	Zewail City for Science and Technology  2025-2026
---	--

<b>Course:</b>	Mobile Application Development
<b>Code:</b>	SWAPD 402
<b>type</b>	Project

<b>Due Date:</b>	Nov 2025
<b>Project name:</b>	Flavor Feiasta

Under The Supervision of

Dr. Yousry Abdul Azeem

Team:

<b><u>Name:</u></b>	<b><u>ID:</u></b>
Nour Eldin Mohamed	202201310
Hala Mohamed Salah	202100987
wafaa abdulrazek	202202056
Rana Hazem Almahdy	202201938

# Table of Contents

<b>1. App Purpose &amp; Target Users &amp; Functionality:</b>	<b>3</b>
1.1 Purpose:	3
1.2 Target Users:	3
1.3 Core Functionality:	3
<b>2. Features Implemented:</b>	<b>4</b>
2.1 User Authentication System	4
2.2 Recipe Browsing and Search:	5
2.3 Add Custom Recipes:	5
2.4 Cooking Timer:	6
2.5 Checklist Management:	6
2.6 User Profile:	6
2.7 Onboarding Experience:	7
2.8 Theme Support:	7
2.9 Splash Screen:	7
<b>3. Tools, Technologies, and Frameworks:</b>	<b>8</b>
3.1 Core Framework:	8
3.2 Backend Services (Firebase):	8
3.3 State Management:	9
3.4 Local Storage:	9
3.5 Media Handling:	9
3.6 Networking:	9
3.7 Utilities:	9
3.8 UI/UX Enhancements:	10
3.9 Development Tools:	10
3.10 Platform Support:	10
<b>4. Project Architecture:</b>	<b>11</b>
4.1 Design Patterns	11
<b>5. Conclusion:</b>	<b>11</b>
<b>6. Sample Output (Final):</b>	<b>12</b>
<b>7. Appendices:</b>	<b>13</b>

# 1. App Purpose & Target Users & Functionality:

## 1.1 Purpose:

The Food Recipes App is a comprehensive mobile application designed to simplify cooking and meal planning for users of all skill levels. The application serves as a digital cookbook that allows users to discover new recipes, save their favorite meals, create shopping lists, and manage their cooking activities efficiently. The app bridges the gap between recipe discovery and practical cooking by providing interactive features such as cooking timers and ingredient checklists.

In today's fast-paced world, people often struggle to plan meals, organize ingredients, and follow recipes effectively. The Food Recipes App addresses these challenges by providing an all-in-one solution that combines recipe browsing, personal recipe management, and cooking assistance tools in a single, user-friendly interface.

## 1.2 Target Users:

The Food Recipes App is designed to cater to a diverse range of users:

### **Home Cooks:**

Individuals who cook regularly at home and are looking for new recipe ideas to expand their culinary repertoire. These users benefit from the extensive recipe database and the ability to save and organize their favorite recipes.

### **Beginners in Cooking:**

People who are new to cooking and need guidance. The app provides detailed recipe information including preparation time, cooking time, difficulty level, and serving sizes, making it easier for beginners to choose appropriate recipes.

### **Meal Planners:**

Users who like to plan their meals in advance. The checklist feature allows them to create shopping lists based on recipe ingredients, ensuring they have everything needed before starting to cook.

### **Food Enthusiasts:**

Culinary enthusiasts who enjoy experimenting with different cuisines. The app offers recipes from various categories including Italian, Indian, Salads, Fast Food, and more.

### **Busy Professionals:**

Working individuals who need quick access to recipes with clear timing information. The built-in cooking timer helps them manage cooking time efficiently.

## 1.3 Core Functionality:

The Food Recipes App provides the following core functionalities:

### **Recipe Discovery:**

- Browse through a curated collection of recipes from various cuisines

- Search recipes by name or keywords
- Filter recipes by meal type and category
- View detailed recipe information including ingredients, preparation steps, cooking time, and difficulty level

#### **Personal Recipe Management:**

- Create and add custom recipes with images
- Store recipes securely in the cloud using Firebase
- Access personal recipe collection from any device
- Edit and delete user-created recipes

#### **User Authentication:**

- Secure user registration and login system
- Password recovery functionality
- Personalized user profiles with profile picture support
- Session management for seamless app experience

#### **Cooking Assistance:**

- Built-in cooking timer with alarm functionality
- Audio notifications when cooking time is complete
- Pause and resume timer capabilities

