4.0 Methodology, Results and Discussion

4.1 Methodology

The proponents used the Rapid Application Development (RAD) methodology in developing the LifeSync mobile application. The RAD model emphasizes iterative development and quick user feedback, making it suitable for applications with multiple integrated features such as logins, diary management, weather updates, movie reviews, plant care tracking, FAQ page, and also a task list organization.

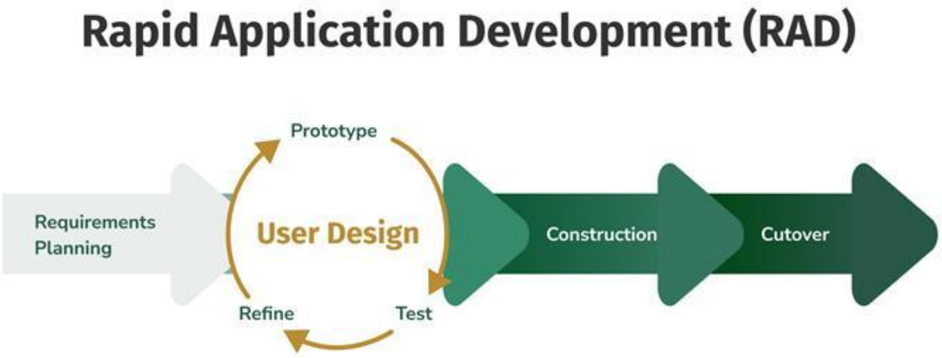


Figure 1. Rapid Application Development (RAD)

4.1.1 Requirements Planning

In this phase the proponents collected all the needed requirements in developing LifeSync mobile application. Research, data collection and also conducted interviews to identified the needs of creating an app that has all the features that is preferred by the user. Also, each member initially developed a personal mobile app with the shared modules which is the login, diary, weather and a unique special feature. These individual apps served as the foundation for merging into one application.

4.1.2 User Design

After all the necessary requirements are gathered, the proponents proceeded to design the interface of LifeSync mobile application.

4.1.2.1 Prototype

During this phase the proponents used Figma to visualize the possible layout and structure of LifeSync. All the needed icons and graphic elements were collected using Canva and Google.

4.1.2.2 Test

The proponents tested the design after finishing the prototype in order to evaluate its usability and user experience. Selected individuals were requested to test out the prototype in order to evaluate its clarity, usability, and accessibility.

A. Alpha Testing

The proponents performed alpha testing to identify and fix early issues such as unresponsive buttons, navigation errors, and minor design inconsistencies. Each module was tested individually to ensure that all components worked as intended.

B. Beta Testing

After resolving issues found in alpha testing, beta testing was conducted by a new set of users. This stage aimed to detect any remaining bugs or performance issues that were not previously visible.

4.1.2.3 Refine

The user interface and performance of the LifeSync mobile application were improved through refinement. To ensure an improved performance, changes were made to the navigation system, icon consistency, and general layout.

4.1.2.4 Cutover

The cutover phase marked the transition from development to actual implementation. During this stage, the final version of LifeSync was prepared for release and deployment to target users. All tested modules were integrated, and final configurations were made to ensure the system operated efficiently. Backup measures and documentation were also completed to support maintenance and future updates.

4.3 Requirements Specification

4.3.1 Operational Feasibility

- Fishbone Diagram

LifeSync Fishbone Diagram

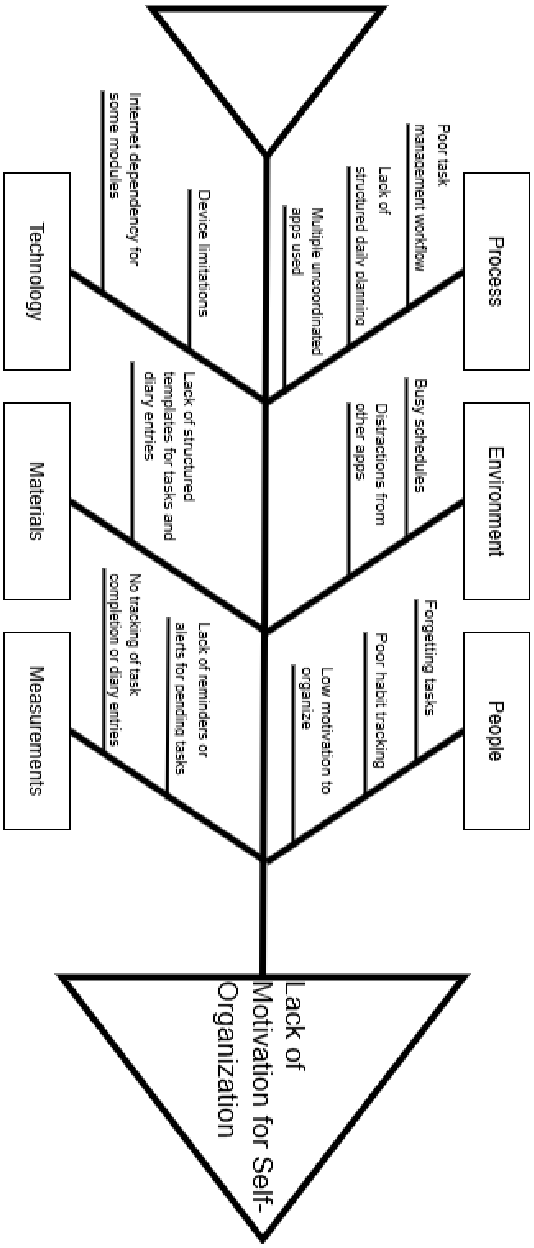

Technology
Internet dependency for some modules
Device limitations
Process
Poor task management workflow
Lack of structured daily planning
Multiple uncoordinated apps used
Materials
Lack of structured templates for tasks and diary entries
Environment
Busy schedules
Distractions from other apps
Measurements
No tracking of task completion or diary entries
Lack of reminders or alerts for pending tasks
People
Forgetting tasks
Poor habit tracking
Low motivation to organize
Lack of Motivation for Self-Organization

