**SOFTWARE DESIGN DOCUMENT**

**HOMELY FOOD DELIVERY SYSTEM**

Team members

Dona Jacob(25)

Hala P(32)

Nanditha P(49)

Shana Sherin M(57)

Contents

1. Introduction

1.1. Project Overview

1.2. Project Scope

2. System Architecture Design

3. Application Architecture Design

4. User Interface Design

4.1. GUI Designs (Mockups)

5. API Design

6. Database Design

7. Technology Stack

8. Conclusion

# Introduction

## Purpose

The purpose of this Software Design document is to provide a comprehensive overview of the software design and architecture for our system, which helps to provide homely meals who are in need especially travelers and those people with health problems.

## Product Scope

The scope of the project includes the development of a user-friendly web application that enables users to:

* Browse and select homely meal options based on their preferences and dietary requirements
* View nutritional information, pricing, and delivery options for each meal option
* Provide feedback on meal quality and delivery experience
* Create and manage user profiles and payment methods
* Allow homemakers to register as vendors and offer their homely meal options to customers
* Allow vendors to manage their product offerings, pricing, and delivery options

# System Architecture Design

1. System Components:

The system components of the homely food delivery system application include the following:

* Frontend: The frontend of the application will be developed using Flutter, a cross-platform UI toolkit(Figma) to create user-friendly screens, widgets, navigation, and state management for the mobile application.
* Backend: The backend of the application will consist of APIs for handling various functionalities such as user authentication, meal listings, order management, and payment processing. Technologies like Django will be used for server-side development, while Firebase, a cloud-based backend-as-a-service (BaaS) platform, will be used for hosting and managing the backend APIs and databases.
* Third-party Services: The application will also integrate with third-party services such as payment gateways for handling online payments, mapping services for displaying delivery locations, and authentication providers for secure user authentication.

1. System Communication:

The communication between the frontend and backend components of the application will be through RESTful APIs. The frontend will send HTTP requests to the backend APIs for retrieving meal listings, placing orders, and managing user profiles. The backend APIs will respond with JSON or XML data formats, which will be parsed by the frontend to display the relevant information to the users.