#### **Shopping & Planning:**

- Create multiple shopping checklists
- Add, edit, and delete checklist items
- Mark items as completed
- Organize ingredients by categories

#### **Personalization:**

- Dark mode and light mode theme support
- Profile customization with photo upload
- Persistent user preferences

## **2. Features Implemented:**

### **2.1 User Authentication System**

#### **Sign Up:**

- New users can create an account using their email address and password
- Full name collection during registration
- User data is stored securely in Firebase Firestore
- Input validation for email format and password strength

#### **Login:**

- Existing users can sign in with email and password
- Remember me functionality using SharedPreferences
- Error handling with user-friendly messages
- Secure authentication through Firebase Auth

**Password Recovery:**

- Users can reset their password via email
- Firebase handles secure password reset links

**Profile Management:**

- View and edit personal profile information
- Upload and change profile picture
- Local storage of profile images for faster loading
- Sign out functionality with confirmation dialog

**2.2 Recipe Browsing and Search:****Recipe Listing:**

- Display recipes in an attractive card-based grid layout
- Show recipe image, title, category, preparation time, and rating
- Support for both local sample recipes and API-fetched recipes
- Real-time updates when new recipes are added

**Search Functionality:**

- Real-time search as user types
- Search by recipe name
- Clear search button for convenience
- Instant filtering of results

**Category Filtering:**

- Horizontal scrollable category chips
- Filter recipes by meal type (Italian, Indian, Salad, Fast Food, etc.)
- "All" category to show all recipes
- Dynamic category list based on available recipes

**Recipe Details:**

- Full recipe information display
- Preparation time and cooking time
- Serving size information
- Difficulty level indicator
- Recipe description and instructions

**2.3 Add Custom Recipes:****Recipe Form:**

- Comprehensive form for adding new recipes
- Fields include: name, category, difficulty, prep time, cook time, servings, and description
- Form validation to ensure required fields are filled
- User-friendly input fields with appropriate keyboard types

**Image Handling:**

- Pick images from device gallery
- Capture new images using device camera
- Image compression for optimal storage
- Local image storage in app documents directory

- Permission handling for camera and storage access

#### **Cloud Storage:**

- Recipes are saved to Firebase Realtime Database
- Real-time synchronization across devices
- User-specific recipe collections
- Automatic timestamp tracking for creation and updates

## **2.4 Cooking Timer:**

#### **Timer Input:**

- Input minutes and seconds using dedicated text fields
- Input validation for numeric values
- Clean, intuitive interface design

#### **Timer Controls:**

- Start, pause, and reset functionality
- Visual countdown display in MM:SS format
- Progress indicator during countdown
- State management for running, paused, and finished states

#### **Alarm System:**

- Audio alarm when timer completes
- Uses external audio source for alarm sound
- Volume control integration
- Visual notification with color change when timer ends

## **2.5 Checklist Management:**

#### **Multiple Checklists:**

- Create unlimited shopping checklists
- Custom naming for each checklist
- Swipeable page view for navigating between checklists
- Delete checklists with confirmation

#### **Checklist Items:**

- Add items with name, quantity, and unit
- Toggle items as checked/unchecked
- Delete individual items
- Visual distinction for completed items

#### **Empty State Handling:**

- Attractive empty state display when no checklists exist
- Clear call-to-action to create first checklist
- Helpful guidance for new users

## **2.6 User Profile:**

#### **Profile Display:**

- User's full name and email display

- Profile picture with fallback avatar
- Account creation date
- Loading states during data fetch

**Profile Picture:**

- Select image from gallery
- Local storage for fast loading
- Circular avatar display
- Upload progress indicator

**Settings:**

- Theme toggle (Light/Dark mode)
- Notification preferences
- Sign out option
- Account management

## **2.7 Onboarding Experience:**

**Interactive Onboarding:**

- Three-screen onboarding flow for new users
- Animated illustrations using Lottie animations
- Page indicator dots showing progress
- Skip option for returning users

**Onboarding Content:**

- Screen 1: Recipe selection introduction
- Screen 2: Shopping list feature explanation
- Screen 3: Cooking and enjoyment encouragement

## **2.8 Theme Support:**

**Light Mode:**

- Clean, bright interface
- Standard Material Design colors
- Easy readability in well-lit environments

**Dark Mode:**

- Eye-friendly dark interface
- Reduced eye strain in low-light conditions
- Automatic theme persistence using Provider

