

# Development of Household Management Mobile Application

Lee Shuan Qian<sup>1</sup>, Firkhan Ali Hamid Ali<sup>1\*</sup>

<sup>1</sup> *Fakulti Sains Komputer dan Teknologi Maklumat,  
Universiti Tun Hussein Onn Malaysia, Parit Raja, Batu Pahat, 86400, MALAYSIA*

\*Corresponding Author: [firkhan@uthm.edu.my](mailto:firkhan@uthm.edu.my)

DOI: <https://doi.org/10.30880/aitcs.2025.06.01.013>

## Article Info

Received: 13 June 2024

Accepted: 8 May 2025

Available online: 30 June 2025

## Keywords

Household management, Mobile application, Android-based, Family collaboration

## Abstract

Household management involves the strategic coordination and organization of resources within a family setting to meet the household's objectives and day-to-day needs. The "HouseLink" household management mobile application targets the prevalent disorganization in family management methods. Presently, disjointed approaches like handwritten notes and digital calendars hinder coherence and collaboration within households. To address this, the project aimed to create a specialized Android-based mobile application utilizing Android Studio IDE and Java for backend development, facilitating streamlined household management. Modules encompass user management, tasks, meal planning, and record-keeping, fostering collaboration and accessibility among family members. The study's significance lies in its potential to centralize household management, enhance collaboration, and simplify daily routines. Through a user-friendly interface, "HouseLink" aspires to empower families in achieving more coordinated and harmonious daily living.

## 1. Introduction

Household management refers to the strategic planning, regulation, and assessment of the family's resources to achieve the family's objectives [1]. It naturally arises from human interactions within the home setting. As a family is formed, managing the household becomes a significant responsibility for all family member. The main household responsibilities in both urban and rural communities encompass cleaning, laundry, meal preparation, obtaining water and general house maintenance [2]. Household management plays a crucial role within every family, significantly impacting the overall health, happiness, and overall well-being of its members.

Currently, household management processes often rely on various disconnected methods, such as handwritten notes, digital applications, and physical calendars. For instance, a family might use sticky notes on the fridge to track groceries, a digital note app on their smartphones for shopping lists, and a physical calendar for important dates and events. This fragmented approach often leads to disorganized data, as critical household information is scattered across various formats and platforms. Moreover, there is a lack of consistency in managing household activities and records, resulting in limited accessibility and collaboration within the family.

The proposed solution is a household management mobile application designed to centralize and streamline home organization for families. By addressing data fragmentation and inconsistent management practices, the application simplifies tasks and promotes collaboration. This system is designed for individuals seeking to establish a well-organized household management system within their family, yet it is versatile enough to be utilized by anyone interested in efficient household management. The application will be developed using Android Studio IDE with Java programming language and Firebase Firestore as the database. It will consist of five modules

including User management, Daily task, Meal planning, Record keeping, and Date and event. Three objectives have been set to achieve the aim of this study, which are:

1. To design a household management mobile application tailored for family usage
2. To develop a household management system using Android technology
3. To test and evaluate the functionality of the developed mobile application

The rest of the paper is organized as follows: Section 2 discusses the related works on household management and the review between the existing systems and the proposed system. Section 3 discusses the development methodology of the proposed mobile application. Section 4 discusses the analysis and design, implementation and testing of the proposed application. Section 5 discusses the conclusion of the proposed application.

## 2. Related Work

The literature review of the study and the analysis between the existing application and the proposed application are discussed in this section.

### 2.1 Smartphone Technology Utilization in Household Management

The integration of smartphone technology has significantly transformed household management practices, offering a versatile platform for overseeing various domestic activities. Smartphones offer three primary benefits in household management which are communication, information dissemination, and transaction facilitation [3]. Communication features encompass voice, text, and video capabilities, fostering seamless connectivity among family members. Furthermore, smartphones serve as information hubs that provide access to websites and social media platforms. For example, platforms like Instagram and Facebook accessed through smartphones often serve as a gateway to discovering cooking recipes [4].

Moreover, smartphones extend their utility beyond mere communication and information access, significantly impacting household-related functions. The crucial role of smartphones in accessing e-commerce platforms, facilitating essential functions such as food ordering, shopping, and transactions [5]. With a smartphone, individuals can conveniently purchase groceries by simply browsing through social media or e-commerce platforms. Furthermore, smartphones enable the creation of record-keeping lists and task reminders for household responsibilities. This particular feature serves as a pivotal tool in efficiently managing and organizing household chores, fostering a more streamlined approach to handling daily responsibilities.

### 2.2 Study of Existing Application

A comparison table is displayed and compared between the existing application and the proposed application based on their specific features. The chosen existing application are Homer [6], Cozi [7] and FamilyWall [8]. The comparison is diagnosed and verified by different aspects as depicted in Table 1.

The first mobile application, Homer, is a home management application designed to store and access household information anytime, anywhere. It offers six main functions: Home Timeline for tracking events, shareable to-do lists, inventory management, expense tracking, AI-assisted Homer Helper, and a 2D floorplan generator. The proposed household management app will also implement the features of shareable task lists to enhance collaboration and organization within the family.

The second mobile application, Cozi, enhances family organization with features like a shared, color-coded Family Calendar for managing schedules, To-Do Lists for tracking tasks and chores, and Shopping Lists for compiling and sharing needed items. It also offers Cozi Meals, which simplifies meal planning by storing recipes, scheduling dinners, and integrating ingredients into shopping lists. The proposed household management system incorporates similar meal planning functionalities, allowing users to organize recipes, plan meals, and streamline grocery shopping, thereby supporting efficient routines and healthier eating habits.

