**COIMBATORE INSTITUTE OF TECHNOLOGY**

(Government-Aided Autonomous Institution Affiliated To Anna University, Chennai)

COIMBATORE – 14.

**MOBILE APPLICATION DEVELOPMENT LAB-19MAM66**

**SUBMITTED TO:**

**DR. D. KAVITHA**

**MS. P. HEMASHREE**

**DEPARTMENT OF COMPUTING**



**SUBMITTED BY:**

**NAME :** SARVESH PV

**REGISTER NO**: 71762234042

**DEPARTMENT:** 3rd YEAR MSc-AIML

**MEAL PLANNER**

**Abstract**

This project presents a dynamic Meal Planner application designed to provide users with an efficient, curated, and engaging experience for discovering meal recipes tailored to their preferences. Engineered with Flutter for cross-platform compatibility and utilizing SQLite for persistent local storage, the application seamlessly integrates modern functionalities such as API-based recipe retrieval, location-based services, secure local login management, real-time content updates via an RSS feed, and intelligent meal recommendations. The application not only supports user convenience through advanced features like categorization, favorites management, and personalized suggestions but also enriches user interaction by enabling email notifications and integrating multimedia recipe tutorials. The objective is to deliver an intuitive and sophisticated platform that elevates meal planning into a delightful and intelligent daily routine.

**Introduction**

Meal planning is a critical yet often overlooked component of maintaining a balanced, healthy lifestyle. In the era of digital convenience, providing users with an intelligent, mobile-first meal planning solution significantly enhances their culinary journey. This Meal Planner app bridges the gap by offering an aesthetic, user-centric interface that retrieves high-quality recipe data from external APIs, enriched with features such as search capabilities, favorites tracking, personalized recommendations, and detailed tutorials.

Architecturally, the application leverages Flutter's robust UI toolkit combined with SQLite for local storage persistence and Riverpod for state management, ensuring a performant, reactive, and secure mobile experience. This synergy ensures both data security and exceptional responsiveness across devices.

**Modules and Descriptions**

**1. Login & Authentication Module**

* **Description:** Facilitates user authentication with secure credential storage on-device.
* **Features:**
  + Local authentication via SQLite.
  + Input validation and exception handling.
  + Persistent session management within app lifecycle.
  + Triggering email notifications post-login success.
  + Redirection to the primary dashboard upon successful authentication.

**2. Recipe Retrieval and Display Module**

* **Description:** Core functionality allowing users to fetch and explore meal recipes.
* **Features:**
  + Categorized recipe browsing (e.g., Breakfast, Lunch, Desserts, etc.).
  + Real-time API integration to fetch up-to-date meal data.
  + Recipe of the Week highlight section.
  + Intelligent recommendation engine based on popular trends and user interactions.

**3. Favorites Management Module**

* **Description:** Empowers users to save, categorize, and manage their preferred recipes.
* **Features:**
  + Persistent storage of favorites within SQLite.
  + Category-wise favorites organization.
  + Add/remove favorite recipes seamlessly.
  + Offline access to saved recipes.

**4. Search Functionality**

* **Description:** Provides a swift, responsive search feature across all available recipes.
* **Features:**
  + Keyword-based dynamic search.
  + Search suggestions based on popular queries.
  + Instant retrieval and display of search results.

**5. Detailed Recipe View**

* **Description:** Displays in-depth information for each recipe.
* **Features:**
  + Ingredient listing and preparation instructions.
  + Nutritional information (where available).
  + Direct link to corresponding YouTube tutorial videos.
  + Interactive UI elements for enhanced readability.

**6. RSS Feed Integration**

* **Description:** Supplies users with the latest culinary news, trends, and new recipes.
* **Features:**
  + Periodic feed updates using background fetch (Dart Isolates for multithreading).
  + Instantaneous display of latest entries without disrupting main UI.
  + Real-time refresh to keep users abreast of new content.

**7. Location Retrieval**

* **Description:** Captures and displays the user's current geographical coordinates.
* **Features:**
  + Utilizes device GPS to fetch latitude and longitude.
  + Displays location data alongside personalized recommendations (future extensibility).

**9. Logout Module**

* **Description:** Ensures secure user session termination.
* **Features:**
  + Cleans user session data from SQLite.
  + Redirection to the login screen.
  + Prevention of unauthorized navigation post-logout.

**Methods Used**

**Frontend (Flutter)**

* Dart programming language.
* Highly responsive UI crafted with stateless and stateful widgets.
* Navigator 2.0 for seamless screen transitions.
* Riverpod for robust and scalable state management.
* **Packages utilized:**
  + http for RESTful API interactions.
  + sqflite for local database operations.
  + geolocator for location services.
  + share\_plus for sharing recipes.
  + flutter\_email\_sender for email notifications.
  + webfeed for RSS parsing.
  + flutter\_riverpod for efficient app-wide state control.