Figure 3. LifeSync Fishbone Diagram

- Functional Decomposition Diagram

LifeSync Functional Decomposition Diagram

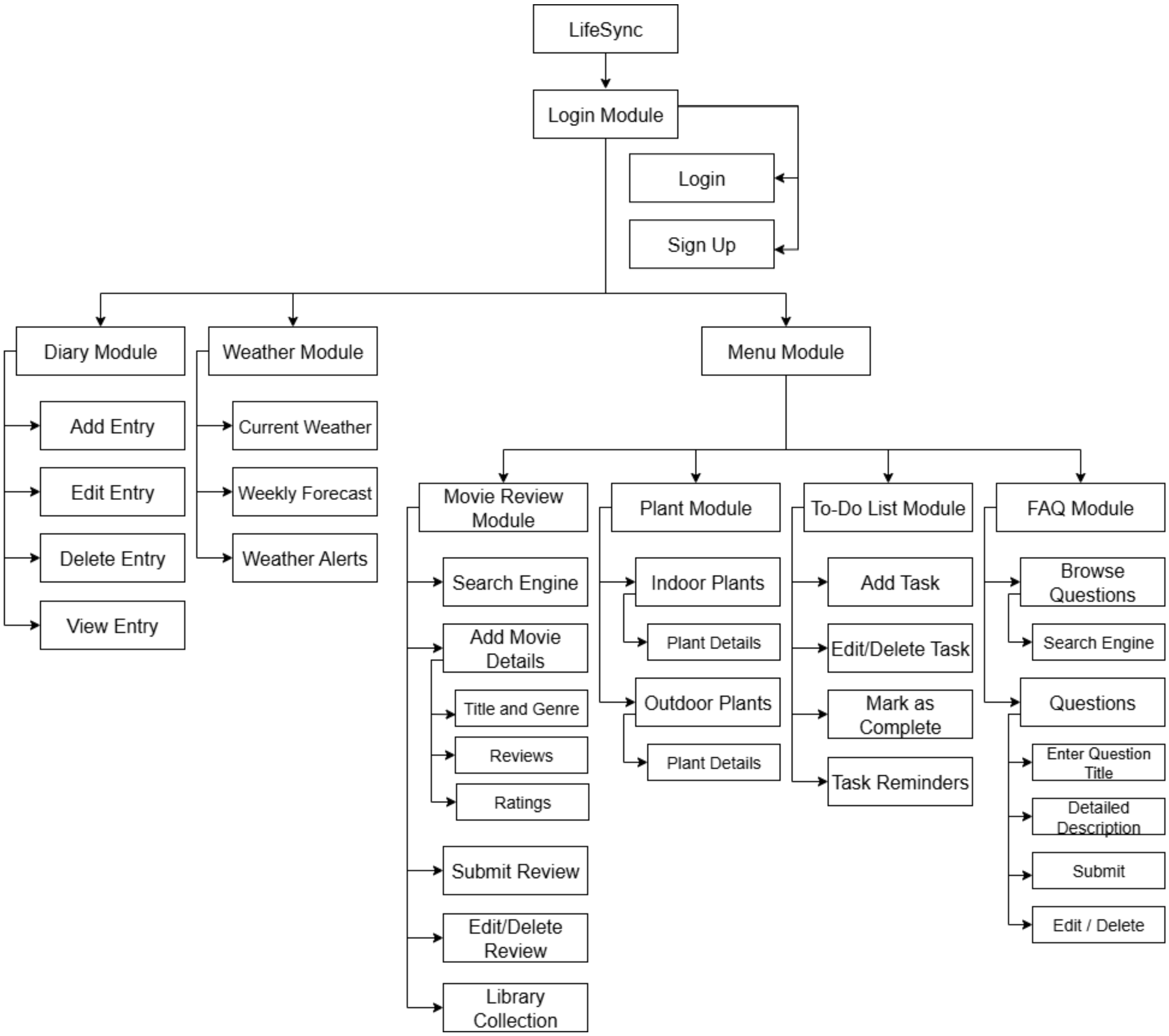


Figure 4. LifeSync Functional Decomposition Diagram

4.3.2 Technical Feasibility

LifeSync is compatible with both online and mobile platforms, it is technically possible and accessible on a variety of devices. Flutter and Dart were used in its development to enable complete cross-platform performance on iOS and Android (8.0 and up) devices. Firebase is used in its backend to provide secure authentication, cloud storage, and real-time database management. Basic hardware requirements for the system include a PC or smartphone with at least 2GB of RAM and consistent internet access. Because of their efficiency, scalability, and dependability, the selected technologies are extremely pertinent to the project's goals and allow LifeSync to provide a smooth, synchronized, and intuitive experience for managing personal data and lifestyle.

4.3.3 Schedule Feasibility

- Gantt Chart

Table 1: Gantt Chart for the month of July 2025

KEY ACTIVITIES	WEEK 1	WEEK 2	WEEK 3	WEEK 4
Brainstorming with the Proponents	■			
Identify project objectives, scope, and purpose	■	■		
Gather requirements through research and interviews		■	■	
Draft proposal outline and define features and modules				■

Table 2: Gantt Chart for August 2025

KEY ACTIVITIES	WEEK 1	WEEK 2	WEEK 3	WEEK 4
Designing the User Interface and Layout	■	■		
Create user flow diagrams and wireframes		■	■	
Design system data flow and database schema			■	■
Prepare prototype for initial testing				■

Table 3: Gantt Chart for September 2025

KEY ACTIVITIES	WEEK 1	WEEK 2	WEEK 3	WEEK 4
Conduct alpha testing for prototype	■	■		
Collect user feedback and identify issues	■	■		
Perform beta testing with selected users		■	■	
Refine user interface and fix performance issues			■	■

Table 4: Gantt Chart for October 2025

KEY ACTIVITIES	WEEK 1	WEEK 2	WEEK 3	WEEK 4
Develop Login and Authentication module	■	■		
Develop Diary, Weather, and Menu modules		■	■	
Develop Movie Review, Plant, To-Do List, and Q&A modules		■	■	■
Integrate all modules and test system connectivity			■	■

Table 5: Gantt Chart for November 2025

KEY ACTIVITIES	WEEK 1	WEEK 2	WEEK 3	WEEK 4
Deploy final version to users		■	■	
Collect user feedback after deployment			■	■
Perform maintenance and prepare system documentation				■

4.3.4 Economic Feasibility

LifeSync's development and maintenance expenses are affordable in regard to the advantages it offers, the recommended system is financially feasible. Software development platforms like Flutter and Firebase, which provide cheap or free plans ideal for startup companies, as well as low hosting and maintenance fees, are the primary costs. Because the idea works well on common computers and smartphones, it doesn't require expensive equipment, which eases financial burden.