The third mobile application, FamilyWall, includes features of a Shared Family Calendar for managing schedules and reminders, collaborative Shopping Lists, To-Do Lists for task organization and progress tracking, Family Messaging for quick communication, a Family Gallery for private photo and video sharing, an Important Contacts directory, and a Family Locator for locating members and setting location-based alerts. The proposed household management system aims to emulate FamilyWall's user-friendly interface to ensure seamless navigation and accessibility. By prioritizing an intuitive design, the proposed system aspires to be a convenient and efficient tool for streamlined household organization and family coordination.

A comparison between existing applications and the proposed application is conducted to identify features in current applications that can be used to enhance the proposed application. Table 1 presents the findings of a study on existing applications, highlighting various features identified in those applications.

**Table 1** Comparison of the existing application with the proposed application

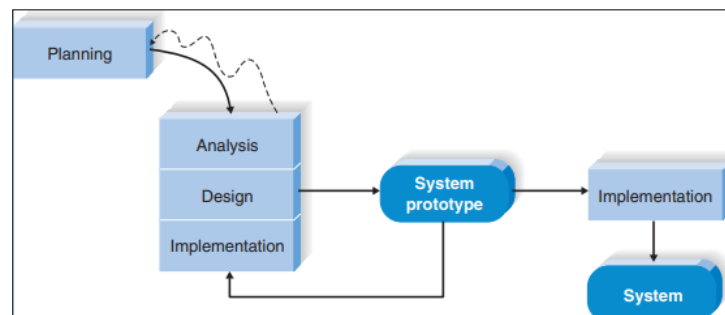
Categories	Feature	Homer [6]	Cozi [7]	FamilyWall [8]	Proposed application
Platform/ Operating System	Android-based	√	√	√	√
	iOS-based	√	√	√	X
	Web-based	√	√	√	X
User Experience Design (UX)	Clean navigation	√	√	√	√
	Simple interface	√	√	√	√
	Attractive user interface	√	X	√	√
User Interface Design (UI)	Advertisement free	√	X	√	√
	Concise contact	√	X	√	√
Functions	User management (register, log in/out)	√	√	√	√
	Collaborative	√	√	√	√
	Daily Task	√	√	√	√
	Meal Planning	X	√	√	√
	Expense/budget	√	X	√ (premium)	√
	Store household information	√	X	X	√
	Date and event calendar	X	√	√	√
	Family Gallery	X	X	√	X

### 3. Methodology/Framework

An appropriate model is chosen to ensure all the progress in completing this project is smooth and completed on schedule. The chosen model to develop this project is Prototype Model.

#### 3.1 Prototype Model

In software development, methodology provides a systematic framework to organize tasks, manage resources, handle complexities, and achieve project goals efficiently [9]. The choice of methodology significantly influences the development process [10]. The prototyping model is utilized for developing the household management mobile application. This model involves constructing, testing, and refining a prototype until an acceptable version is achieved [11]. This model is chosen for the household management mobile application development due to its iterative nature, which permits continuous enhancement, ensuring that the application aligns closely with user needs and expectations. The prototyping model consists of five main phases: planning, analysis, design, implementation, and testing, as shown in Fig. 1.



**Fig. 1** The Prototyping Process Model [10]

##### 3.1.1 Planning Phase

The prototype model starts with the planning phase, emphasizing project and product management. Tasks include setting priorities, strategy development, selecting development frameworks, and scheduling activities, visualized with Gantt Charts. The outputs of this phase include project plans, timelines, and procurement needs. Research involves gathering information from online resources and journals for the proposed application. Comparative analysis with existing systems is conducted to evaluate features and functionalities.

### 3.1.2 Analysis Phase

During the analysis phase, data is scrutinized to match project goals, translating them into specific system functions. This stage involves gathering system requirements through data analysis and user interviews. Object-Oriented Analysis and Design (OOAD) ensures functionalities meet requirements, integrating data and functions seamlessly. To visually articulate the system's flow, a comprehensive suite of diagrams including the system architecture diagram, UML diagrams such as use-case diagram, sequences diagram and activity diagram and class diagram are employed. Other than that, the software and hardware requirements for this project are outlined in this phase.

### 3.1.3 Design Phase

During the design phase, the application begins to take shape through the creation of visual content based on information gathered in earlier phases. Key activities include designing for the target audience and using wireframing to guide the user interface design. This phase also involves converting the logical system design from the analysis phase into a physical structure, including user interfaces and database architectures. Figma are used for constructing the interface design, and the database is designed to ensure data integrity throughout development and beyond.

### 3.1.4 Implementation Phase

The implementation phase begins after getting user acceptance of the proposed application. It involves creating a functional prototype based on requirements and design specifications from previous phases, translating conceptual models into a working prototype. The Android Studio IDE is used for developing the household management mobile application, with Java for backend functionality and XML Android Programming for frontend. Firebase is integrated for system configuration, ensuring security and recovery protocols. An initial rudimentary system prototype will be developed for user testing and feedback gathering.

### 3.1.5 Testing Phase

The testing phase is crucial for validating the proposed application's functionality, reliability, and alignment with user requirements. Rigorous testing procedures are conducted to examine all lines of code for faults, bugs, and flaws. This ensures the system's dependability and compliance with both functional and non-functional criteria established during user analysis. User acceptability testing will be conducted for thorough review and engagement with the planned application.