**Backend / Data Storage**

* External recipe API integration for real-time recipe data.
* SQLite database to persist login credentials, favorites, and user preferences.
* RESTful API standards for secure and structured data fetching.

**Multithreading (Dart Isolates)**

* Employed to offload RSS feed parsing and updates.
* Guarantees uninterrupted UI responsiveness during heavy background operations.
* Concurrent retrieval and rendering of new feed entries.

**Email Notification**

* Post-login email confirmation via flutter\_email\_sender.
* Configurable templates for user engagement and security acknowledgment.

**Location Services**

* Fetches device coordinates using geolocator.
* Future scope for location-based recipe recommendations or event-specific meals.

**Conclusion**

The Meal Planner App exemplifies the successful integration of cutting-edge mobile development practices to create a comprehensive, aesthetically pleasing, and highly functional meal discovery platform. By synergizing Flutter’s modern UI capabilities, Riverpod’s scalable state management, and SQLite’s local persistence, the application offers users an intelligent and deeply personalized meal planning experience. Features such as recipe recommendations, multimedia tutorial integration, real-time RSS updates, and location awareness collectively enhance user engagement, usability, and retention.

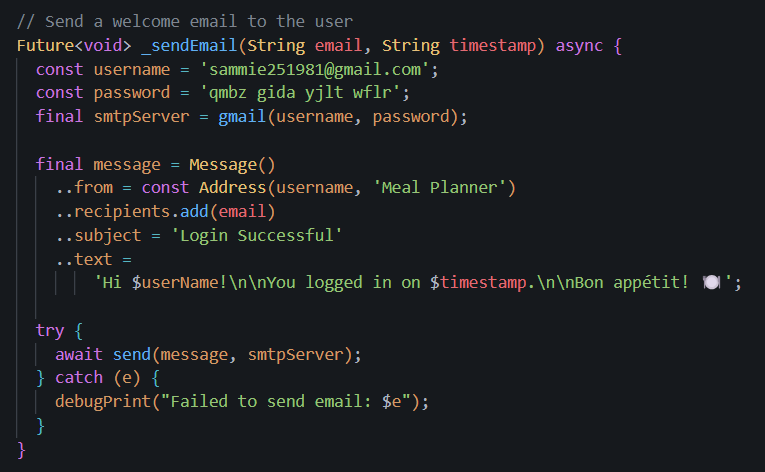
Anticipated future upgrades include calendar-based meal scheduling, personalized nutrition tracking, AI-based smart meal suggestions, and voice command integrations, further cementing Meal Planner's position as a next-generation lifestyle application.

**Code Snippets**

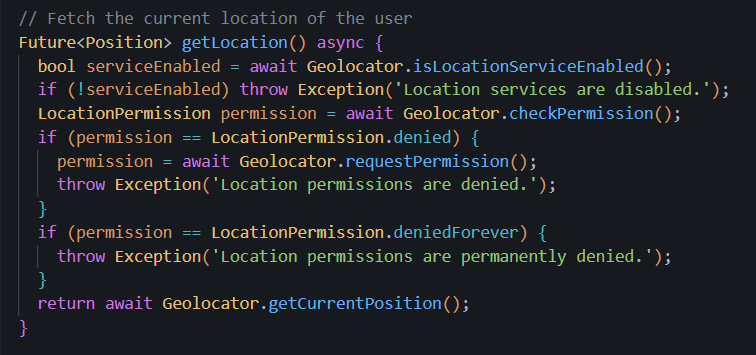
**1.Database**

****

**2. E-Mail**

****

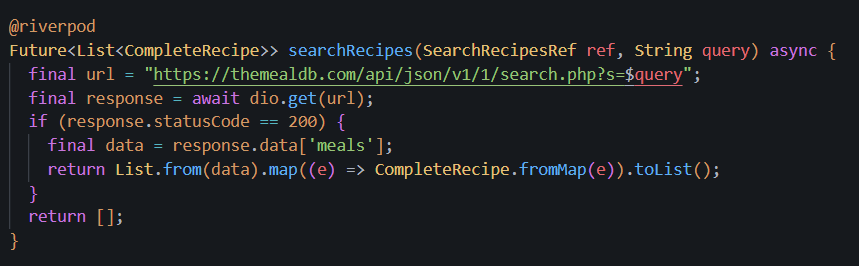
**3. Location Service**

****

**4. RSS Feed**

****

**5. Search**

****

**Output :**