4.3.5 Requirements Modeling

- Object Modeling

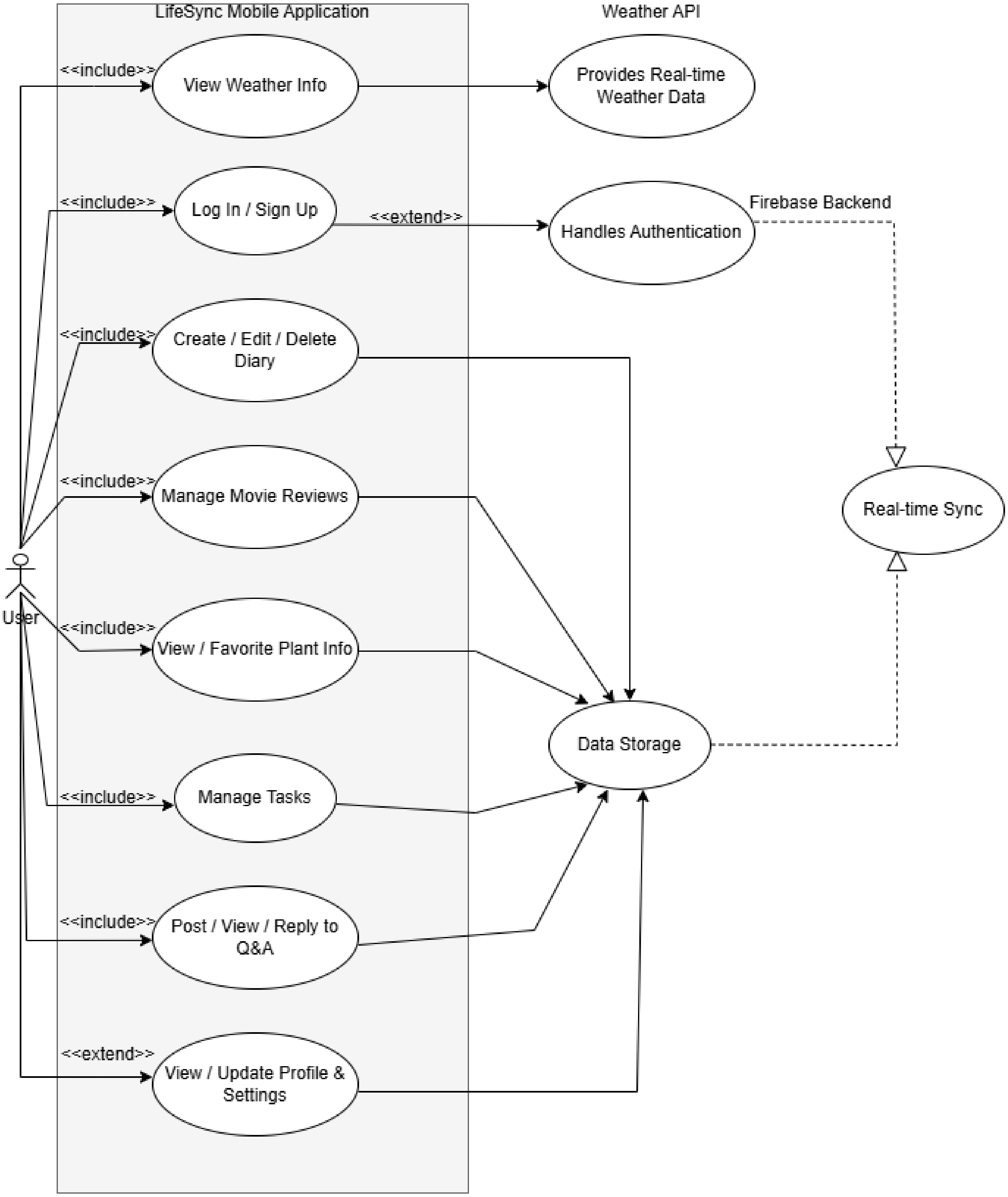


Figure 5. LifeSync's Use Case Diagram

- Sequence Diagram

Figure 7. LifeSync’s Sequence Diagram

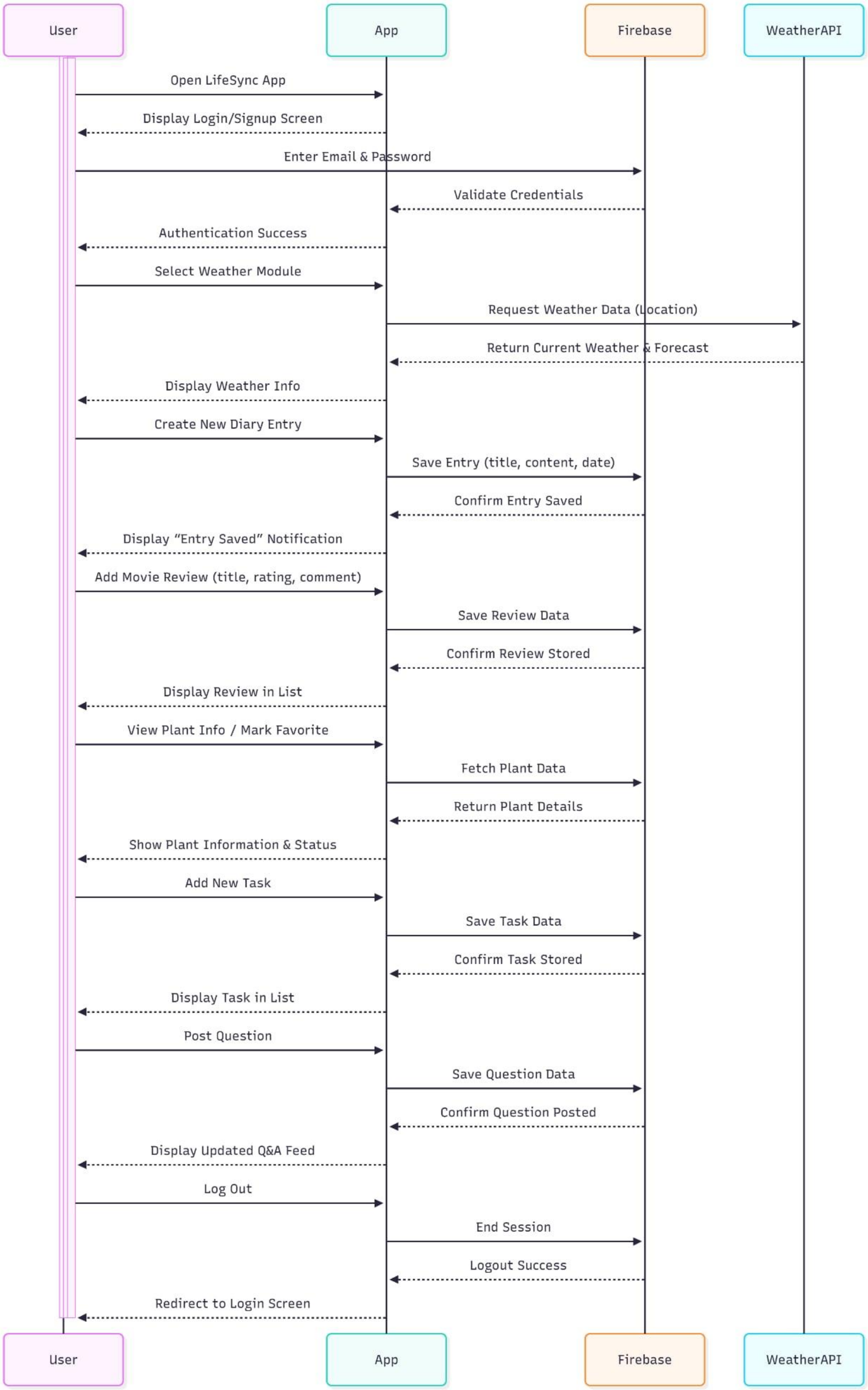


Figure 7. LifeSync’s Sequence Diagram

- Activity Diagram

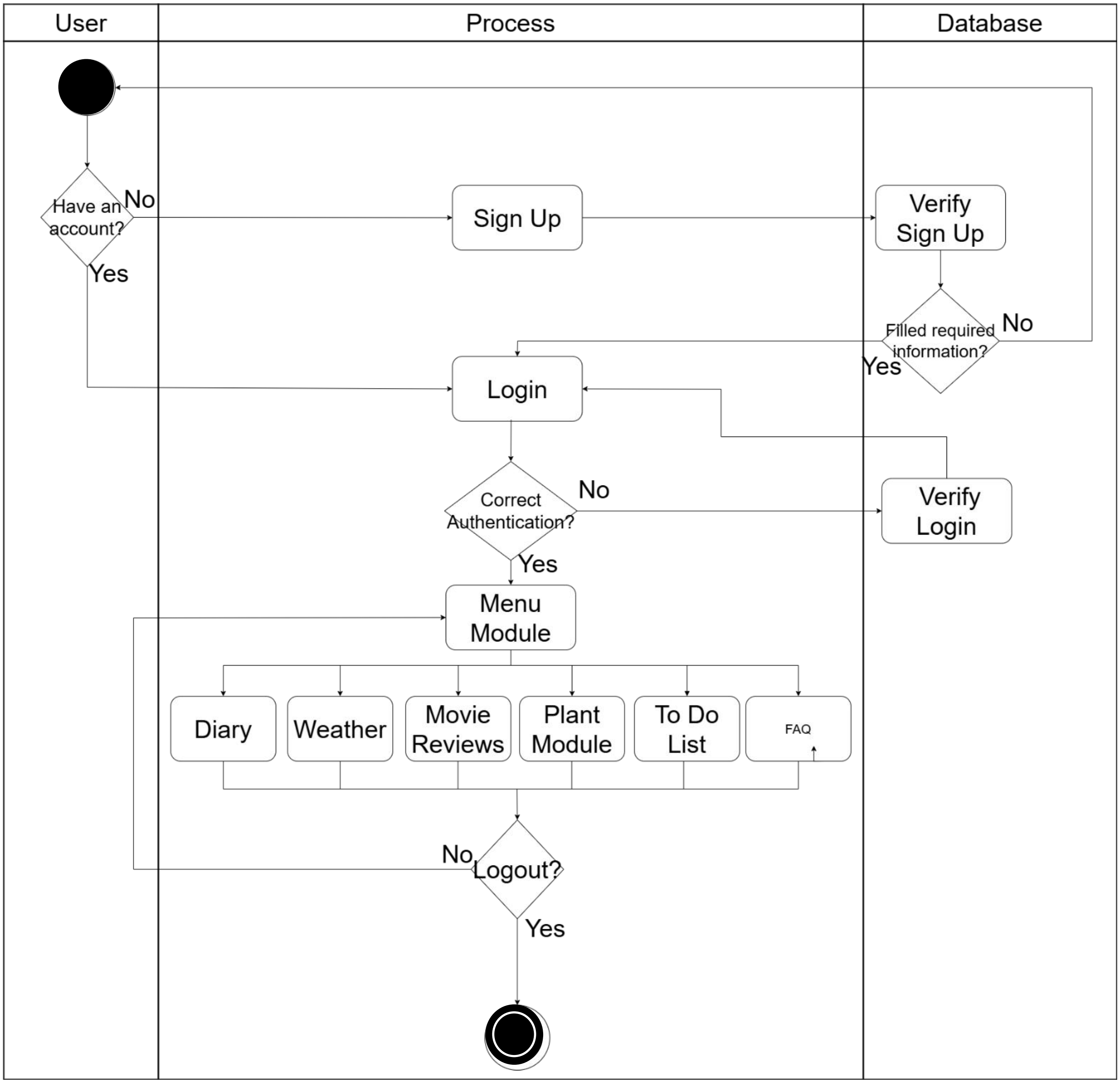


Figure 8. Life Sync's Activity Diagram

4.3.6 Risk Assessment/ Analysis

During the development of LifeSync, the proponents encountered several challenges that affected both its functionality and deployment. Technical issues such as system crashes, data loss, and compatibility problems arose due to software updates and variations in device performance. To address these, the proponents conducted extensive compatibility testing, implemented regular data backups, and performed routine system maintenance. Additionally, financial constraints and unexpected project expenses created budgeting challenges; however, these were managed by adopting cost-effective solutions and careful project planning. Through these measures, the proponents ensured the stability, security, and sustainability of the LifeSync application.

4.4 Design

4.4.1 Output and User-interface Design

- Color scheme/pallets

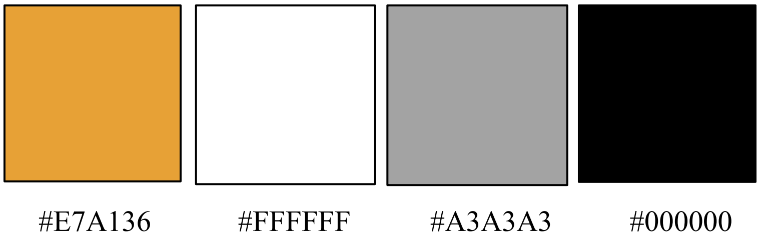