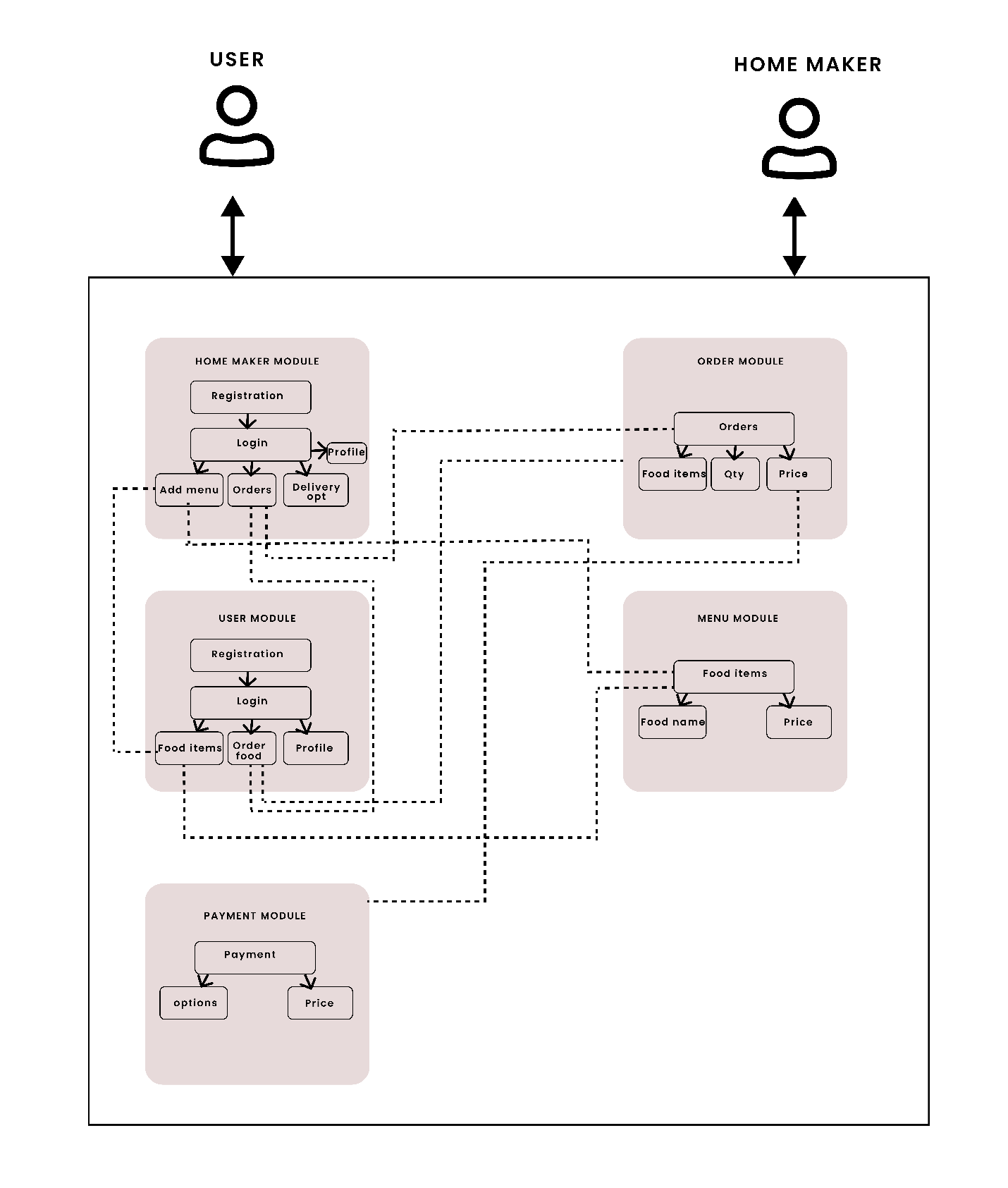
1. Security:

To ensure the security of the homely food delivery system application, the following measures will be implemented:

* Authentication and Authorization: Implementing secure authentication and authorization mechanisms to protect user accounts and ensure that only authorized users can access the application.
* Data Encryption: Encrypting sensitive data such as user passwords and payment information to protect against data breaches.
* Secure Communication: Implementing HTTPS and SSL/TLS protocols for secure communication between the frontend and backend components.

**3.Application Architecture Design**

1. User Module: The User Module is responsible for managing the authentication and registration of users in the system. It allows users to create a new account, log in with their credentials, and manage their personal information, such as their name, email address, and password. This module also handles the authorization of users, determining what actions they are allowed to perform within the system.
2. Homemaker Module: The Homemaker Module is responsible for managing the accounts of homemaker users in the system. It allows homemaker users to create and update their profile, add and manage menu items, and view and manage orders placed by users. This module also includes functionalities for setting up delivery locations, managing reviews, and communicating with users.
3. Order Module: The Order Module is responsible for managing orders placed by users in the system. It allows users to place an order, view the status of their order, and cancel the order if necessary. This module also includes functionalities for managing payment information and generating receipts.
4. Menu Item Module: The Menu Item Module is responsible for managing the menu items added by homemaker users in the system. It allows homemaker users to add new menu items, update existing menu items, and delete menu items if necessary. This module also includes functionalities for managing the price, ingredients, and availability of menu items.



