chauhan student of **B.Tech. Semester-VII (Computer Engineering)** has completed her full semester on site project work titled “**MoneyMingle**” satisfactorily in partial fulfillment of the requirement of Bachelor of Technology degree of Computer Engineering of Ganpat University, Kherva, Mehsana in the year 2023-2024.

|  |  |
| --- | --- |
| **Prof. Hiten Sadani** | **Paresh Solanki** |
| **College Project Guide** | **Head, Computer Engineering** |

**ACKNOWLEDGEMENT**

I express our sincere gratitude towards internal guide Prof. Hiten Sadani for her constant help, encouragement, suggestions and inspiration throughout the project work. Without his invaluable advice, suggestions and assistance it would not have been possible for me to complete this project work. This project helped us in doing a lot of research and we came to know about so many new things. We are really thankful to them. We are not only making this project for marks but to also increase our knowledge.

**Thank you, sir**

**Abstract**

In an increasingly interconnected world, managing group finances efficiently has become a necessity. "MoneyMingle: Group Finance Organizer" is a comprehensive software solution designed to simplify and streamline the process of tracking, managing, and optimizing group finances. This project encompasses the development of both web and mobile applications, integrating seamlessly with Firebase for real-time data synchronization.

The primary goal of MoneyMingle is to empower users to create, join, and manage financial groups effortlessly. Users can track expenses, split bills, set budgets, and gain valuable financial insights through user-friendly interfaces accessible via the web and mobile devices.

Key features of the project include user registration and authentication, group creation and management, expense tracking, transaction splitting, and secure data synchronization using Firebase. The mobile application, inspired by the principles of GnuCash Mobile, ensures offline data access, local storage, and powerful reporting capabilities for users on the go.

Security and privacy are paramount concerns, with robust data encryption and access controls in place to safeguard sensitive financial information. The project leverages Firebase for its cloud infrastructure, real-time database, and authentication services, ensuring scalability, reliability, and a seamless cross-platform experience.

MoneyMingle aims to simplify the complexities of group finance management, offering a user-friendly and intuitive solution that empowers individuals and groups to take control of their financial well-being. With a commitment to ease of use, data security, and real-time accessibility, MoneyMingle transforms the way users interact with their finances, providing a powerful tool for financial management in an increasingly collaborative world.

### INDEX

#### Table of Content

Contents

[FaceVibe I](#_Toc145325390)

[B. Tech Semester-VII (Information Technology) I](#_Toc145325391)

[C E R T I F I C A T E II](#_Toc145325392)

[TO WHOM SO EVER IT MAY CONCERN II](#_Toc145325393)

[INDEX 5](#_Toc145325394)

[Fig: 8.4 Waterfall Model 10](#_Toc145325395)

[Fig : 9.1 Reservation page ……………………………………………………………………….36 10](#_Toc145325396)

[Fig :9.2 Our services………………………………………………………………………………37 10](#_Toc145325397)

[Fig : 9.3 Rooms and Rate …………………………………………………………………………38 10](#_Toc145325398)

[Fig :9.4 Contact us Page…………………………………………………………………………...39 10](#_Toc145325399)

[Chapter 1 – Introduction 12](#_Toc145325400)

[1.1 Overview 12](#_Toc145325401)

[1.2 Purpose 12](#_Toc145325402)

[1.2.1Features 12](#_Toc145325403)

[1.3 Scope of Project 13](#_Toc145325404)

[1.4Problem Statement 13](#_Toc145325405)

[1.5 Objective 14](#_Toc145325406)

[Chapter 2 – Literature Survey 15](#_Toc145325407)

[Chapter 3 – Feasibility Study 17](#_Toc145325408)

[3.1 Technical Feasibility: 17](#_Toc145325409)

[3.2 Operational Feasibility: 17](#_Toc145325410)

[3.3Economical Feasibility: 18](#_Toc145325411)

[Chapter 4 – Hardware and Software Requirements: 19](#_Toc145325412)

[4.1 Hardware requirement: 19](#_Toc145325413)

[4.2 Software requirement: 19](#_Toc145325414)

[Chapter 5 – Software Requirement Specification 22](#_Toc145325415)

[5.1 Functional Requirement: 22](#_Toc145325416)

[5.1.1 Customer 22](#_Toc145325417)

[5.1.2 Admin 22](#_Toc145325418)

[5.2 Non-Functional Requirement: 22](#_Toc145325419)

[Chapter 6 – System Design 23](#_Toc145325420)

[6.1 Use case Diagram: 23](#_Toc145325421)

[6.2 Class Diagram: 24](#_Toc145325422)

[6.3 Sequence Diagram: 25](#_Toc145325423)

[6.4 Activity Diagram: 26](#_Toc145325424)

[6.5 State Diagram: 27](#_Toc145325425)

[6.1 ER Diagram 28](#_Toc145325426)

[1 Data Flow Diagram 28](#_Toc145325427)

[6.1.1 Data Flow Diagram-0 28](#_Toc145325428)

[6.1.2 Data Flow Diagram-1 29](#_Toc145325429)

[6.1.3 Data Flow Diagram-2 30](#_Toc145325430)

[7.1 Data Dictionary 30](#_Toc145325431)

[Chapter 8 – Implementation/Prototype 32](#_Toc145325432)

[8.1 Login page: 32](#_Toc145325433)

[Role: Admin wishes to login to the system 32](#_Toc145325434)

[Precondition: Username and Password 32](#_Toc145325435)

[Success end Condition: Main option of screen display 32](#_Toc145325436)

[Failed end Condition: User has entered incorrect Username and 32](#_Toc145325437)

[Password or both 32](#_Toc145325438)

[8.2 Home Page: 32](#_Toc145325439)

[8.3 Destination 33](#_Toc145325440)

[8.4 Testing Model used:- 34](#_Toc145325441)

[Here we have used Waterfall testing model . 34](#_Toc145325442)

[34](#_Toc145325443)

[Fig: 8.4 Waterfall Model 34](#_Toc145325444)

[The sequential phases in Waterfall model are − 34](#_Toc145325445)

[Requirement Gathering and analysis − All possible requirements of the system to be developed are 34](#_Toc145325446)

[captured in this phase and documented in a requirement specification document. 34](#_Toc145325447)

[ System Design − The requirement specifications from first phase are studied in this phase and 34](#_Toc145325448)

[the system design is prepared. This system design helps in specifying hardware and system 34](#_Toc145325449)

[requirements and helps in defining the overall system architecture. 34](#_Toc145325450)

[ Implementation − With inputs from the system design, the system is first developed in small 34](#_Toc145325451)

[programs called units, which are integrated in the next phase. Each unit is developed and tested 34](#_Toc145325452)

[for its functionality, which is referred to as Unit Testing. 34](#_Toc145325453)

[ Integration and Testing − All the units developed in the implementation phase are integrated 34](#_Toc145325454)

[into a system after testing of each unit. Post integration the entire system is tested for any faults 34](#_Toc145325455)

[and failures. 34](#_Toc145325456)

[4 34](#_Toc145325457)