Figure 9. Colors

A. #E7A136

The proponents used this color in logo because it symbolizes energy and creativity.

B. #FFFFFF

The proponents used this color for background for simplicity and minimalist design for the user interface.

C. #A3A3A3

The proponents used this color to background to blend it with the white background and also creates a shadow in each module.

D. #000000

The proponents used this color to text so that the user can see or read the text clearly.

- Typography

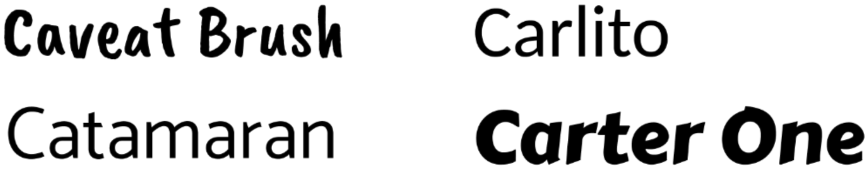


Figure 10. Typography

A. Caveat Brush

The proponents used this font for titles.

B. Catamaran

The proponents used this font for information purposes

C. Carlito

The proponents used this font is used in inputting text to create simplicity and also readable.

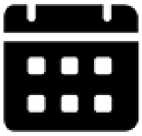
D. Carter One

The proponents used this font use for logo in each module.

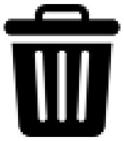
- Icons

Logo

Info Icon

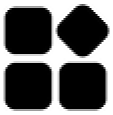


Calendar Icon



Delete Icon

Back Icon



Menu Icon

Diary Icon

Weather Icon

Add Icon

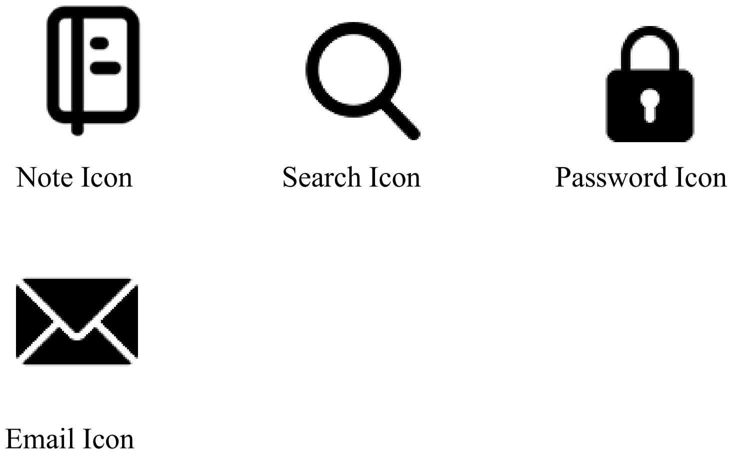


Figure 11. Icons

A. **Logo**

The proponents used this logo to represent that the application focuses on diaries and daily activities.