## 3.2 System Development Workflow

The process of the system development summarizes the main phases of prototyping including planning, design, development, implementation and testing. Each phase encompasses the activities to achieve deliverables. Table 2 depicts the workflow for the development of the proposed application.

**Table 2** *Workflow of the proposed application development*

Phase	Task	Output
Planning	<ul style="list-style-type: none"> <li><input type="checkbox"/> Identify problem statements, objective, scope of the project, expected outcome and significant of the project.</li> <li><input type="checkbox"/> Construct a work schedule</li> <li><input type="checkbox"/> Conduct online research for resources and article that related to the title</li> <li><input type="checkbox"/> Study features and functionality of similar existing systems.</li> </ul>	<ul style="list-style-type: none"> <li><input type="checkbox"/> Project proposal</li> <li><input type="checkbox"/> Gantt chart</li> <li><input type="checkbox"/> Literature review</li> <li><input type="checkbox"/> Comparison between existing application and the proposed application</li> </ul>

**Table 2** Workflow of the proposed application development (continue)

Phase	Task	Output
Analysis	<input type="checkbox"/> Conduct informal interviews <input type="checkbox"/> Identify hardware and software requirements Identify functional and non-functional requirements	<input type="checkbox"/> User requirements <input type="checkbox"/> Hardware and software requirement <input type="checkbox"/> Functional and non-functional requirements
Design	<input type="checkbox"/> Construct system architecture diagram <input type="checkbox"/> Design use-case diagram, sequences diagram and activity diagram <input type="checkbox"/> Design interface of the system <input type="checkbox"/> Design ERD and database	<input type="checkbox"/> System architecture diagram <input type="checkbox"/> UML diagram (use-case diagram, sequence diagram, activity diagram) <input type="checkbox"/> Interface design <input type="checkbox"/> ERD design <input type="checkbox"/> Database design
Implementation	<input type="checkbox"/> Develop the system module <input type="checkbox"/> Complete system integration <input type="checkbox"/> Establish database connection <input type="checkbox"/> Develop a simple system prototype initially for testing purpose	<input type="checkbox"/> Proposed system <input type="checkbox"/> Error found and fixes
Testing	<input type="checkbox"/> Conduct system testing <input type="checkbox"/> Investigate the area of improvement	<input type="checkbox"/> Fix and improve the bugs <input type="checkbox"/> Fix and ready to release the proposed system

## 4. Result and Discussion

Analysis and design involve the systematic examination and creation of systems, where analysis focuses on understanding requirements and constraints, and design involves developing detailed plans to meet those requirements and constraints effectively.

### 4.1 Analysis and Design

Requirement analysis is a method used to ascertain user needs for either developing or updating an application. This process involves thorough analysis, documentation, validation, and management of software or system requirements. At this stage, both functional and non-functional requirements of the proposed application are precisely defined and extensively presented.

#### 4.1.1 System Requirement Analysis

Requirement analysis is a method used to ascertain user needs for either developing or updating an application. This process involves thorough analysis, documentation, validation, and management of software or system requirements. At this stage, both functional and non-functional requirements of the proposed application are precisely defined and extensively presented in Table 3 and Table 4.

**Table 4** Functional requirements of the proposed system

No.	Module	Description
1	User management	The application should allow user to register, login, log out and update their profile.
2	Daily task	The application should allow user to organize and track household tasks collaboratively, and upon completion, user can mark task as done. It should provide the record of pending and completed task for easily tracking.