[ Deployment of system − Once the functional and non-functional testing is done; the product is 34](#_Toc145325458)

[deployed in the customer environment or released into the market. 34](#_Toc145325459)

[ Maintenance − There are some issues which come up in the client environment. To fix those 34](#_Toc145325460)

[issues, patches are released. Also to enhance the product some better versions are released. 34](#_Toc145325461)

[Maintenance is done to deliver these changes in the customer environment 34](#_Toc145325462)

[Chapter 9- Implementation Details 35](#_Toc145325463)

[Admin login page code :- 35](#_Toc145325464)

[36](#_Toc145325465)

[Fig : 9.1 Reservation page 36](#_Toc145325466)

[37](#_Toc145325467)

[37](#_Toc145325468)

[Fig :9.2 Our services 37](#_Toc145325469)

[38](#_Toc145325470)

[38](#_Toc145325471)

[Fig : 9.3 Rooms and Rate 38](#_Toc145325472)

[39](#_Toc145325473)

[Fig : 9.4 Contact us page 39](#_Toc145325474)

[Chapter 10 – Conclusion and Further Enhancement 39](#_Toc145325475)

[10.1 Conclusion 39](#_Toc145325476)

[10.2 Future Work 40](#_Toc145325477)

[Reference 40](#_Toc145325478)

#### List of Figure

Figure 6.1 Use case Diagram 20

Figure 6.2 Class Diagram 21

Figure 6.3 Sequence Diagram……………………………………………………………………22

Figure 6.4 Activity Diagram 23

Figure 6.5 State Diagram………………………………………………………………………….24

Figure 6.6.1 Data Flow Diagram-0 25

Figure 6.6.2 Data Flow Diagram-1 26

Figure 6.6.3 Data Flow Diagram-2 27

Figure 8.1 Login Page 30

Figure 8.2 Home page 30

Figure 8.3 Destination page 31

### Fig: 8.4 Waterfall Model

### Fig : 9.1 Reservation page ……………………………………………………………………….36

### Fig :9.2 Our services………………………………………………………………………………37

### Fig : 9.3 Rooms and Rate …………………………………………………………………………38

### Fig :9.4 Contact us Page…………………………………………………………………………...39

### 

### 

### 

*List of Table*

Table 7.1.1 Table for Login\_data………………………………………………………………………… 28

Table 7.1.2 Table for admin\_data………………………………………………………………………… 28

Table 7.1.3 Table for Customer\_data…………………………………………………………………………29

### Chapter 1 – Introduction

## Overview

"MoneyMingle: Group Finance Organizer" is a web and mobile app project that simplifies group finance management. Users can effortlessly create and manage financial groups, track expenses, split bills, set budgets, and gain financial insights. It offers robust security and real-time data synchronization through Firebase, ensuring accessibility and data protection. MoneyMingle aims to transform the way individuals and groups manage finances in an interconnected world.

## 1.2 Purpose

The primary purpose of "MoneyMingle: Group Finance Organizer" is to simplify and streamline the management of group finances. It aims to make it easier for individuals and groups to collaboratively manage their financial activities, track expenses, split bills, set budgets, and gain insights into their financial transactions. The purpose is to provide a user-friendly and efficient solution for financial management within groups, empowering users to take control of their financial well-being in an increasingly interconnected world.

### 1.2.1Features

**Web Application Features:**

1. User Registration and Authentication

2. User Profile Management

3. Dashboard Overview of Financial Accounts

4. Group Creation and Management

5. Group Expense Tracking

6. Transaction Splitting within Groups

7. Rights Management for Adding Transactions

8. Real-time Data Synchronization with Firebase

9. Access to Financial Data Anywhere with an Internet Connection

10. User-friendly Web Interface

**Mobile Application Features**

1. User Registration and Login

2. User Profile Editing

3. Mobile Dashboard with Offline Data

4. Local Storage for Secure Offline Data Access

5. Comprehensive Account Management

6. Transaction Entry and Editing on Mobile

7. Split Transactions with Multiple Categories

8. Powerful Reports and Graphs for Financial Insights

9. Budget Management and Tracking

10. Data Export and Import for Convenience

11. Scheduled Transactions and Reminders

12. Data Encryption and Security Features

13. Cross-Platform Accessibility for Seamless Experience

14. Synchronization with Firebase for Offline Data

15. Firebase Integration for Real-time Data Exchange

**Firebase Data Server API Technical Features:**

1. Firebase Realtime Database for Data Storage and Retrieval

2. Firebase Authentication for Secure User Access

3. Firebase Cloud Functions for Server-Side Logic

4. Data Validation and Error Handling with Firebase Rules

5. Scalable and Reliable Firebase Cloud Infrastructure

6. Firebase Security Rules for Data Privacy

7. Logging and Monitoring through Firebase Analytics

8. Firebase Hosting for Web Application Deployment

9. Version Control for Firebase Cloud Functions

10. Comprehensive Firebase Documentation for Developers

11. Firebase Rate Limiting and Access Controls

12. Data Backup and Restore Using Firebase Services

These features collectively provide a comprehensive financial management and collaboration platform for users, offering real-time data synchronization, data security, and a user-friendly interface across both web and mobile platforms.

## 1.3 Scope of Project

The scope of the **"MoneyMingle: Group Finance Organizer"** project is substantial and encompasses various aspects of financial management and collaboration. It aims to provide users with a robust set of features and capabilities for tracking, managing, and optimizing group finances. Below is an overview of the project's scope:

1. **Group Finance Management:** MoneyMingle allows users to create and manage financial groups, making it suitable for families, roommates, friends, or small businesses looking to collaboratively manage their finances.

2**.Expense Tracking:** Users can track expenses within their groups, categorize them, and assign them to specific members. This feature is essential for keeping a record of all financial activities.

3.**Transaction Splitting:** MoneyMingle facilitates the splitting of expenses within groups, simplifying the process of sharing costs among members. It calculates who owes how much to whom, streamlining financial transactions.

4. **User Registration and Authentication**: The project includes user registration and authentication features to ensure secure access to financial data.

5. **User Profiles**: Users can manage their profiles, which may include personal information, preferences, and settings.

6. **Data Synchronization**: Real-time data synchronization with Firebase ensures that users have up-to-date financial information accessible across devices. This feature is crucial for collaborative finance management.

7. **Mobile Application**: MoneyMingle offers a mobile application inspired by the GnuCash Mobile app. It provides offline data access, local storage, and comprehensive financial management capabilities on the go.

8. **Security and Privacy**: The project places a strong emphasis on data security and privacy, implementing data encryption and access controls to protect sensitive financial information.

9. **Dashboard and Reports**: Users can access a dashboard that provides an overview of their financial accounts, groups, and recent transactions. The project also offers powerful reports and graphs for gaining financial insights.

10. **Budget Management:** Users can set and track budgets, helping them manage their spending and financial goals effectively.

11. **Data Export and Import**: MoneyMingle allows users to export and import financial data, providing convenience and backup options.

12. **Firebase Integration:** The project leverages Firebase's cloud infrastructure, real-time database, authentication services, and other features to ensure scalability, reliability, and a seamless cross-platform experience.