B. **Info Icon**

The proponents used this icon to provide users with helpful information or guidance about the application’s features.

C. **Calendar Icon**

The proponents used this icon to signify dates, schedules, and time-related activities within the app.

D. **Delete Icon**

The proponents used this icon to allow users to remove or delete unwanted entries, notes, or data.

E. **Back Icon**

The proponents used this icon to help users navigate back to the previous page or section.

F. **Menu Icon**

The proponents used this icon to give users access to various options or sections of the application through a navigation menu.

G. **Diary Icon**

The proponents used this icon to symbolize personal diary entries or journals created by the user.

H. **Weather Icon**

The proponents used this icon to display or indicate current weather information related to the user’s daily activities.

I. **Add Icon**

The proponents used this icon to allow users to create or add new entries, notes, or events.

J. **Delete Icon**

The proponents used this icon to remove any unnecessary or incorrect records, ensuring data management within the app.

K. **Note Icon**

The proponents used this icon to represent written notes, thoughts, or reminders that users can store in the application.

L. **Search Icon**

The proponents used this icon to enable users to quickly find specific diary entries, dates, or information within the app.

M. **Password Icon**

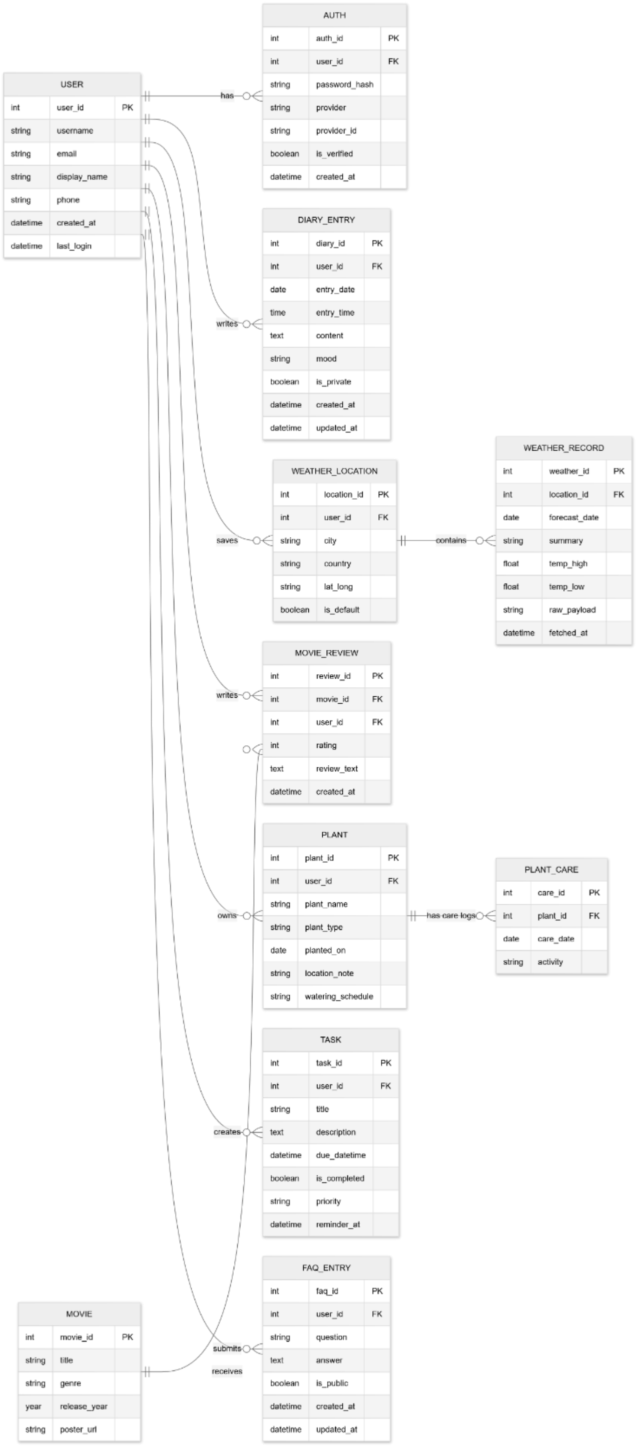
The proponents used this icon to indicate password input fields, ensuring account security and privacy.

N. **Email Icon**

The proponents used this icon to represent email input or communication features, such as user login or registration.

4.4.2 Data Design

- Entity Relationship Diagram



USER		
int	user_id	PK
string	username	
string	email	
string	display_name	
string	phone	
datetime	created_at	
datetime	last_login	

AUTH		
int	auth_id	PK
int	user_id	FK
string	password_hash	
string	provider	
string	provider_id	
boolean	is_verified	
datetime	created_at	

DIARY_ENTRY		
int	diary_id	PK
int	user_id	FK
date	entry_date	
time	entry_time	
text	content	
string	mood	
boolean	is_private	
datetime	created_at	
datetime	updated_at	

WEATHER_LOCATION		
int	location_id	PK
int	user_id	FK
string	city	
string	country	
string	lat_long	
boolean	is_default	

WEATHER_RECORD		
int	weather_id	PK
int	location_id	FK
date	forecast_date	
string	summary	
float	temp_high	
float	temp_low	
string	raw_payload	
datetime	fetched_at	

MOVIE_REVIEW		
int	review_id	PK
int	movie_id	FK
int	user_id	FK
int	rating	
text	review_text	
datetime	created_at	

PLANT		
int	plant_id	PK
int	user_id	FK
string	plant_name	
string	plant_type	
date	planted_on	
string	location_note	
string	watering_schedule	

PLANT_CARE		
int	care_id	PK
int	plant_id	FK
date	care_date	
string	activity	

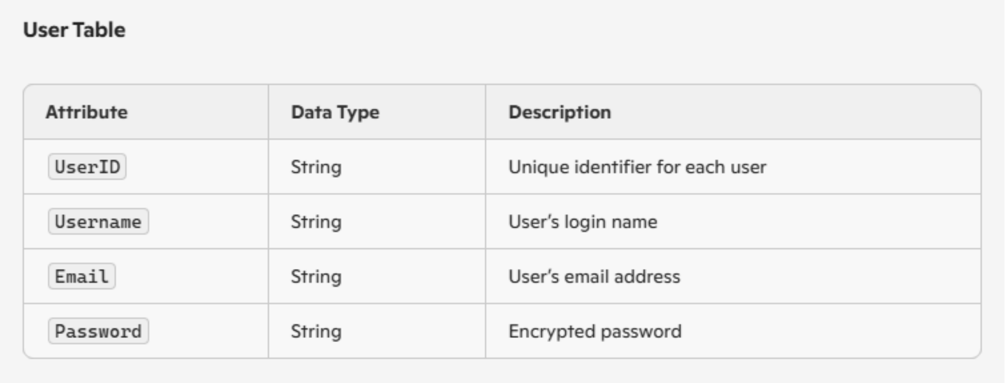
TASK		
int	task_id	PK
int	user_id	FK
string	title	
text	description	
datetime	due_datetime	
boolean	is_completed	
string	priority	
datetime	reminder_at	