**Table 4** *Functional requirements of the proposed system (continue)*

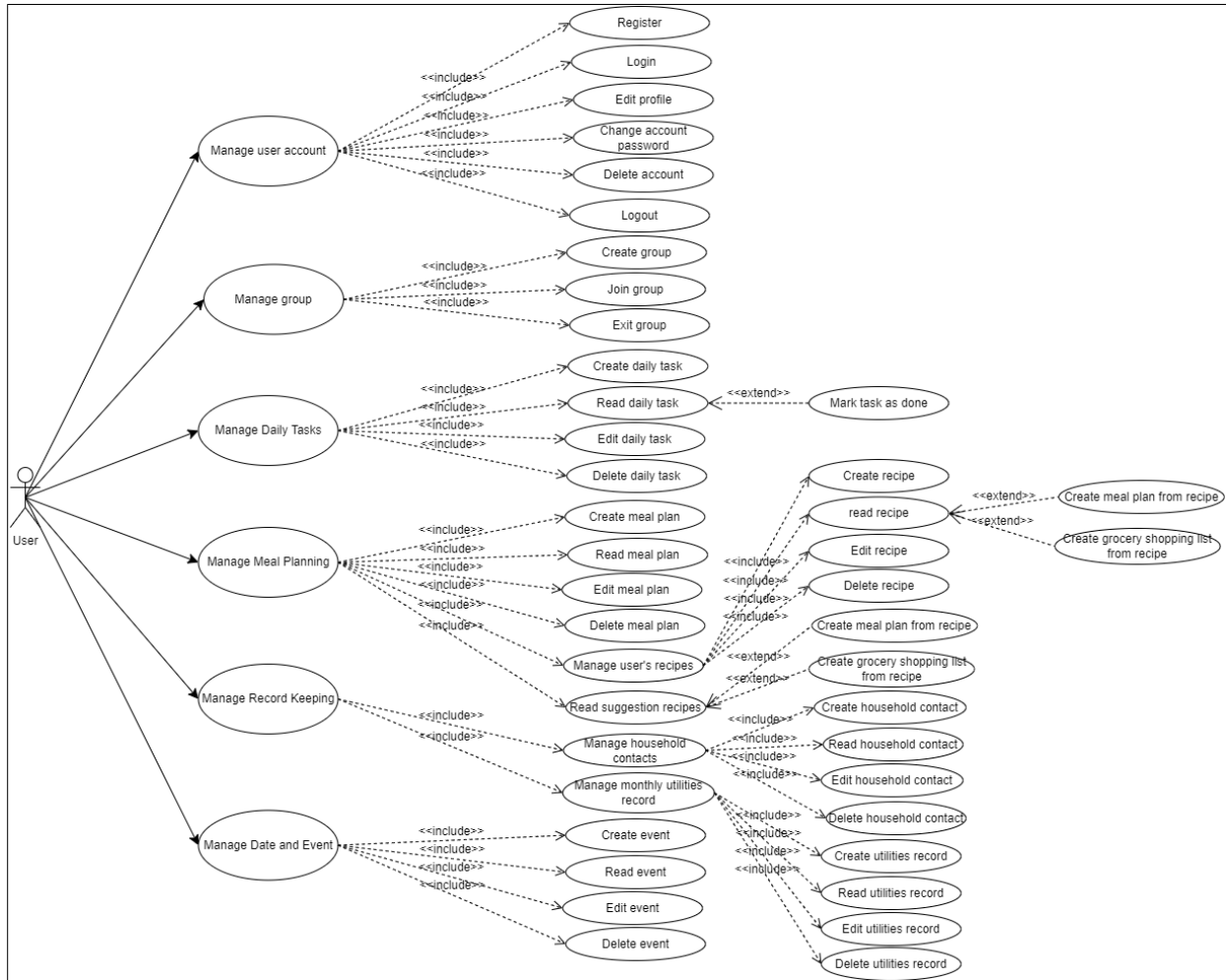
No.	Module	Description
3	Meal planning	The application should allow user to organize their meal plan such as add meal plan from recipe suggestion or customize their own meal plan. It should allow user to store their own recipe for reference and user can add grocery shopping task of the recipe's ingredients from the recipe collection.
4	Record-keeping	The application should allow user to organize crucial household information such as useful contact and utilities record
5	Date and event	The application should allow user to manage family events, appointment and important dates.

**Table 5** *Non-functional requirements of the proposed system*

No.	Module	Description
1	Performance	The application should respond quickly to user interactions, ensuring minimal latency in loading data or performing actions.
2	Security	Users can access the system only by providing correct login credentials. Besides, users can access shared family information only if they belong to the specific family group within the application.
3	Operational	The application should have always-on data connection to allow real-time database updates.
4	Availability	The application should allow user to organize crucial household information
5	Date and event	The application should allow user to manage family events, appointment and important dates

#### 4.1.2 Use Case Diagram

Use case diagram serves as a graphical representation illustrating how users interact with a system, displaying the various functionalities or actions users can perform. It provides a visual overview of the system's functionalities and the relationships between different actors and use cases. Fig. 2 shows the use case diagram for the household management mobile application. There is only one main actor which is the user. The major use cases include managing user accounts, groups, daily tasks, meal planning, record keeping, and dates and events.



**Fig. 2** Use case diagram of the proposed Application

### 4.1.3 Activity Diagram

Activity diagram depicts the workflow or flow of activities within a system, displaying the sequence of actions or steps required to accomplish a specific task or process. Fig. 3 depicts the activity diagram of the proposed application. The application starts with login or register. After successfully authentication, user will be redirected to the home page with displayed joined group. User can access to the user profile at home page and manage their account such as edit profile, change password and logout. Besides, user that had not any joined group need to create a group or join existing group with correct group code. Then, user can access to group dashboard with four main functionalities displayed: Daily Task, Meal Planning, Record Keeping and Date and Event.