**4. User Interface Design**

4.1. GUI Designs

Tool used to design user interface : Figma

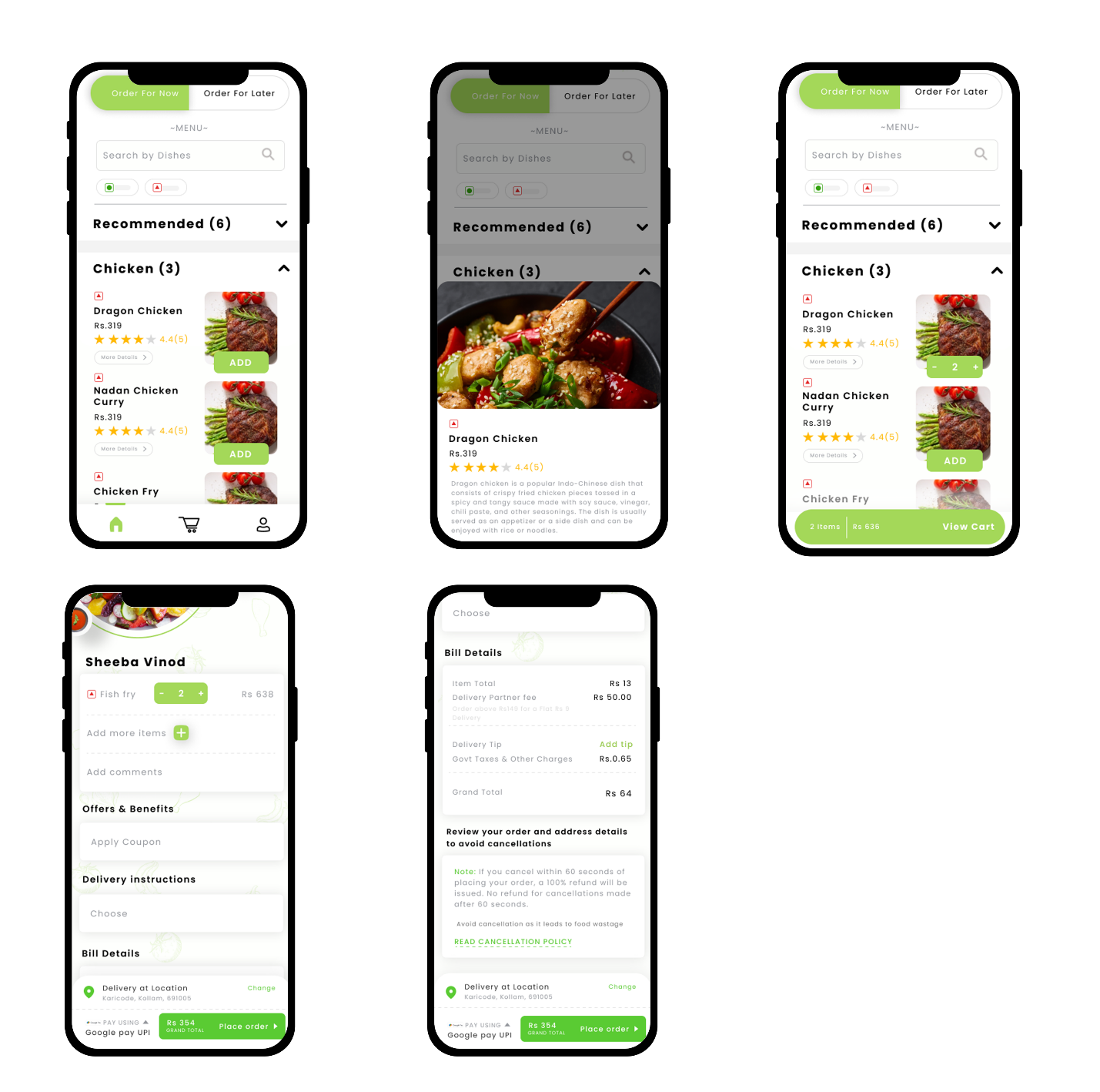
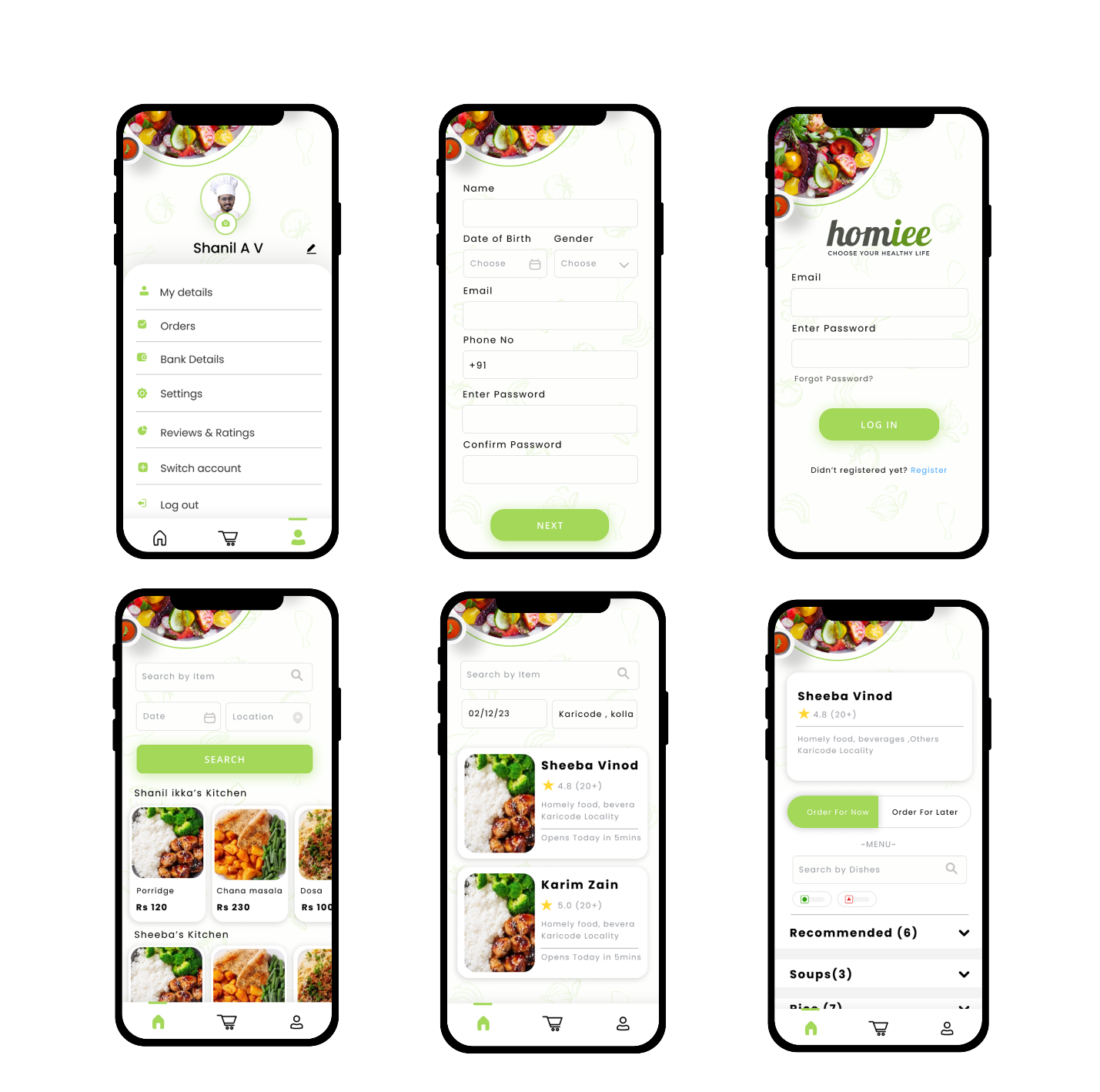