13. **Comprehensive Documentation:** MoneyMingle provides comprehensive documentation for developers, making it easier to work with the project's technical features.

14. **Web and Mobile Accessibility:** MoneyMingle is accessible on both web and mobile platforms, ensuring users can manage their finances from anywhere with an internet connection.

The scope of the project is ambitious, as it aims to provide a holistic financial management solution with a strong focus on user-friendliness, security, and real-time accessibility. Its potential applications span personal finance management, shared household expenses, group travel, and collaborative small business finance management.

## 1.4 Problem Statement

In collaborative finance management, people face inefficiencies in tracking expenses, splitting bills, and gaining financial insights. This leads to disorganization, errors, and limited financial control.

**Solution Approach:**

**"MoneyMingle: Group Finance Organizer**" offers an intuitive platform for effortless group finance management, expense tracking, fair bill splitting, and insightful financial reporting. It ensures real-time data synchronization, empowering users to take control of their finances efficiently and collaboratively.

##### **1.5 Objective**

"MoneyMingle: Group Finance Organizer" aims to simplify group finance management by providing user-friendly tools for expense tracking, bill splitting, and financial insights. It ensures real-time data synchronization, prioritizes security, and transforms how users interact with their finances.

### Chapter 2 – Literature Survey

##### Introduction

Literature review is a part of a process of reading, analyzing, evaluating and summarizing scholarly material about a specific topic. This is use as a guideline to develop a new system. Therefore, the new system can provider better system and more functional from other system. In addition, it also including the existing system and comparison the current system with new system to overcome the weakness of the system.

##### Literature of Review

The literature review for the "MoneyMingle: Group Finance Organizer" project provides an overview of existing research, software solutions, and relevant concepts in the field of personal and group finance management. It encompasses studies, articles, and software applications that address similar challenges and provide insights into collaborative financial management.

**1. Personal Finance Management Software**: Numerous personal finance management software solutions exist, such as Mint, YNAB (You Need A Budget), and Quicken. These applications offer individual users tools for budgeting, expense tracking, and financial planning.

**2. Group Expense Management Applications**: Some applications, like Splitwise and Settle Up, focus on simplifying group expense tracking and bill splitting. These apps are often used by roommates, travelers, and friends to manage shared expenses.

**3. Financial Collaboration Tools**: Slack, Trello, and other collaboration tools have been adapted for financial purposes within groups or small businesses. These platforms enable users to discuss finances, track expenses, and collaborate on financial decisions.

**4. Research on Collaborative Finance Management:** Academic research explores various aspects of collaborative finance management within groups and small businesses. Studies highlight the importance of efficient expense tracking, equitable bill splitting, and real-time data synchronization in group finance.

**5. Firebase Realtime Database**: Firebase, a cloud-based platform, is commonly used for real-time data synchronization in web and mobile applications. Its features are crucial for achieving real-time collaboration in financial management apps.

**6. GnuCash Principles:** GnuCash, an open-source financial accounting software, serves as an inspiration for the project's mobile application. GnuCash Mobile offers offline access, local storage, and reporting capabilities, providing valuable insights.

### 

### Chapter 3 – Feasibility Study

Feasibility Study for "MoneyMingle: Group Finance Organizer" Project

A feasibility study assesses the viability of a project, taking into account various aspects such as technical, economic, operational, scheduling, and legal considerations. The feasibility study for the "MoneyMingle" project is as follows:

**1. Technical Feasibility:**

- Technology Stack: The project relies on well-established technologies such as Django (for the web application), Firebase (for real-time data synchronization), and cross-platform mobile app development frameworks. These technologies are readily available and have a strong developer community.

- Development Expertise: The team possesses the necessary technical skills and experience in web and mobile application development, including familiarity with Firebase integration.

- Scalability: Firebase offers scalable cloud infrastructure, ensuring the system can handle increased data and user load as the project grows.

**2. Economic Feasibility:**

- Cost-Benefit Analysis: A cost-benefit analysis indicates that the potential benefits of the project, such as user subscriptions or premium features, outweigh the development and operational costs.

- Revenue Generation: The project has the potential to generate revenue through various monetization strategies, including premium features, advertising, or subscription models.

**3. Operational Feasibility:**

- User Base: The project targets a wide user base, including individuals, families, friends, and small businesses. The demand for collaborative finance management solutions exists.

- User Experience: A user-friendly interface and intuitive features will enhance the project's operational feasibility, making it accessible to users with varying levels of financial expertise.

**4. Scheduling Feasibility:**

- Project Timeline: The project timeline is reasonable, allowing for development, testing, and deployment phases. Adequate time is allocated for each aspect of the project.

- Milestones: Clear milestones are defined to track progress, and contingency plans are in place to address unexpected delays.

**5. Legal and Regulatory Feasibility:**

- Data Privacy and Security: Stringent measures are in place to ensure data privacy and security, aligning with legal and regulatory requirements such as GDPR (General Data Protection Regulation).

- Intellectual Property: The project respects intellectual property rights and licenses, adhering to open-source principles where applicable.

### Chapter 4 – Hardware and Software Requirements:

## 4.1 Hardware requirement:

The hardware and software requirements for the "MoneyMingle: Group Finance Organizer" project include those necessary for both development and deployment of the web and mobile applications. Below is an overview of the requirements:

**Hardware Requirements:**

**1. Development Workstations:**

- Desktop or laptop computers for developers.

- Adequate processing power and memory for coding and testing.

- Operating System: Windows, macOS, or Linux.

**2. Server Hardware (for Deployment):**

- A cloud server or hosting provider for deploying the web application.

- Sufficient processing power, memory, and storage for web hosting.

- Scalability options to accommodate growing user base.

**3. Mobile Devices (for Mobile App Testing):**

- iOS and Android devices for testing the mobile applications.

- Different screen sizes and resolutions for comprehensive testing.

**Software Requirements:**

**1. Development Tools:**

- Integrated Development Environment (IDE) such as Visual Studio Code, PyCharm, or similar for web application development.

- Mobile app development tools for iOS (Xcode) and Android (Android Studio).

- Version control system (e.g., Git) for code management.

**2. Web Application (Backend):**

- Programming Language: Python

- Web Framework: Django

- Database Management System: PostgreSQL or another relational database system

- Web Server: Gunicorn or similar

- Real-time Data Synchronization: Firebase

**3. Web Application (Frontend):**

- HTML, CSS, JavaScript

- Frontend Framework: Vue.js, React, or Angular

- User Interface (UI) Libraries: Bootstrap, Material-UI, or similar

- AJAX or Fetch for making asynchronous requests to the server.

**4. Mobile Application (Frontend):**

- Mobile App Development Frameworks: React Native, Flutter, or native development (Swift for iOS, Kotlin for Android)

- UI Components: Native or third-party libraries for building mobile interfaces.

- Real-time Data Synchronization: Firebase SDK for mobile.

**5. Database Requirements:**

- Relational Database Management System (RDBMS) like PostgreSQL or MySQL for storing user and financial data.