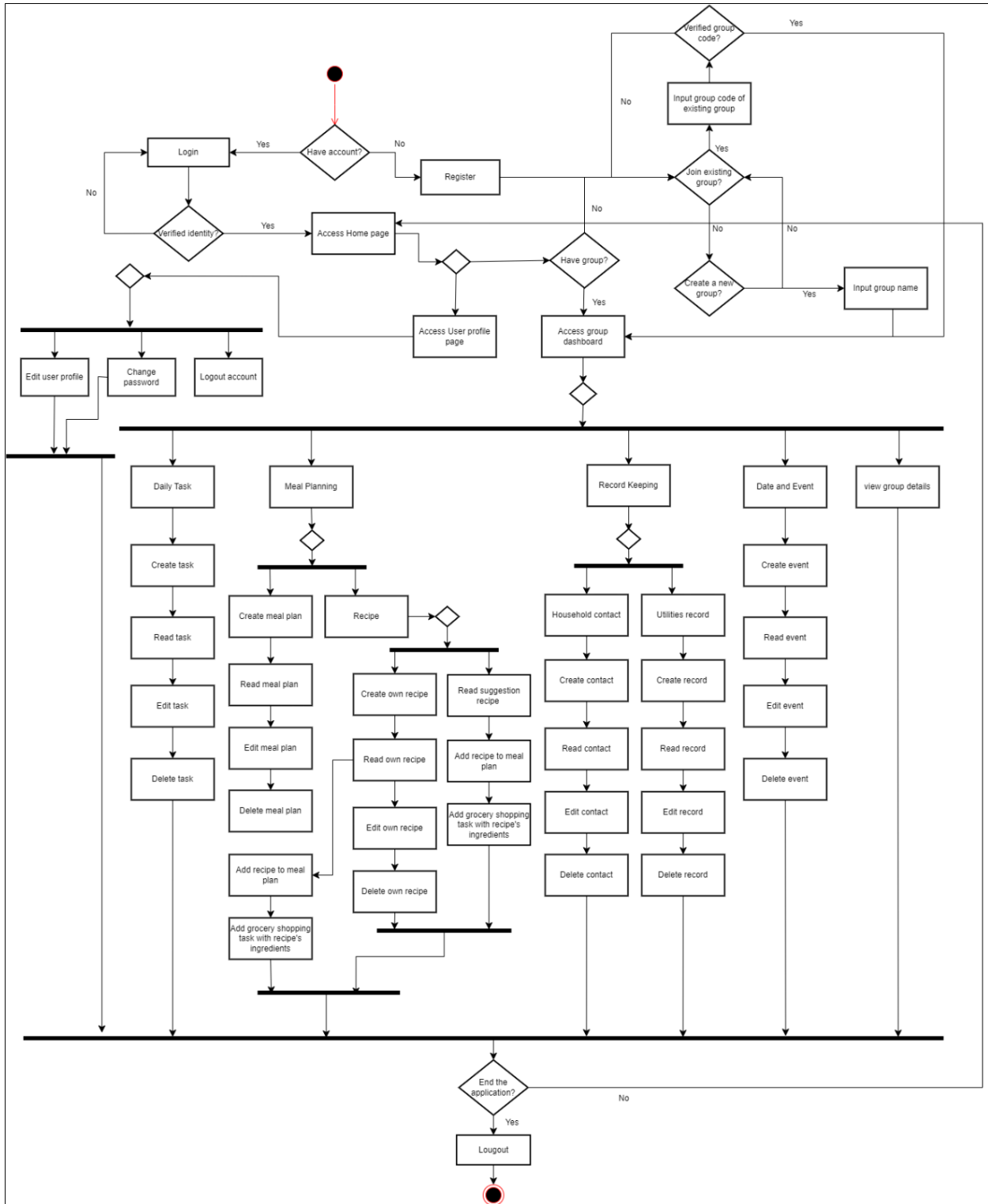


Fig. 3 Activity diagram of the proposed Application

#### 4.1.4 Class Diagram

A class diagram is a static structure diagram in UML that illustrates a system's classes, attributes, operations, and object relationships, modeling the static view of an application and its interactions. Fig. 4 illustrate the class diagram of the proposed application. There are eight classes including, User, Group, DailyTask, MealPlanning, Recipe, ContactRecord, UtilitiesRecord and DateAndEvent. Each class has attributes and methods for performing operations. Relationships are shown as associations, where Users can create Groups and manage DailyTasks, MealPlanning, Recipes, ContactRecord, UtilitiesRecord, and DateAndEvent.

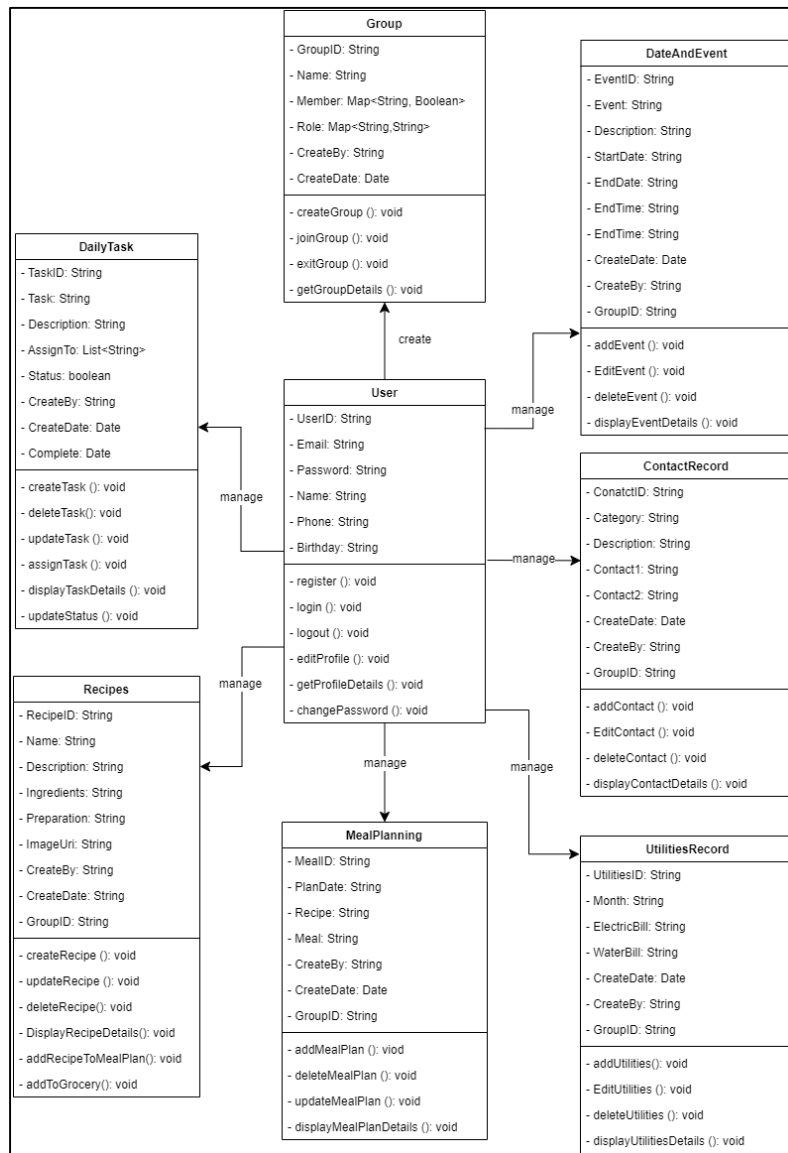


Fig. 4 Class diagram of the proposed Application

### 4.2 Implementation

During the implementation phase, the conceptual design is transformed into a functional system, ensuring the proposed solution is successfully realized. The household management mobile application is developed using Android Studio, employing Java for backend functionality and XML for frontend visualization. The system configuration relies on Firebase for various services.