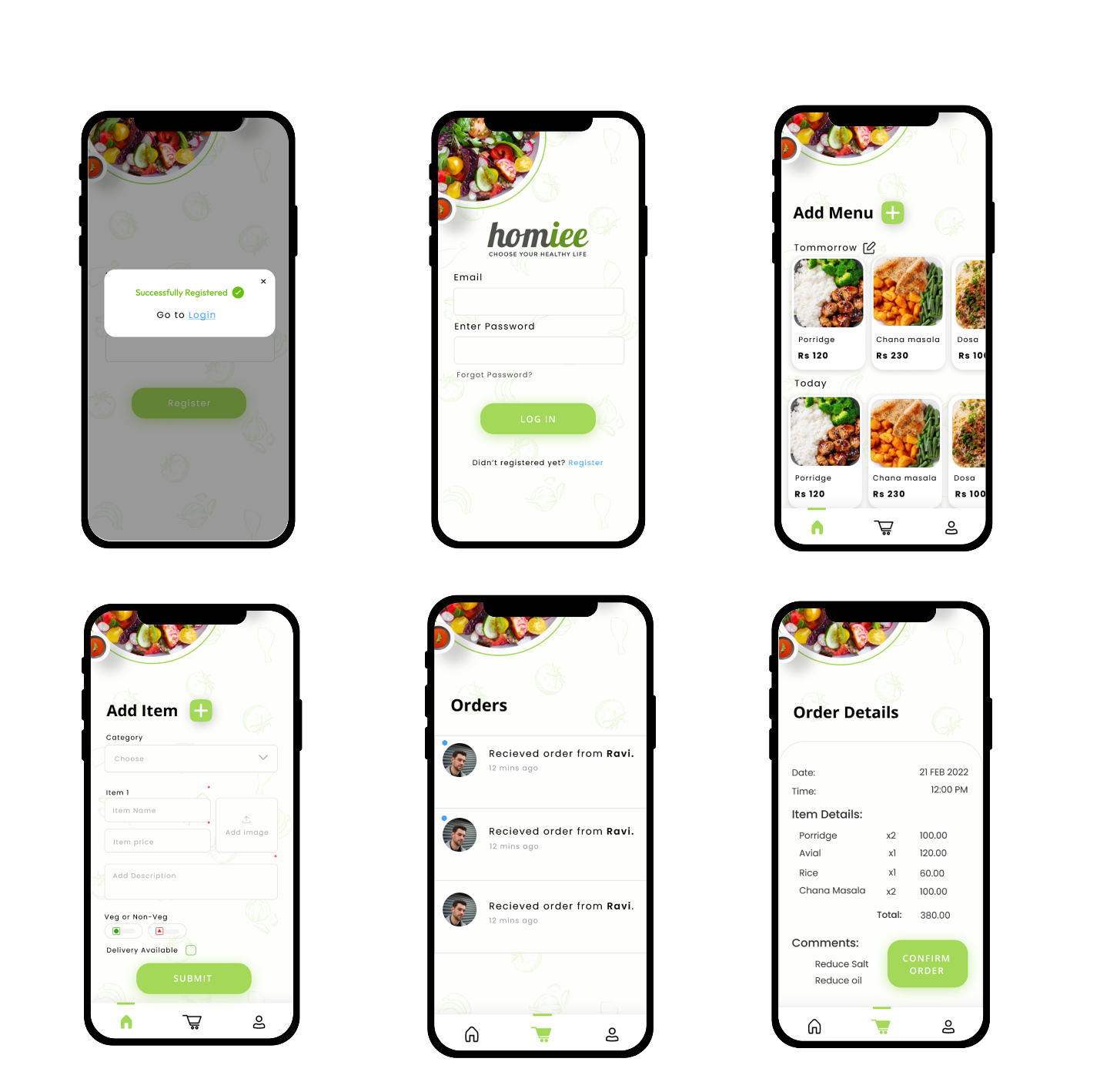
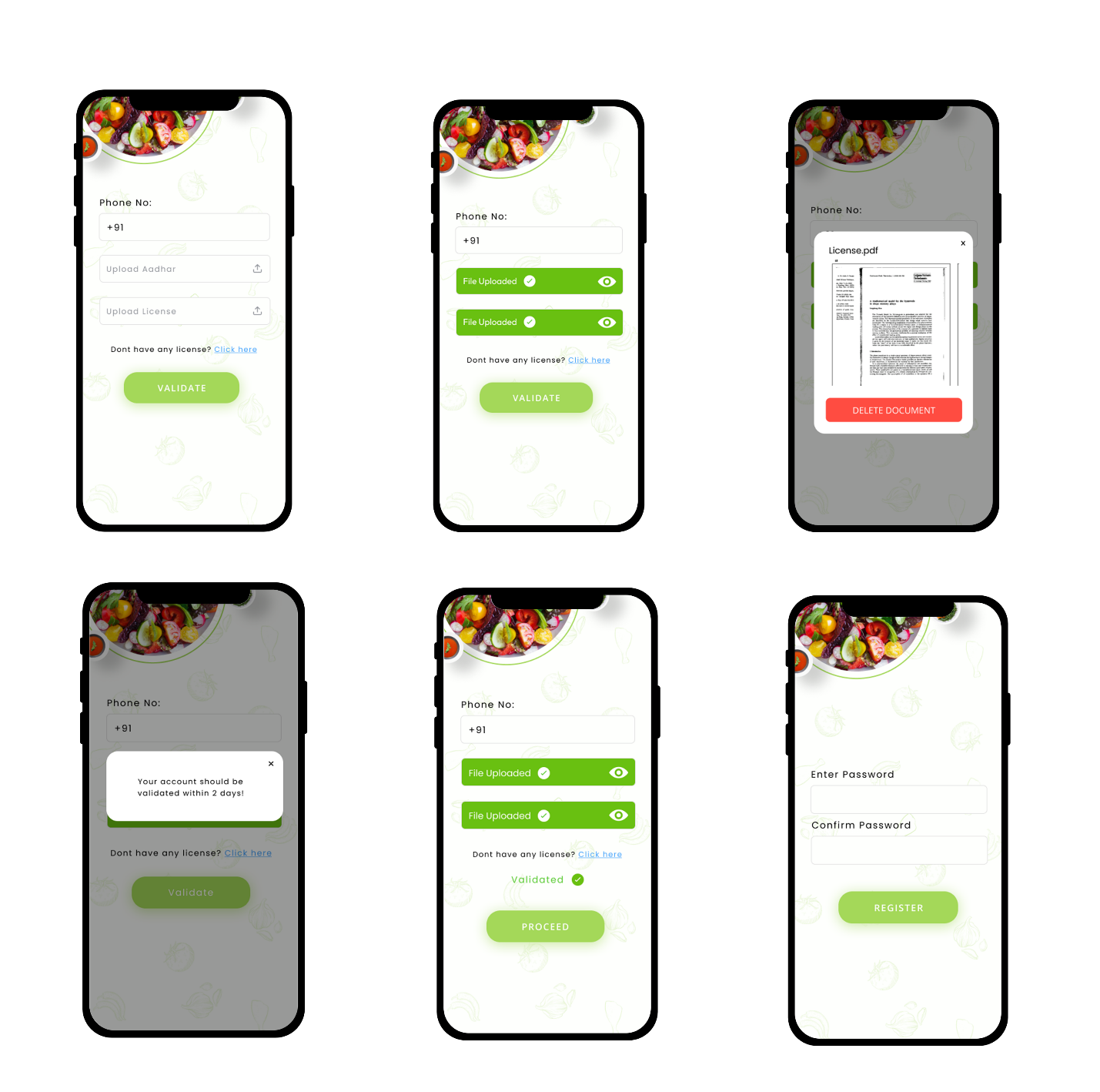
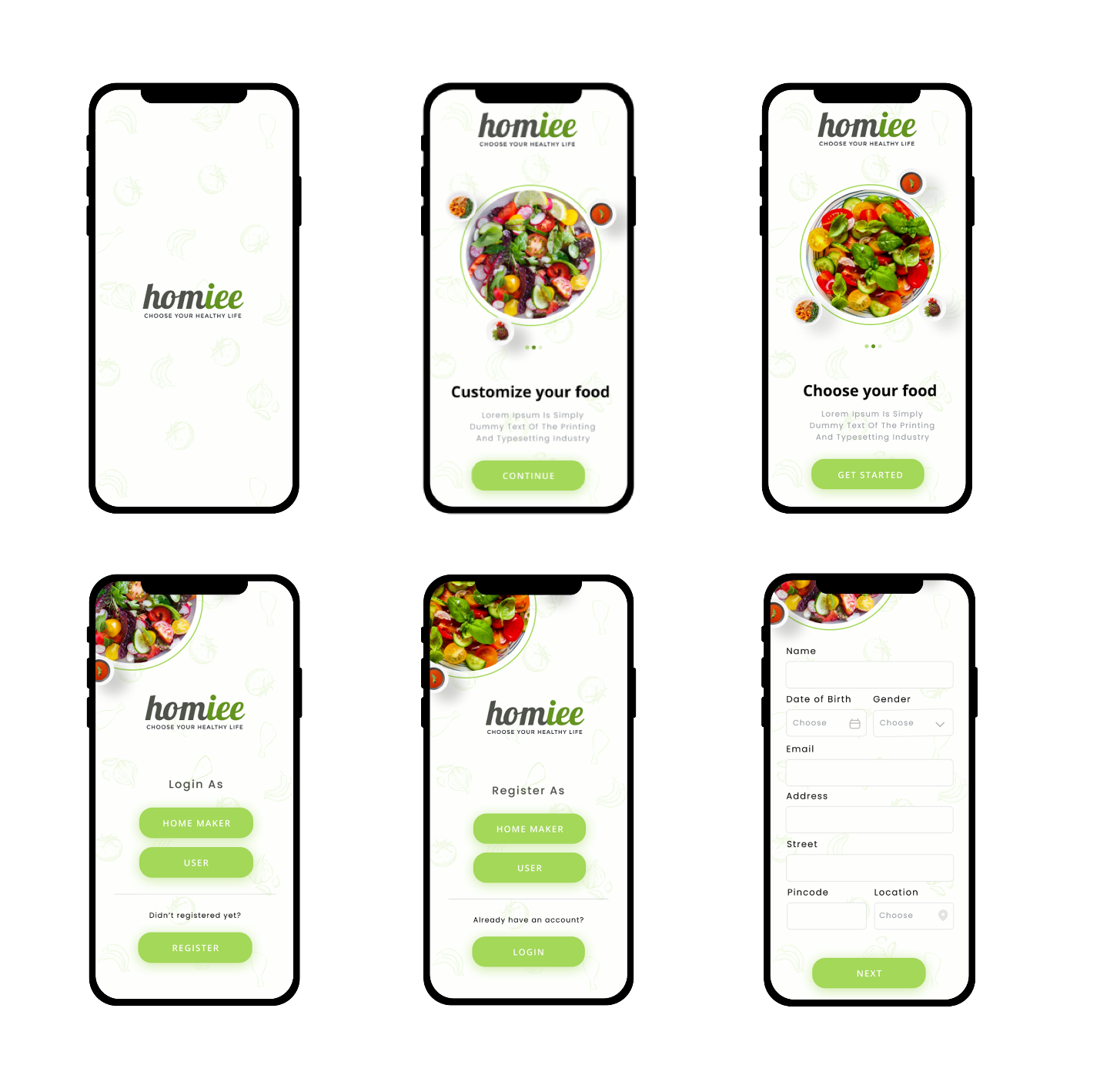
1. Login Page: The login can be performed for two types of users “Homemaker” and “User” , which typically consist of input fields for the user to enter their login credentials, such as email and password. The form may also include buttons for actions such as "Login" and "Forgot Password".
2. Registration page: In the GUI design for the homely food delivery system is a screen that allows new users to create an account and register themselves in the application. It typically includes input fields for users to enter their personal information, such as name, email ,address, password, and other relevant details.