## **2.9 Splash Screen:**

**App Launch:**

- Branded splash screen on app launch
- Initialization of required services
- Loading of user preferences
- Smooth transition to main app content

### 3. Tools, Technologies, and Frameworks:

#### 3.1 Core Framework:

##### Flutter:

- Version: SDK 3.9.2
- Type: Cross-platform UI framework
- Developer: Google
- Purpose: Flutter serves as the primary framework for building the Food Recipes App. It enables the development of natively compiled applications for mobile (iOS and Android), web, and desktop from a single codebase. Flutter's widget-based architecture allows for rapid UI development with highly customizable components.

##### Dart:

- Type: Programming Language
- Developer: Google
- Purpose: Dart is the programming language used with Flutter. It provides features like strong typing, null safety, and async/await patterns that make it ideal for building robust mobile applications.

#### 3.2 Backend Services (Firebase):

##### Firebase Core:

- Package: `firebase_core`:
- Purpose: Provides the essential Firebase services initialization and configuration required for all other Firebase services to function.

##### Firebase Authentication:

- Package: `firebase_auth`:
- Purpose: Handles user authentication including sign up, sign in, sign out, and password reset functionality. Provides secure token-based authentication with support for various authentication providers.

##### Cloud Firestore:

- Package: `cloud_firestore`:
- Purpose: NoSQL cloud database used for storing user profile information. Provides real-time data synchronization, offline support, and scalable data storage.

##### Firebase Realtime Database:

- Package: `firebase_database`:
- Purpose: Real-time NoSQL database used for storing recipe data. Enables real-time data synchronization across all connected clients, making recipe updates instantly visible to all users.

##### Firebase Storage:

- Package: `firebase_storage`:
- Purpose: Cloud storage solution for storing user-uploaded images including recipe photos and profile pictures. Provides secure file uploads with progress monitoring.



### 3.3 State Management:

#### Provider:

- Package: provider:
- Purpose: State management solution used throughout the application. Manages theme state (light/dark mode) and provides reactive updates to the UI when state changes occur. Provider offers a simple yet powerful way to manage app-wide state without boilerplate code.

### 3.4 Local Storage:

#### SharedPreferences:

- Package: shared\_preferences:
- Purpose: Persistent key-value storage for storing user preferences, authentication state, and locally cached recipe data. Used for remembering user settings between app sessions.

#### Path Provider:

- Package: path\_provider:
- Purpose: Provides access to commonly used locations on the device file system. Used for storing recipe images and profile pictures locally to reduce network usage and improve performance.

### 3.5 Media Handling:

#### Image Picker:

- Package: image\_picker:
- Purpose: Enables users to select images from the device gallery or capture new photos using the camera. Used for adding recipe images and profile pictures.

#### Permission Handler:

- Package: permission\_handler:
- Purpose: Manages runtime permissions for camera and storage access. Ensures proper permission requests and handles denied permissions gracefully.

#### Audioplayers:

- Package: audioplayers:
- Purpose: Audio playback library used for the cooking timer alarm functionality. Plays notification sounds when the cooking timer completes.

### 3.6 Networking:

#### HTTP:

- Package: http:
- Purpose: HTTP client for making API requests to external recipe services. Enables fetching recipes from third-party APIs to supplement the local recipe database.

### 3.7 Utilities:

#### UUID

- Package: uuid:

- Purpose: Generates unique identifiers for recipes, checklists, and checklist items. Ensures each item has a unique ID for proper data management.

### 3.8 UI/UX Enhancements:

#### Lottie:

- Package: lottie:
- Purpose: Renders Adobe After Effects animations exported as JSON files. Used in the onboarding screens to provide engaging, animated illustrations that enhance the user experience.

#### Cupertino Icons:

- Package: cupertino\_icons:
- Purpose: Provides iOS-style icons for maintaining platform consistency and offering a wide range of icon options for the user interface.

### 3.9 Development Tools:

#### Flutter Lints:

- Package: flutter\_lints:
- Purpose: Provides a set of recommended lint rules that encourage good coding practices. Helps maintain code quality and consistency across the project.

#### Icons Launcher:

- Package: icons\_launcher:
- Purpose: Simplifies the process of generating app icons for all required sizes and platforms. Ensures consistent branding across iOS and Android.