Firebase Cloud Firestore serves as the primary database, storing and managing the application's data. Firebase Storage is utilized for storing image data, and Firebase Authentication handles user authentication, ensuring secure and efficient access to the application. To integrate Firebase into the application, the Firebase SDK is installed and paired with the Android project. The dependencies in Fig. 5 are added to the 'build.gradle' file at app level to include Firebase services in the project. Adding these dependencies is crucial as it ensures that the necessary Firebase libraries are available to the application, enabling seamless interaction with Firebase services for database management, storage, and user authentication.

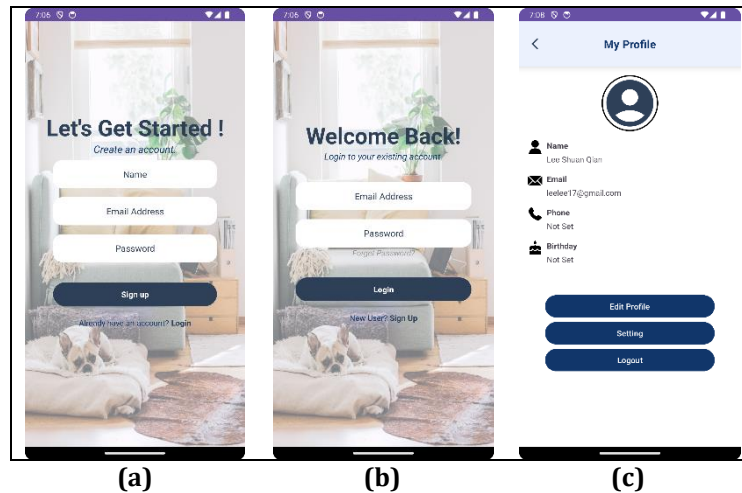
```

implementation 'com.google.firebase:firebase-auth:23.0.0'
implementation 'com.google.firebase:firebase-firestore:25.0.0'
implementation 'com.google.firebase:firebase-storage:21.0.0'
    
```

Fig. 5 Dependencies declared for Firebase Authentication, Cloud Firestore and Cloud Storage

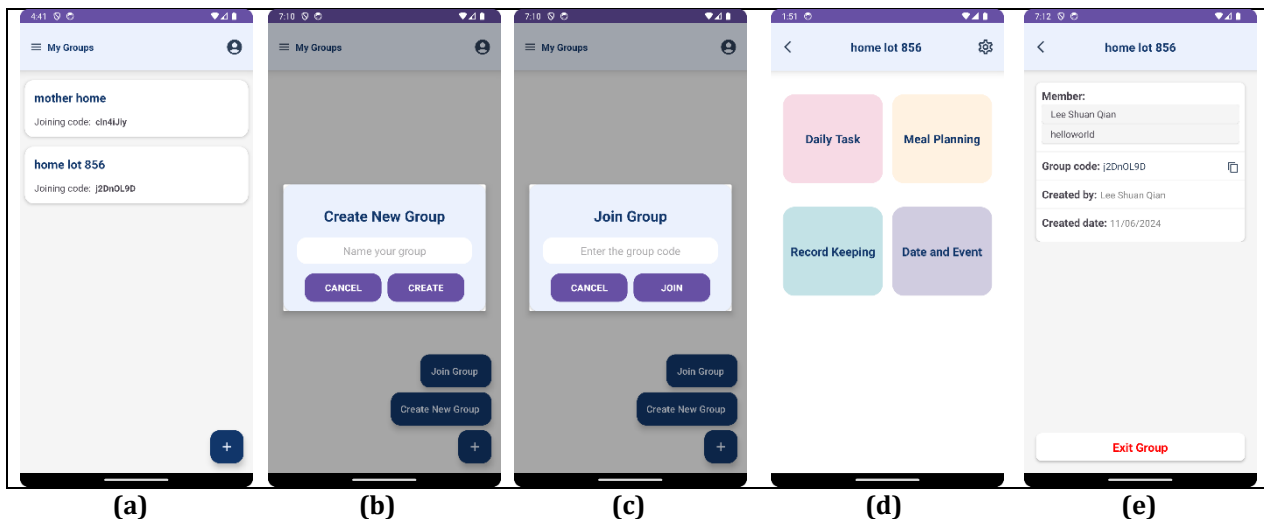
### 4.2.1 User Management Module

Fig. 6 show the user management interfaces of the application where authentication is required for users to access and use the application's features. New user without account need to sign up an account by input all of the fields including username, email address and password as depicted in Fig.6 (a). On the other hand, existing users should log in to their accounts using their email and password to access their account, as shown in Fig. 6 (b). Fig. (c) illustrates the user profile, displaying user details. Users can edit their profile, configure their account settings, and log out by clicking on the respective buttons.