- Firebase Realtime Database for real-time data synchronization between web and mobile applications.

**6. Authentication and Security:**

- Firebase Authentication for user authentication and authorization.

- Encryption libraries or mechanisms for data security.

**7. Web Hosting:**

- Hosting environment for the web application (e.g., AWS, Heroku, Google Cloud, or a similar service).

- Domain and SSL certificate for secure web access.

**8. Mobile App Testing:**

- Emulators and simulators for testing on various iOS and Android devices.

- Mobile device testing on real hardware.

**9. Documentation and Collaboration Tools:**

- Documentation tools for creating user manuals, guides, and technical documentation.

- Collaboration and communication tools (e.g., Slack, project management tools) for the development team.

**10. Operating Systems:**

- Ensure compatibility with various operating systems (Windows, macOS, iOS, Android) for both development and deployment.

These hardware and software requirements provide a foundation for developing and deploying the "MoneyMingle" project, encompassing web and mobile applications. Careful consideration of these requirements will help ensure the project's success and scalability.

### Chapter 5 – Software Requirement Specification

Software requirement specification is the activity of eliciting, analyzing and recording requirement for software system. It shares much in common with requirement analysis for more general kinds of systems.

**5.1 Functional Requirement:**

The system will be able to display the necessary information of the user. The system should allow the administrator and special user to add responses to the database.

Certainly, here are the functional requirements for the "MoneyMingle: Group Finance Organizer" project in a concise format:

**User Registration and Authentication:**

- Users can create an account with email and password.

- Account confirmation emails are sent for verification.

- Users can reset passwords if forgotten.

- Registered users can securely log in.

**User Profile Management:**

* Users can edit their profiles, including name and contact details.

**Group Creation and Management:**

- Users can create financial groups with names and descriptions.

- Groups can be set to private or public.

- Users can invite others to join their groups.

**Expense Tracking and Management:**

- Users record expenses with dates, payees, amounts, and categories.

- Expenses can be assigned to specific group members.

- Users can categorize expenses into predefined or custom categories.

- Expenses can include details and document attachments.

**Transaction Splitting:**

- Users can split transactions among group members.

- System calculates shares based on predefined rules.

- Multiple split methods are supported.

**Financial Reporting and Insights:**

- Users access a dashboard with financial overviews.

- Transaction history displays all group transactions.

- Users generate reports and graphs for insights.

- Budgets are set, tracked, and monitored.

- Notifications warn of budget exceedances.

**Data Synchronization and Accessibility:**

- Firebase ensures real-time data sync across web and mobile.

- Mobile app provides offline data access.

- Changes sync when online.

**Security and Privacy:**

- Data encryption for transit and at rest.

- Access controls and permissions for group data.

- Secure Firebase Authentication.

- User data privacy compliance.

**User-Friendly Interfaces:**

- Responsive web and mobile interfaces.

- Intuitive navigation and guidance.

- Help features for users.

**User Notifications:**

- Notifications for financial events.

- Notifications accessible in a dedicated center.

**Integration with Firebase Services:**

- Firebase Realtime Database for data storage.

- Firebase Authentication for secure user access.

**Budget Management:**

- Users create budgets with names and timeframes.

- The system monitors actual expenses against budgets.

- Budget reports and variances are provided.

**Data Export and Import:**

- Users export financial data in common formats.

- Data imports from external files are supported.

**Scheduled Transactions and Reminders:**

- Users create and manage scheduled transactions.

- Reminders for upcoming transactions are sent.

**Data Encryption and Security Features:**

- Data transmitted securely over HTTPS.

- Compliance with data privacy regulations.

- Option for two-factor authentication (2FA).

**Cross-Platform Accessibility:**

- Web app compatible with major browsers.

- Mobile app compatible with iOS and Android.

**Synchronization with Firebase for Offline Data:**

- Web app offers limited offline data access.

- Mobile app syncs data automatically when online.

These concise functional requirements outline the core features and capabilities of the "MoneyMingle" project, ensuring effective group finance management and user-friendly experiences.