MOVIE		
int	movie_id	PK
string	title	
string	genre	
year	release_year	
string	poster_url	

FAQ_ENTRY		
int	faq_id	PK
int	user_id	FK
string	question	
text	answer	
boolean	is_public	
datetime	created_at	
datetime	updated_at	

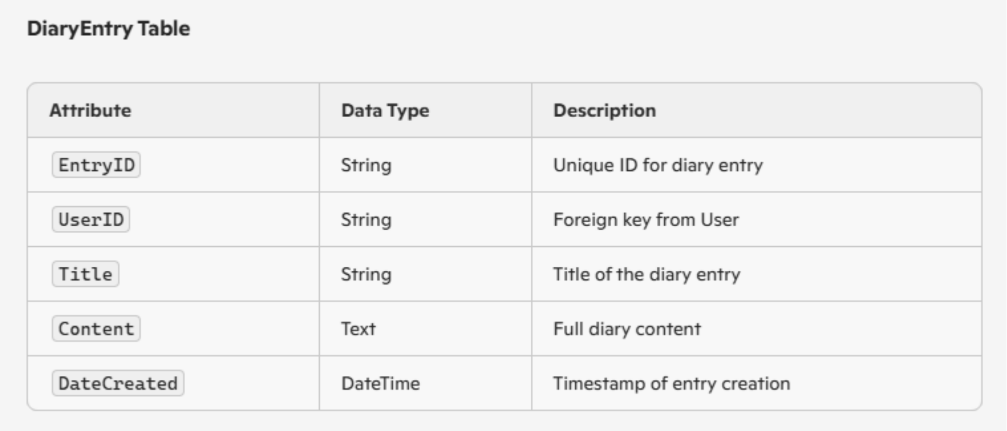
Figure 12. Entity Relationship Diagram

- Data Dictionary



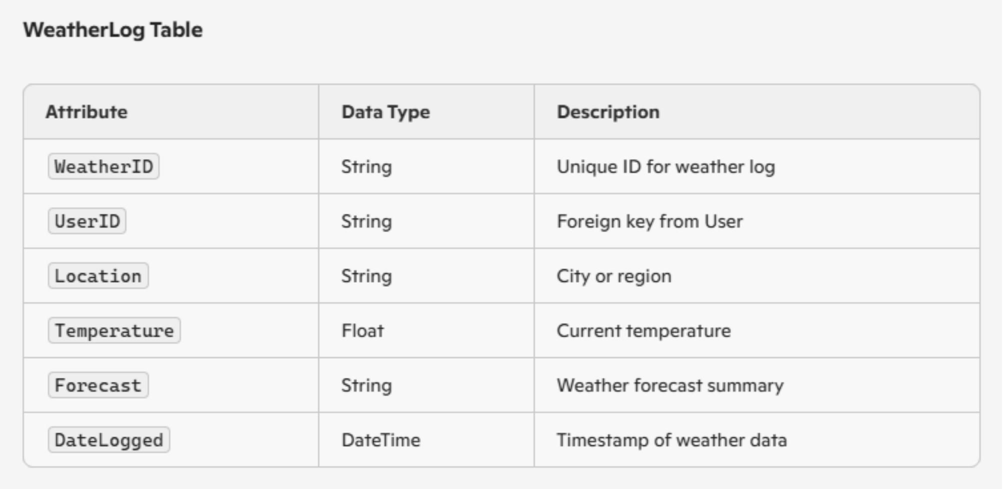
User Table

Attribute	Data Type	Description
`UserID`	String	Unique identifier for each user
`Username`	String	User's login name
`Email`	String	User's email address
`Password`	String	Encrypted password



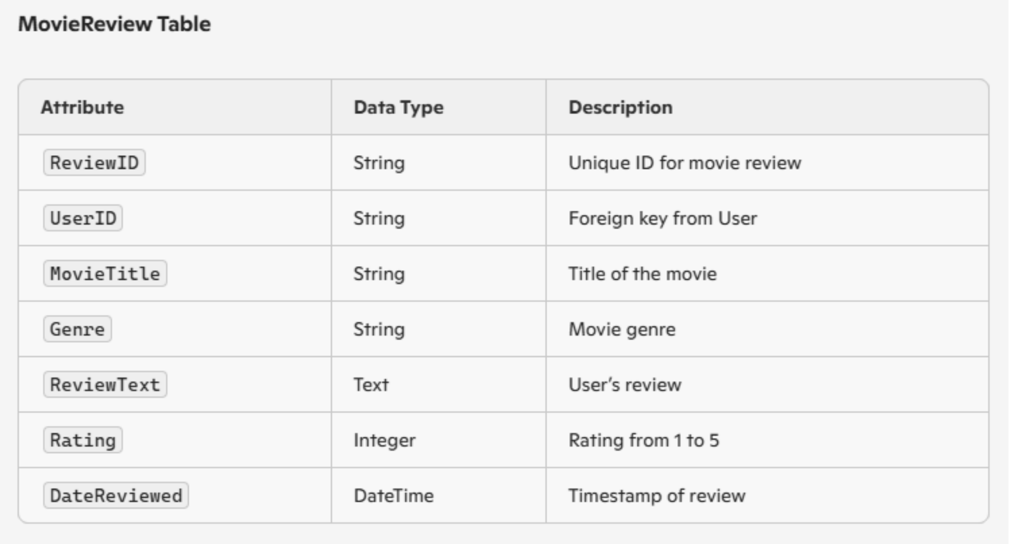
DiaryEntry Table

Attribute	Data Type	Description
`EntryID`	String	Unique ID for diary entry
`UserID`	String	Foreign key from User
`Title`	String	Title of the diary entry
`Content`	Text	Full diary content
`DateCreated`	DateTime	Timestamp of entry creation