#### Flutter Test:

- Package: flutter\_test (SDK)
- Purpose: Testing framework for writing and running unit tests, widget tests, and integration tests. Ensures app reliability and catches bugs early in development.

### 3.10 Platform Support:

The Food Recipes App is built to support multiple platforms:

Platform	Support Status	Configuration Files
Android	Full Support	android/app/build.gradle.kts, google-services.json
iOS	Full Support	ios/Runner/, Podfile
Web	Support	web/index.html, manifest.json
Windows	Support	CMakeLists.txt
macOS	Support	macos/Podfile, Runner/

Linux	Support	CMakeLists.txt
-------	---------	----------------

## 4. Project Architecture:

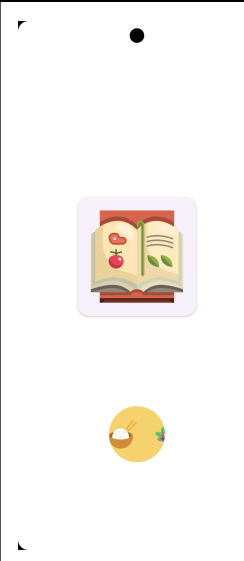

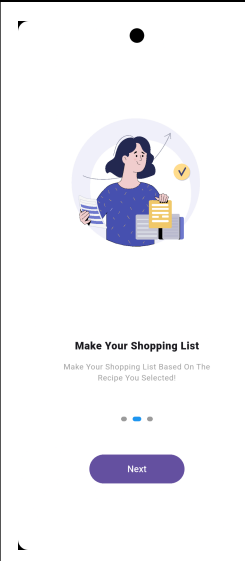
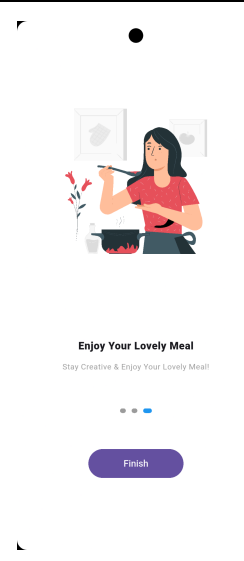
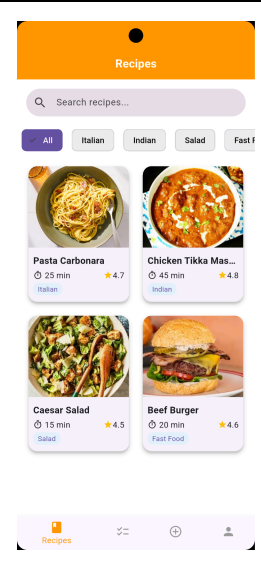
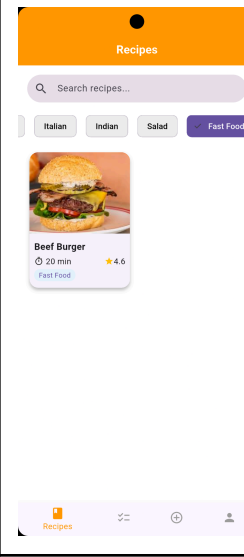
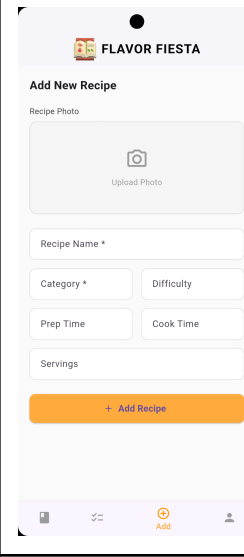
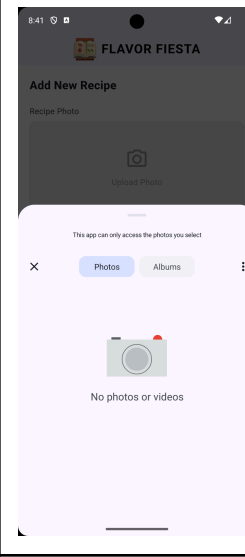
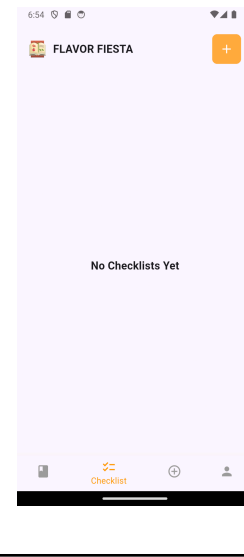
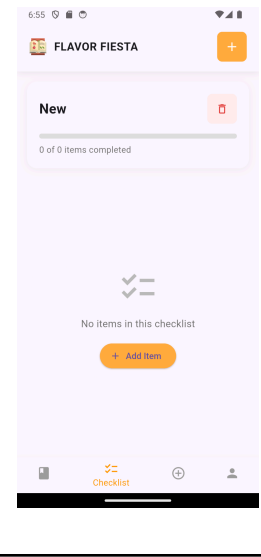
### 4.1 Design Patterns