### Chapter 6 – System Design

## 6.1 Use case Diagram:

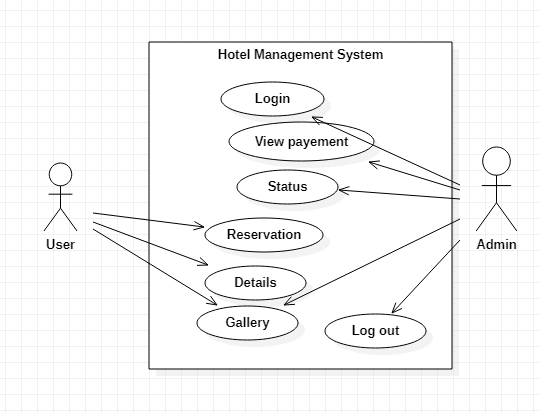


Figure 6.1 Use Case Diagram

## 6.2 Class Diagram:

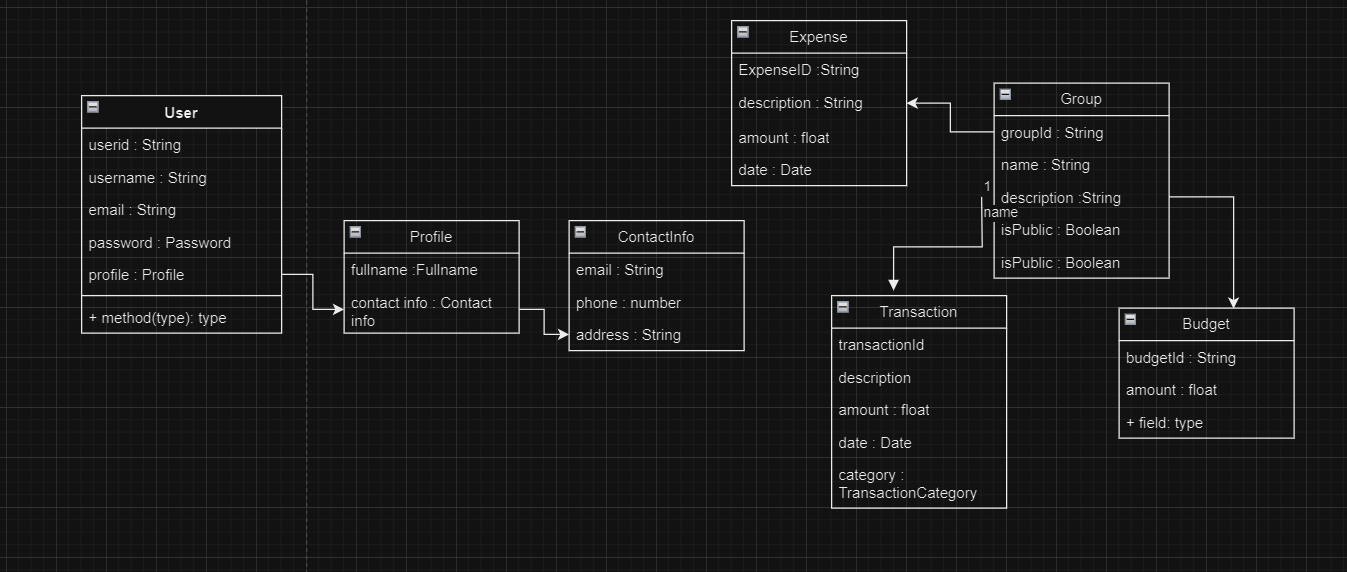
****

Figure 6.2 Class Diagram

## 6.3 Sequence Diagram:

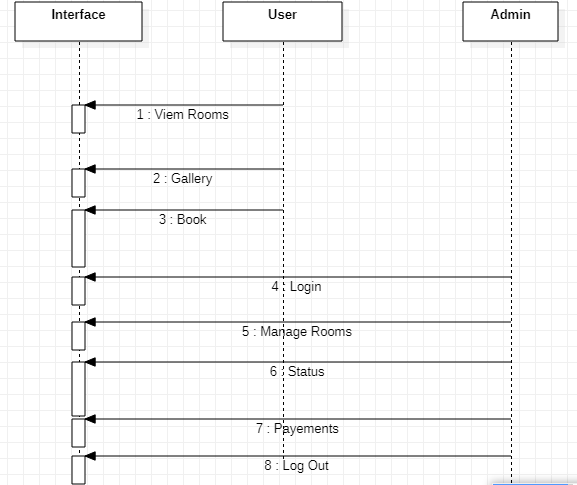


Figure 6.3 Sequence Diagram

## 6.4 Activity Diagram:

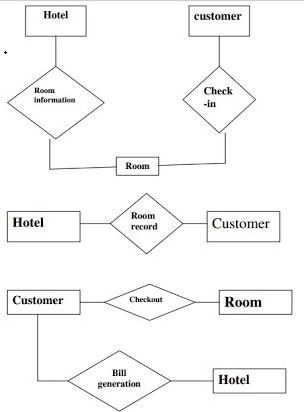


Figure 6.4 Activity Diagram

## 6.5 State Diagram:

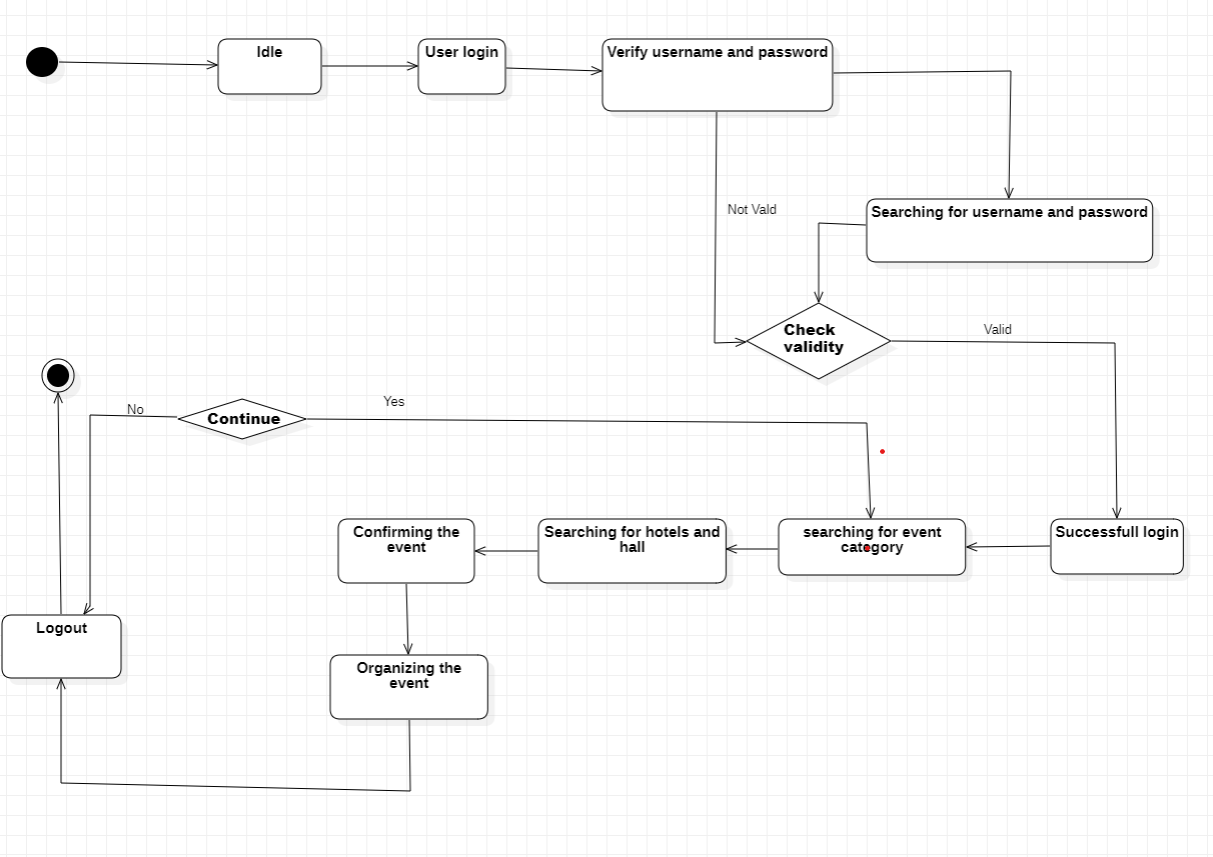
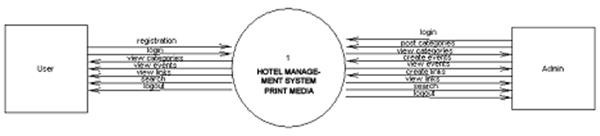