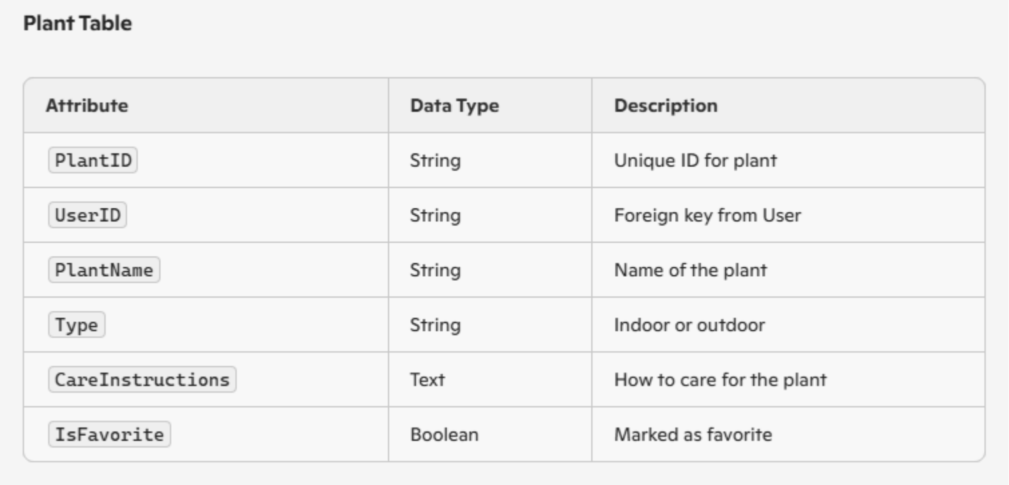
WeatherLog Table

Attribute	Data Type	Description
`WeatherID`	String	Unique ID for weather log
`UserID`	String	Foreign key from User
`Location`	String	City or region
`Temperature`	Float	Current temperature
`Forecast`	String	Weather forecast summary
`DateLogged`	DateTime	Timestamp of weather data



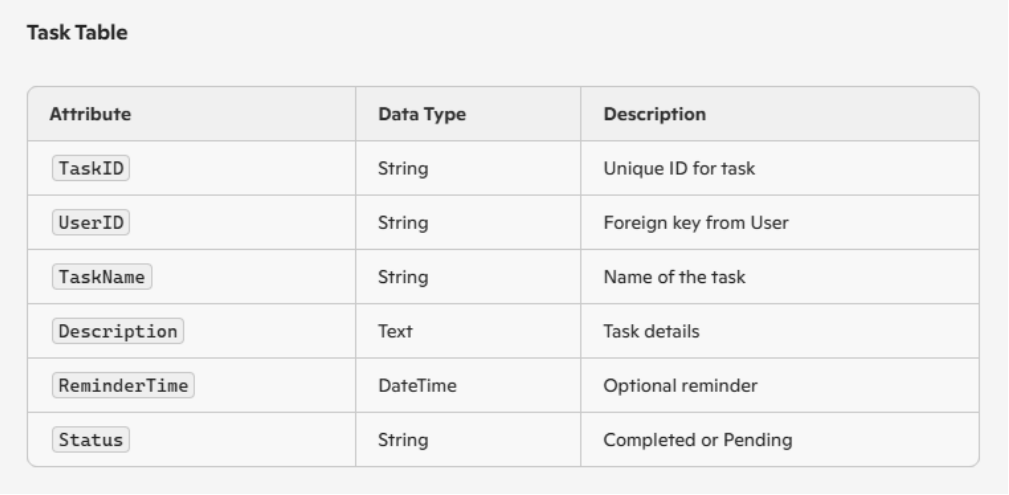
MovieReview Table

Attribute	Data Type	Description
`ReviewID`	String	Unique ID for movie review
`UserID`	String	Foreign key from User
`MovieTitle`	String	Title of the movie
`Genre`	String	Movie genre
`ReviewText`	Text	User’s review
`Rating`	Integer	Rating from 1 to 5
`DateReviewed`	DateTime	Timestamp of review



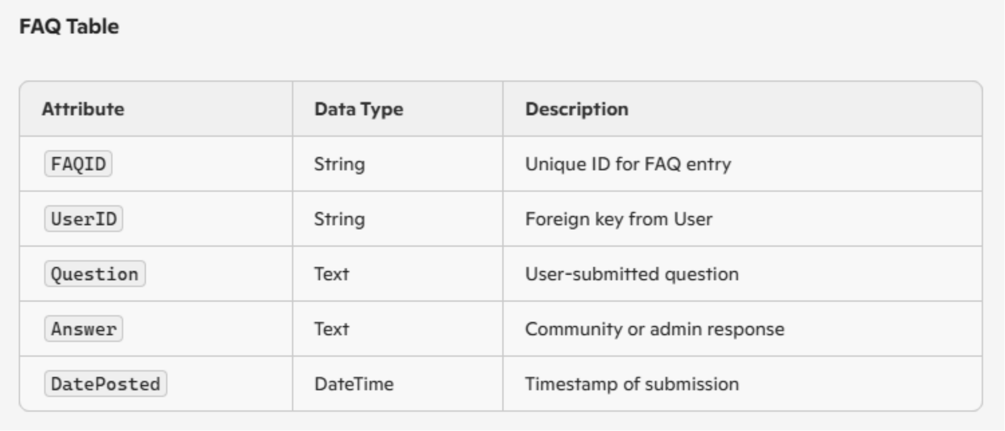
Plant Table

Attribute	Data Type	Description
`PlantID`	String	Unique ID for plant
`UserID`	String	Foreign key from User
`PlantName`	String	Name of the plant
`Type`	String	Indoor or outdoor
`CareInstructions`	Text	How to care for the plant
`IsFavorite`	Boolean	Marked as favorite



Task Table

Attribute	Data Type	Description
`TaskID`	String	Unique ID for task
`UserID`	String	Foreign key from User
`TaskName`	String	Name of the task
`Description`	Text	Task details
`ReminderTime`	DateTime	Optional reminder
`Status`	String	Completed or Pending



FAQ Table

Attribute	Data Type	Description
`FAQID`	String	Unique ID for FAQ entry
`UserID`	String	Foreign key from User
`Question`	Text	User-submitted question
`Answer`	Text	Community or admin response
`DatePosted`	DateTime	Timestamp of submission