* Home maker Registration: Homemaker should upload license and aadhar for personal verification
* User Registration: Registering with providing personal information.

1. Home page:

* Home maker’s home page: Home makers can manage their menus and add the meals they are going to prepare according to dates.
* User’s home page: users can search for menus based on dates and location, view available options, and place orders.

1. Orders Page :Home maker receives notifications of orders and can view details of orders.
2. Profile Page: The profile page of the home maker in the GUI design for the homely food delivery system is a dedicated page where home makers can manage their profile information, view reviews and ratings of their menus, add bank account details for receiving payments, switch to a user account for ordering meals, manage settings, and view their orders.



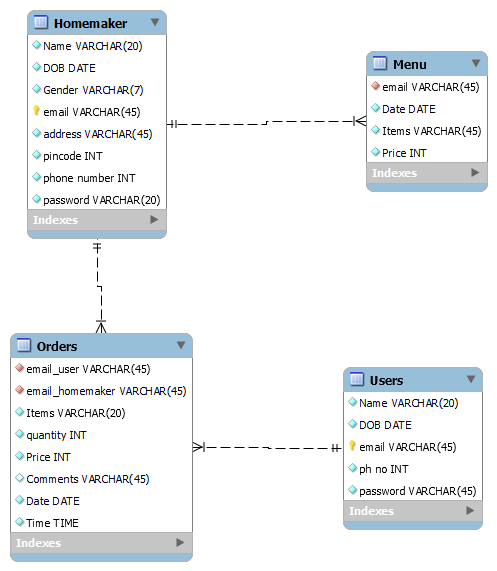
**5.API Design**

* Authentication API: Handling user authentication and authorization, such as registration, login, and session management. This API will ensure that only authorized users can access the application and perform actions such as placing orders or managing user profiles.
* Meal Listings API: Managing the meal listings, including adding new meals, updating meal information, and retrieving meal details. This API will be responsible for interacting with the meal listings database.
* Order Management API: Handling order placement, order tracking, and order history. This API will manage the order information and update the status of orders in real-time.
* Payment Processing API: Integrating with third-party payment gateways, such as Stripe or PayPal, for processing online payments securely. This API will handle payment transactions and ensure that user payment information is protected.
* Database: Using Firebase Firestore or Firebase Realtime Database for storing and retrieving data related to meals, orders, and user profiles. The database will be designed to support efficient data retrieval and update operations and ensure data consistency and integrity.

**6.Database Design**

The database design section for our project include the structure and organization of the system's database. It will define the data entities, their attributes, relationships, and constraints, and provide a clear understanding of how the database will be used to store and retrieve information**.**

It includes the following tables:

****

**7. Technology Stack**

* Front-end Development:

Flutter: A UI toolkit for building natively compiled applications for mobile, web, and desktop from a single codebase using dart language. It provides a rich set of pre-built UI widgets and allows for fast development and customization of the user interface.

* Back-end Development:

Firebase: A Backend-as-a-Service (BaaS) platform by Google that offers features such as authentication, real-time database, cloud storage, and cloud functions. It can be used for handling user authentication, storing menu data, order processing, and other server-side functionalities.

* Database:

Firebase Realtime Database or Firebase Firestore

* Authentication:

Firebase Authentication: A Firebase service that provides built-in authentication methods such as email/password, social media logins (e.g., Google, Facebook), and more. It can be used for managing user authentication and authorization in your homely food delivery system.

**8. Conclusion**

The Software Design Document (SDD) provides a detailed overview of the architecture, design, and functionality of the homely food delivery system. It outlines the system's objectives, user interfaces, application architecture, technology stack, and database design. The SDD serves as a blueprint for the development team, guiding them in implementing the system according to the defined specifications. It promotes separation of concerns, scalability, and maintainability through a layered architecture and utilizes technologies such as Flutter and Firebase. The SDD is crucial for successful implementation and future updates of the system.