Figure 6.5 State Diagram

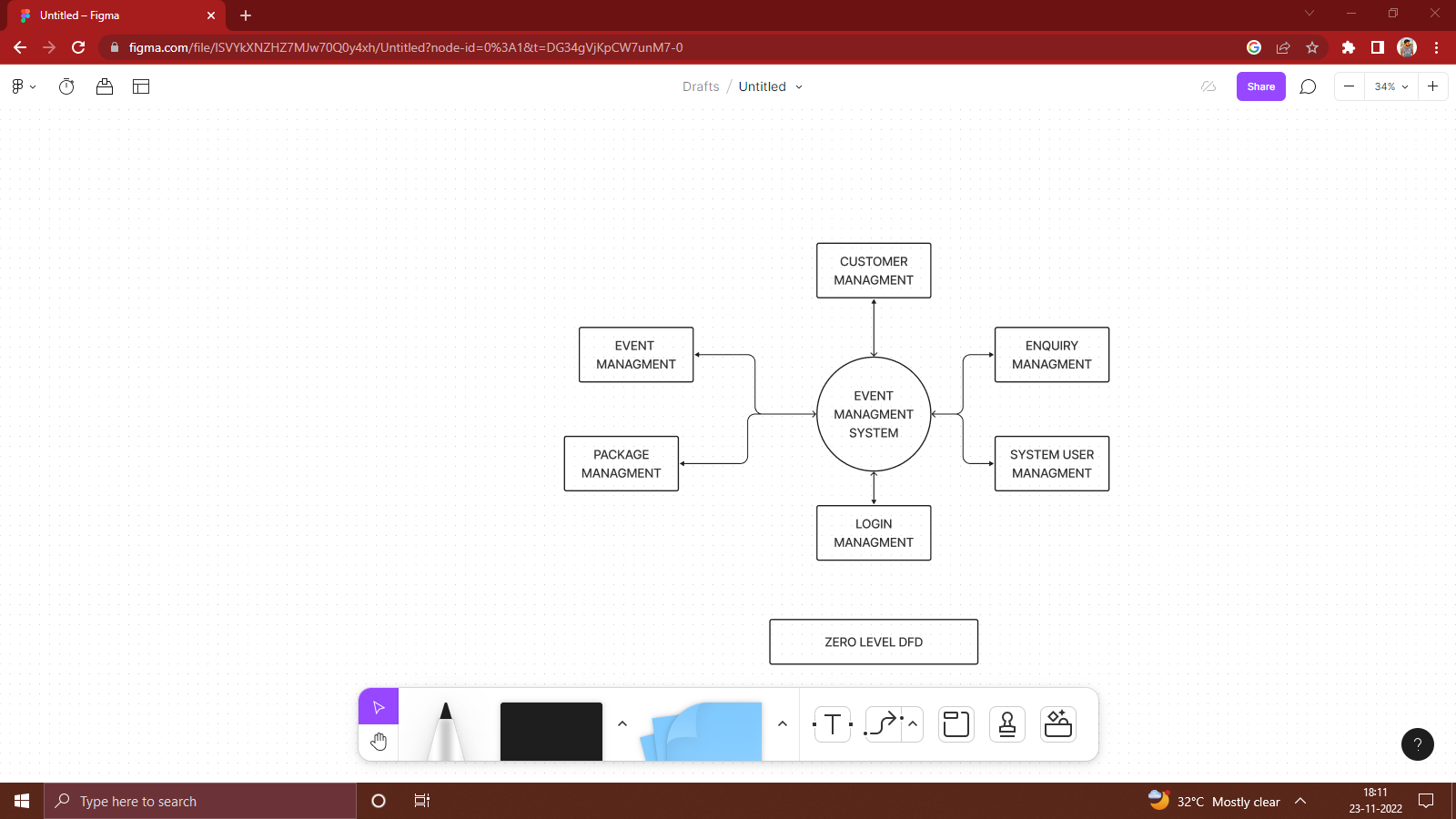
## 6.1 ER Diagram

### 

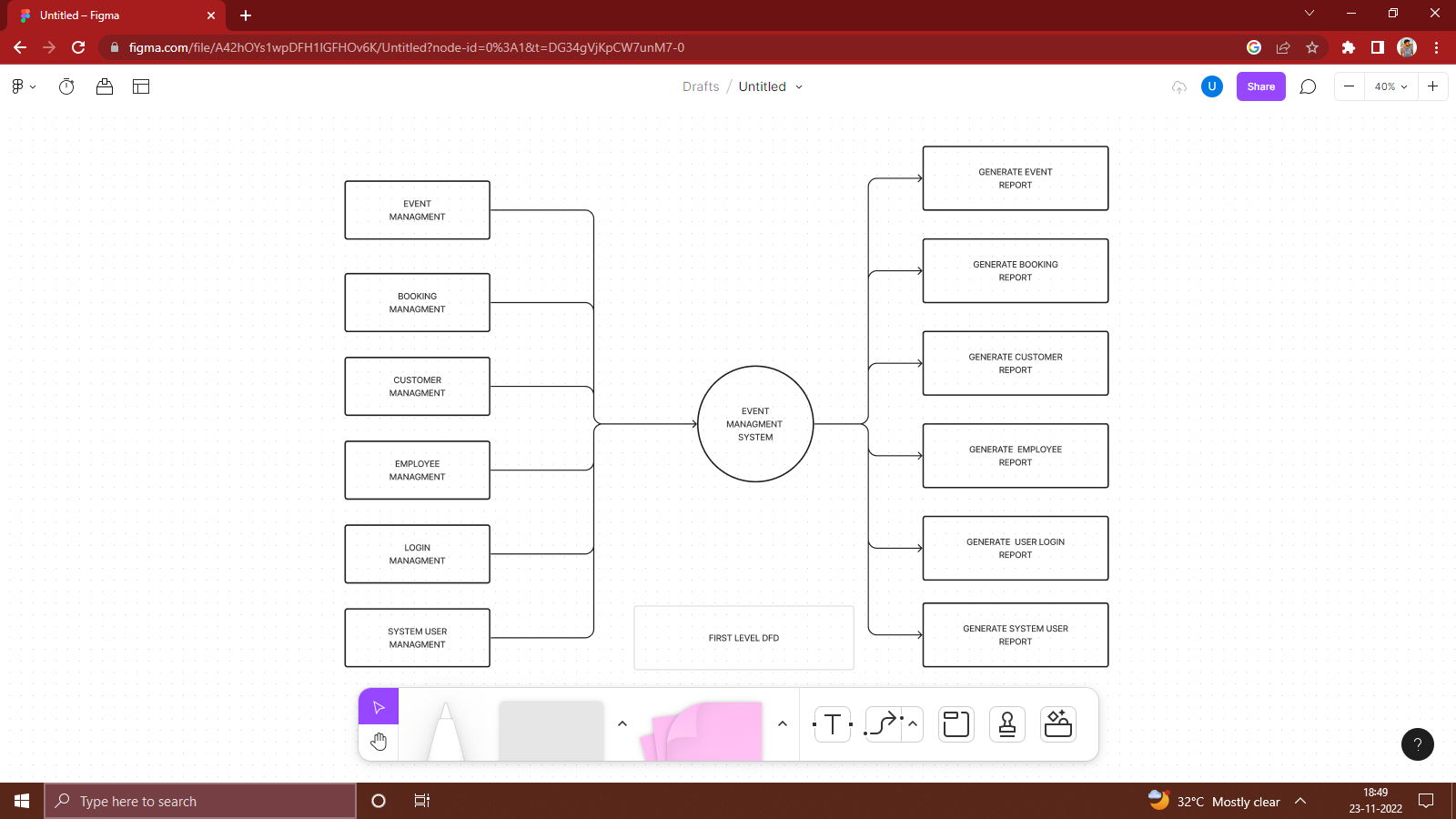


## 1 Data Flow Diagram

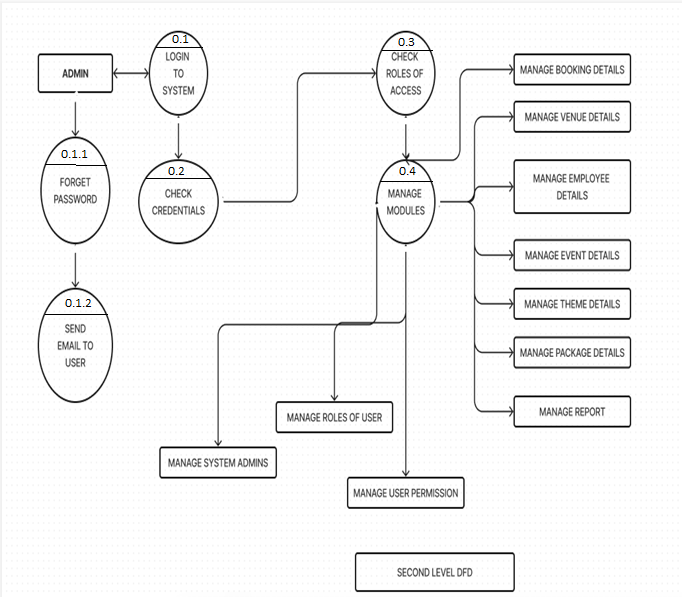
### 6.1.1 Data Flow Diagram-0



### 6.1.2 Data Flow Diagram-1



### 6.1.3 Data Flow Diagram-2

. 