4.4.3 System Architecture

- Network Model

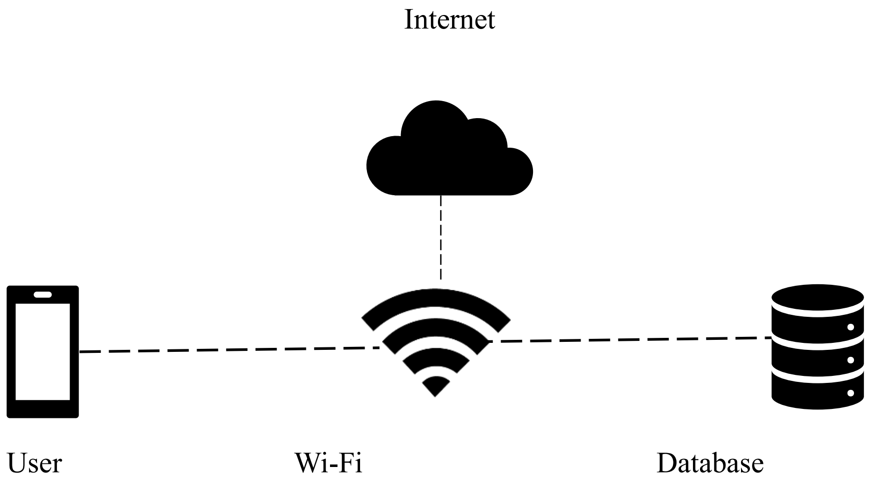


Figure 13. Network Model

- Network Topology

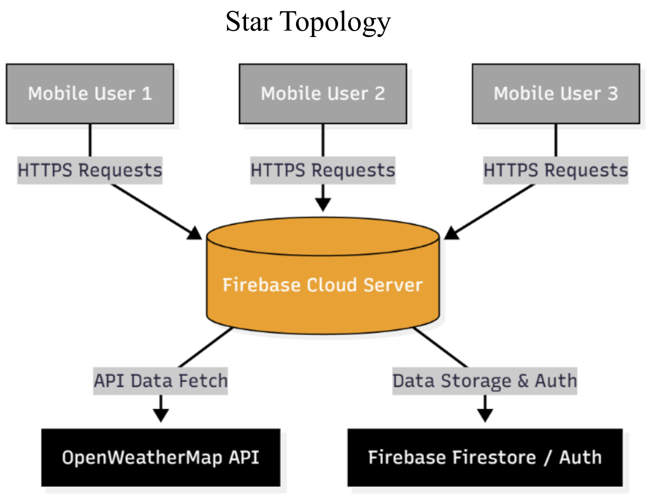


Figure 14. Network Topology

- Security

  The LifeSync mobile app puts user privacy and data protection first. All communications between the client application and the Firebase cloud services are protected, guaranteeing that information sent over the Internet is confidential and secure from exploitation or unauthorized access.

4.5 Development

4.5.1 Software Specification

The LifeSync mobile application was developed using the following software components:

Software	Version	Source
Flutter	3.13.6	https://flutter.dev
Dart	3	https://dart.dev
Firebase	Latest 2025	https://firebase.google.com
Figma	Latest 2025	https://figma.com
Canva	Latest 2025	https://www.canva.com
Android Studio	Narwhal 4	https://developer.android.com/studio

Table 6: Software Specification

4.5.2 Hardware Specification

Category	Hardware Component	Minimum Requirement	Purpose / Description
Development Device	Processor	Intel i3 or higher	To ensure smooth operation during app development and testing.
	RAM	8GB minimum	Provides sufficient memory for running Android Studio and emulators efficiently.
	Storage	256GB HDD/SSD	Stores project files, SDKs, and application resources.
	Internet	Stable connection	Required for API calls, Firebase integration, and online libraries.
User Device (Mobile)	Operating System (Android)	Android 8.0 (Oreo) or higher	Ensures compatibility with the latest app features and updates.
	Operating System (iOS)	iOS 12 or higher	Enables iPhone users to install and run the app smoothly.
	RAM	Minimum 2GB	Supports smooth performance and loading of application components.
	Internet	Required connection	Needed for cloud-based synchronization and real-time features.

Table 6: Hardware Specification

4.5.3 Program Specification

PROGRAM SPECIFICATION OF LIFESYNC: ANDROID MOBILE APPLICATION
Programming Language: Dart
Events:
• Start Screen will be displayed when opening the application.
• When tap it will go to the Authentication screen.
• The application close upon exiting.
Module: Authentication Module
Purpose: To allow users to securely log in or create an account.
Events:
• Shows the Log In.
• Shows the Sign Up.
• Navigation to Menu Module after successful authentication.
Module: Weather Module
Purpose: To provide real-time weather information based on the user's location.
Events:
• Displays current weather conditions and temperature.
• Displays forecast for upcoming days.
• Option to refresh weather data.
• Navigation back to Menu Module.
Module: Diary Module
Purpose: To allow users to record personal thoughts and experiences.
Events:
• User can create a new diary entry.
• User can edit or delete existing entries.
• User can search entries by title.
• Navigation back to Menu Module.

Module: Menu Module
Purpose**:** To serve as the main hub for LifeSync’s special features.
Event:
• Displays buttons for the four special features:
o Movie Review Module
o Plant Module
o To-Do List Module
o Q&A Module
• Navigation to Weather Module**,** Diary Module**,** and Settings.
Module: Movie Review Module
Purpose**:** To allow users to browse, write, and manage movie reviews.
Event:
• User can view movie reviews.
• User can add a new review.
• User can edit or delete their review.
• Navigation back to Menu Module.
Module: Plant Module
Purpose**:** To provide information and care tips for plants.
Event:
• Displays a list of plants with images and descriptions.
• User can view care instructions for each plant.
• User can mark plants as favorites.
• Navigation back to Menu Module**.**
Module: To-Do List Module
Purpose**:** To help users manage tasks and daily activities.
Events:
• User can create a new task.
• User can mark tasks as completed or pending.
• User can edit or delete tasks.

• Navigation back to Menu Module**.**
Module: Q&A Module
Purpose**:** To allow users to ask questions and get answers from the community.
Events**:**
• User can post a new question.
• User can view answers and comments.
• User can reply or like answers.
• Navigation back to Menu Module**.**

Table 7: Program Specification

4.5.4 Programming Environment

- **Front- End**

  Dart with Flutter is used in the development of LifeSync's front-end, providing for the development of an accessible and appealing user interface. All screens, buttons, and navigation menus are included, along with the four unique feature modules—Movie Review, Plant, To-Do List, and Q&A. The primary IDE is Android Studio, and UI/UX prototypes are designed using Figma and Canva to ensure consistent design and usability throughout the program.

- **Back End**

  Dart with Firebase runs LifeSync's back-end, handling real-time updates, data storage, and user authentication. Log-in and sign-up procedures are secured by Firebase Authentication, and tasks, movie reviews, diary entries, Q&A posts, and plant data are stored in Firestore. Firebase Cloud Functions manages additional backend functions, such data processing or automated notifications, ensuring reliable and efficient user interactions and data sharing.

- **Programming Considerations and Issues**

  Current adjustments to modules like Q&A and To-Do List, offline accessibility, and front-end and back-end connection are all carefully considered during development. While app performance is improved across a number of Android devices and screen sizes, user data security and privacy are given first priority. To ensure that navigating through LifeSync's features is quick and informative, it's also important to keep the interface simple and easy to use.

4.6 Installation Processes

The aim of the LifeSync installation process is to switch from manual or disconnected personal management techniques to a complete mobile application. The installation will mostly involve installing the new application on Android smartphones for end users, as there is currently no option that is similar to LifeSync. To ensure an easy transition and minimize interruptions, a gradual, step-by-step changeover method will be employed for this study. This method helps people to become familiar with LifeSync's functions while adopting it progressively.

To install LifeSync on an Android device, follow these steps:

1. Open the Google Play Store on your Android device.
2. In the search bar, type “LifeSync” and press Search.
3. Locate the LifeSync application in the search results and tap on it.
4. Tap the Install button to download and install the application.
5. Once installation is complete, tap Open to launch LifeSync.
6. On the Start Screen, proceed to Sign Up or Log In to start using the app.

LIFESYNC: ANDROID MOBILE APPLICATION

Members:

De Vero, Eunice Jayce F.

Maninang, Ken G.

Natino, Rose Ann M.

Salunga, Joyce Kenn L.