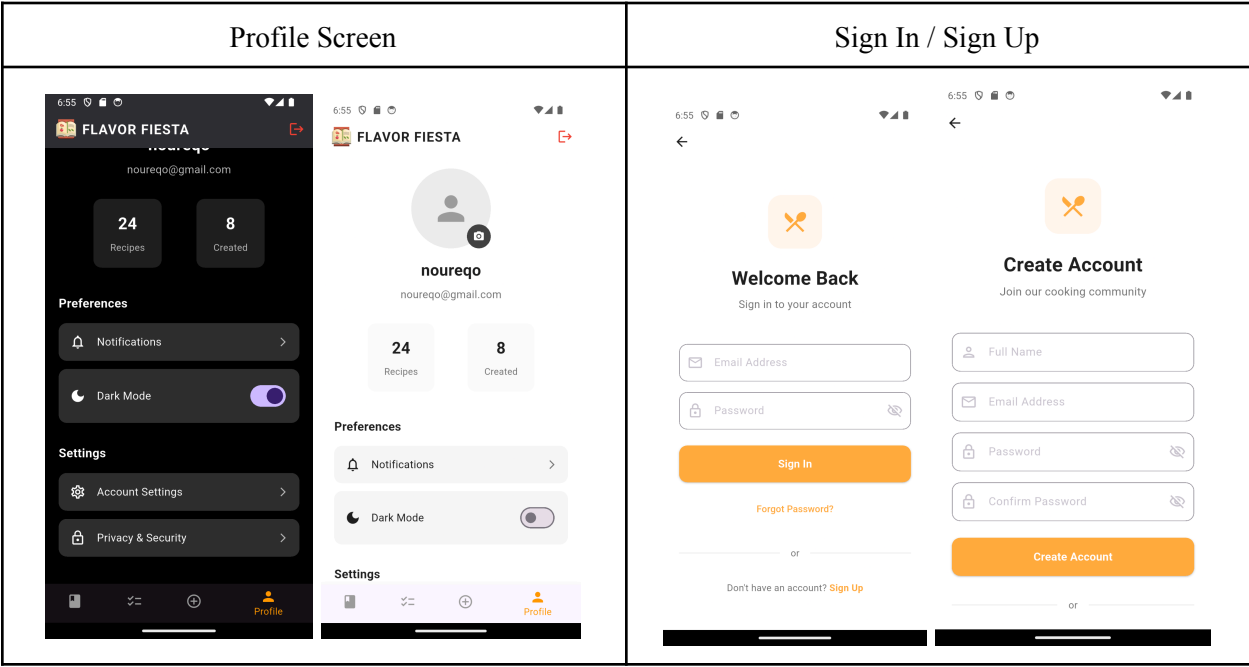
- Provider Pattern: Used for state management across the application
- Service Layer Pattern: Separates business logic from UI components
- Repository Pattern: Data access through dedicated service classes
- Model-View Pattern: Clear separation between data models and UI screens

## 5. Conclusion:

The Food Recipes App is a comprehensive, feature-rich mobile application built with modern technologies and best practices. By leveraging Flutter's cross-platform capabilities and Firebase's robust backend services, the app delivers a seamless cooking experience to users across multiple platforms. The combination of recipe discovery, personal recipe management, cooking assistance tools, and shopping list features makes this app a valuable companion for anyone interested in cooking. The implementation of proper authentication, cloud storage, and state management ensures a secure and reliable user experience.

6.Sample Output (Final):

Loading Screen	Onboarding Screen	Onboarding Screen	Onboarding Screen	Recipe Screen
				
Recipe Filter Screen	Add Recipe Screen	Add Recipe Screen	List Screen	Add List Screen
				



7.Appendices:

WireFrame: [Figma Link](#)  
GitHub: [Food Recipes App](#)