**Fig. 6** User Management interfaces (a) Sign-up page; (b) Login page; (c) User profile page

Fig. 7 represents the group management interfaces of the application where it allows users to join multiple groups, each serving different purposes and offering a variety of features to enhance collaboration and organization. Fig. 7 (a) displays the groups the user has joined, each with a unique joining code that can be shared with others for joining. User may create new group or join a group using the joining code as shown in Fig. 7 (b) and (c). Fig 7 (d) illustrates the group dashboard where user can access the functionalities within the group. By clicking the upper right setting icon, user will be directed to the group details page as in Fig. 7 (e).



**Fig. 7** Group Management interfaces (a) Group page; (b) Create group alert dialog; (c) Join group alert dialog; (d) Group dashboard page; (e) Group detail page

### 4.2.2 Daily Task Module

Fig. 8 shows the Daily Task module interfaces where users can manage and organize their daily household task. Fig. 8 (a) depicts the pending and completed task lists. Users can click on the checkbox next to a pending task to

mark it as done. This feature helps users easily see which tasks are incomplete and prevents them from forgetting tasks. Users are able to add new task and view, update and delete the existing task as shown in Fig. 8 (b), (c) and (d). For task deletion, the application will display an alert dialog to confirm the action, preventing accidental deletions.

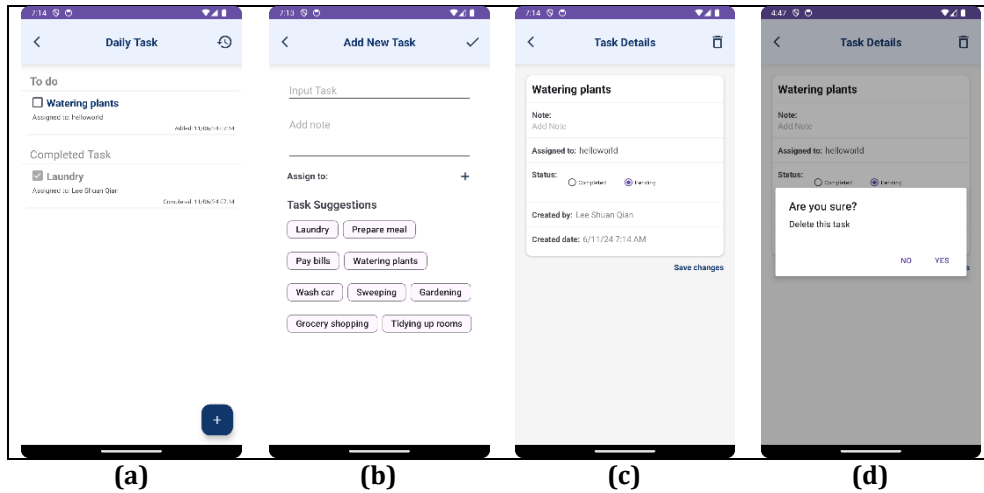


Fig. 8 Daily Task module interfaces (a) Daily Task page (a) Add task page; (c) Task detail page; (d) delete task alert dialog

### 4.2.3 Meal Planning Module

Fig. 9 and Fig. 10 demonstrate the meal planning module interfaces where users can organize their family meal plan. Fig. 9 (a) display a calendar view where a red indicator will appear on the date with planned meal. Users can click on the particular date to view, edit and delete the meal planning on the particular day. Fig. 9 (b) and (c) present the process of adding a new meal plan where user may click the upper right add icon to add a meal planning by select a plan date and the details of the planning. Besides, the application provides recipe suggestion for users as in Fig. 9 (d). Users can add new recipe, and view, edit and delete the existing recipe as depicted in Fig. 10. Additionally, users can add the recipe's ingredients to grocery shopping task or add recipe to meal planning as shown in Fig.10 (d) and (e).