Chapter 7 – Database Strategy

## 7.1 Data Dictionary

### Chapter 8 – Implementation/Prototype

## 8.1 Login page:

## Role: Admin wishes to login to the system

## Precondition: Username and Password

## Success end Condition: Main option of screen display

## Failed end Condition: User has entered incorrect Username and

## Password or both

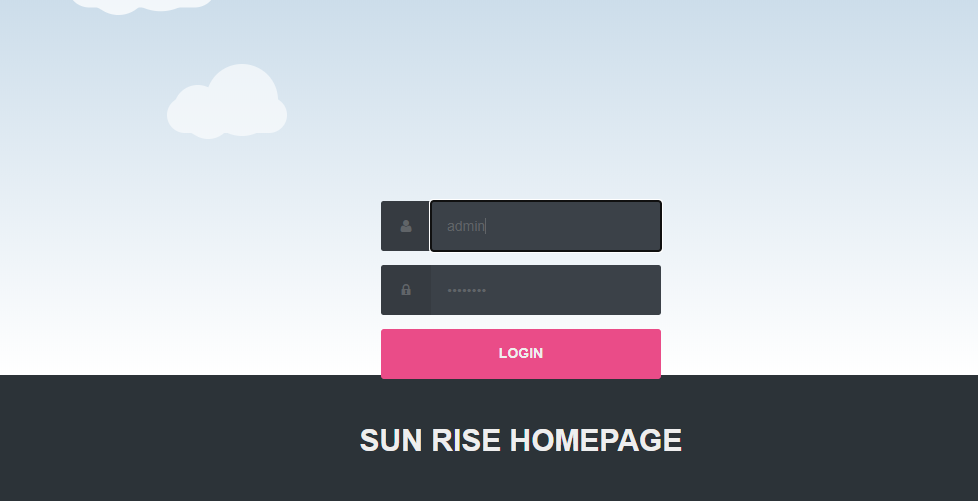
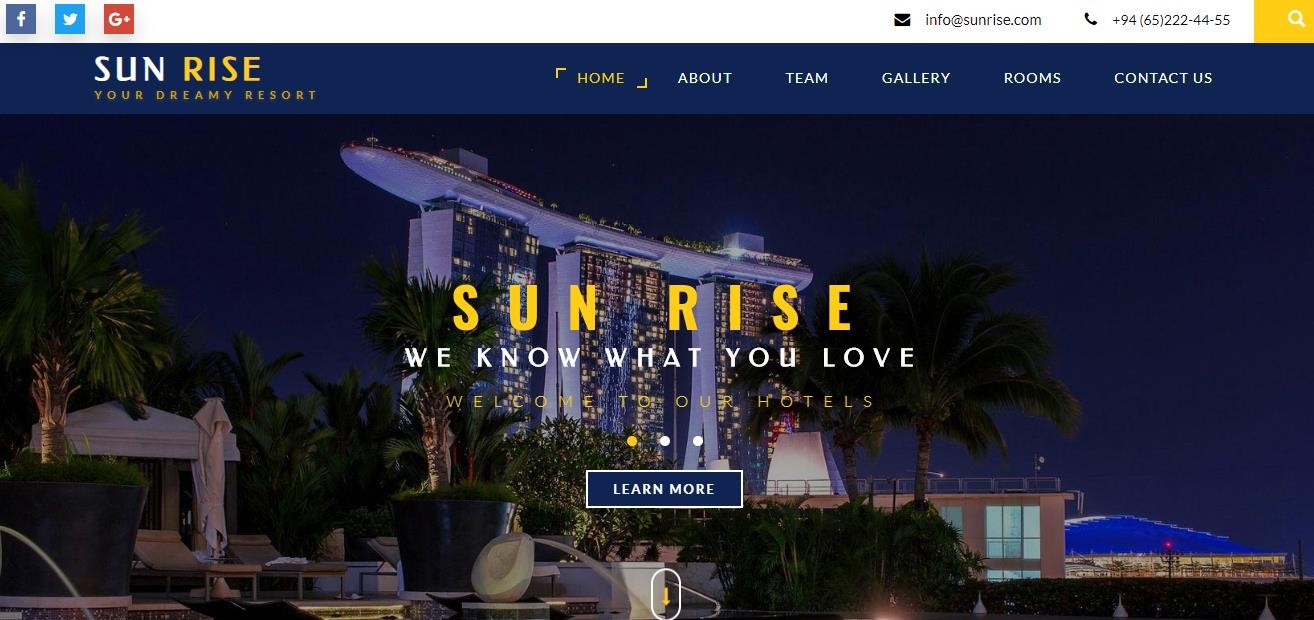


Fig:8.1 home page

## 

## 8.2 Home Page:

Can view Hotels and Hall and can book as per their convenience

Fig: 8.2 Home page

## 8.3 Destination

Customer can book the rooms according to there needs through online reservation system and they can check the availability .

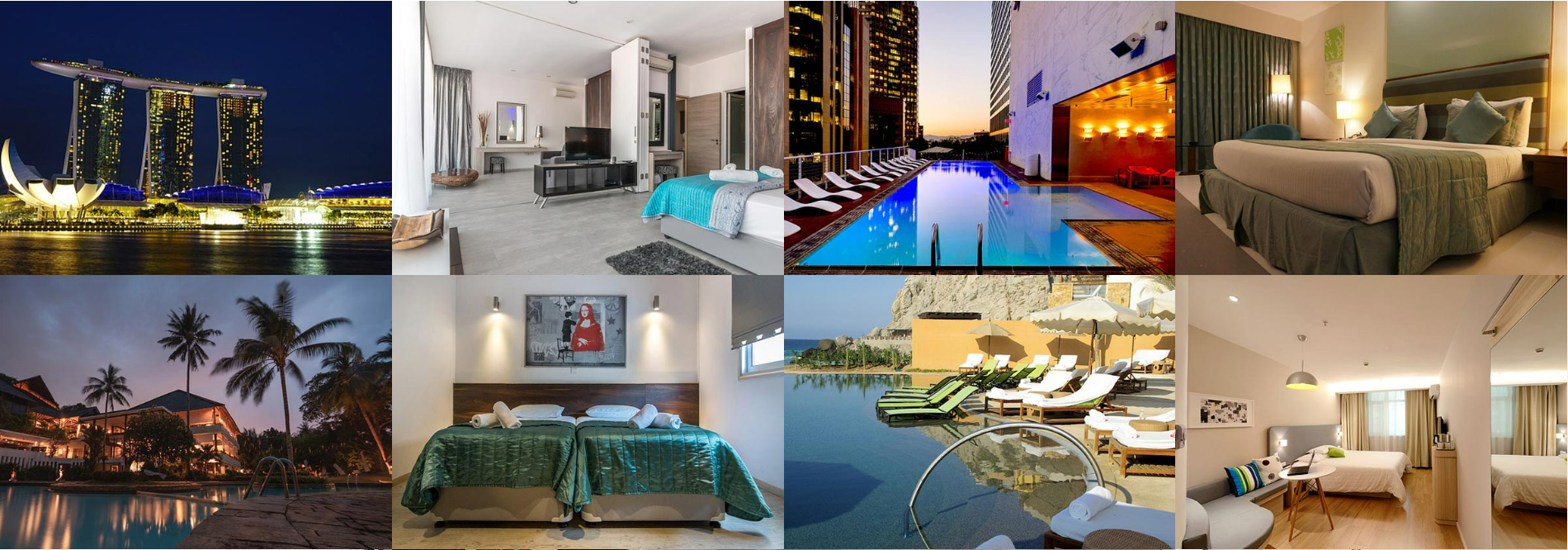


Fig: 8.3 Destination Page

### 8.4 Testing Model used:-

### Here we have used Waterfall testing model .

### 

### Fig: 8.4 Waterfall Model

### 

### The sequential phases in Waterfall model are −

### Requirement Gathering and analysis − All possible requirements of the system to be developed are

### captured in this phase and documented in a requirement specification document.

###  System Design − The requirement specifications from first phase are studied in this phase and

### the system design is prepared. This system design helps in specifying hardware and system

### requirements and helps in defining the overall system architecture.

###  Implementation − With inputs from the system design, the system is first developed in small

### programs called units, which are integrated in the next phase. Each unit is developed and tested

### for its functionality, which is referred to as Unit Testing.

###  Integration and Testing − All the units developed in the implementation phase are integrated

### into a system after testing of each unit. Post integration the entire system is tested for any faults

### and failures.

### 4

###  Deployment of system − Once the functional and non-functional testing is done; the product is

### deployed in the customer environment or released into the market.

###  Maintenance − There are some issues which come up in the client environment. To fix those

### issues, patches are released. Also to enhance the product some better versions are released.

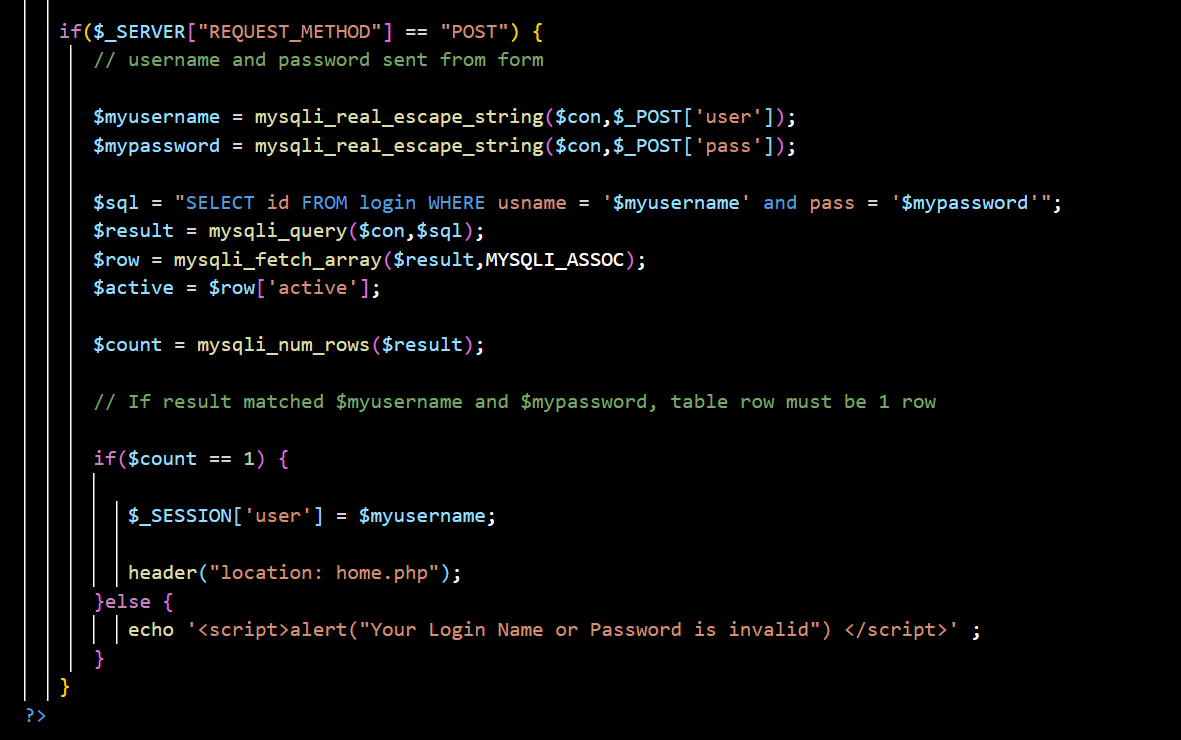
### Maintenance is done to deliver these changes in the customer environment

### Chapter 9- Implementation Details

### 

### Admin login page code :-





### 

### 

### 

### Fig : 9.1 Reservation page

### 

### 

### Fig :9.2 Our services

### 

### 

### Fig : 9.3 Rooms and Rate

### 

### Fig : 9.4 Contact us page

### 

### Chapter 10 – Conclusion and Further Enhancement

## 10.1 Conclusion

In conclusion, FaceVibe represents a groundbreaking fusion of technology and music, where emotions take center stage. This innovative mobile application offers a profound shift in how we connect with music, seamlessly adapting playlists to our ever-changing emotional landscapes. With its real-time emotion analysis, mood-centric recommendations, and unwavering commitment to user privacy, FaceVibe not only enhances the way we experience music but also deepens our personal connection to it

## 10.2 Future Work

The future work based on below:

* + 1. Convert Website into Mobile Application
    2. Suggestion Feature for better experience.
    3. Addition of credit points for customers.
    4. Tour packages for customers .

# 

# Reference

<https://www.freeprojectz.com/>

[Event Management System Document (slideshare.net)](https://www.slideshare.net/AAKASHPANCHAL2/event-management-system-document)

[img-0207 -IT.pdf (kiu.ac.ug)](https://ir.kiu.ac.ug/bitstream/20.500.12306/8070/1/img-0207%20-IT.pdf)

[Srs Event Management System | PDF | Databases | Operating System (scribd.com)](https://www.scribd.com/document/500793536/srs-event-management-system)