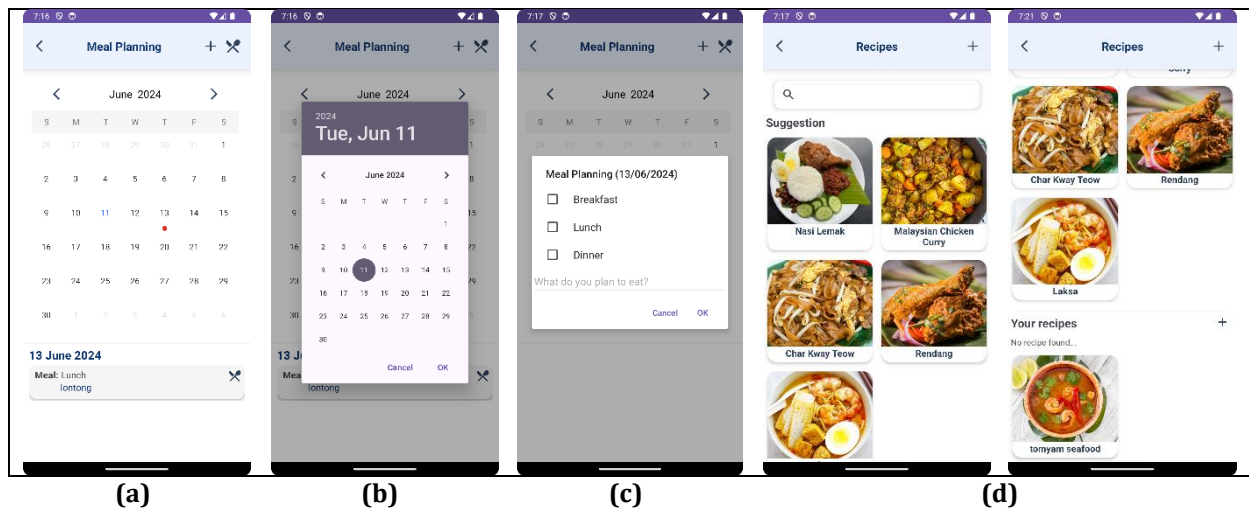
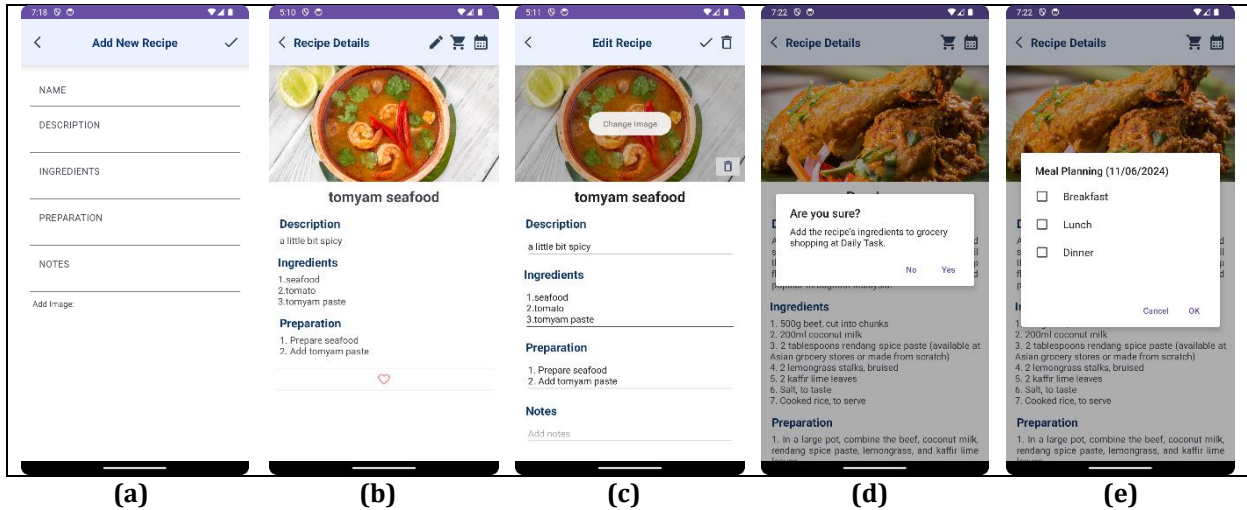


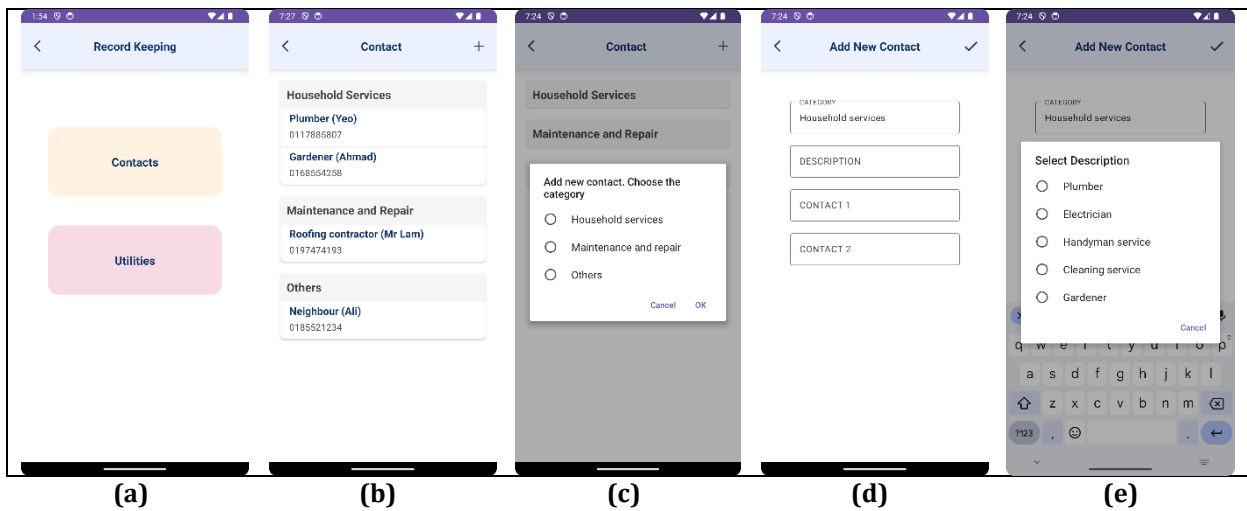
Fig. 9 Meal Planning module interfaces (a) Meal planning page (b) Choose plan date alert dialog (c) Add task alert dialog; (d) Recipe page



**Fig. 10** Meal planning module interfaces (a) Add recipe page (b) Recipe details page; (c) Edit recipe page; (d) Add recipe's ingredients to grocery shopping task; (e) Add recipe to meal planning

### 4.2.4 Record Keeping Module

Fig. 11 and Fig. 12 shows the interfaces of Record Keeping module, which allows users to store and manage useful contact and utility records. In Figure 8 (a), users are presented with the Record Keeping category page, where they can select specific categories to manage their records. Figure 8 (b) displays the contact record page, allowing users to view a list of existing contacts. Before filling in the contact details, as shown in Figure 8 (d), users must first choose the contact category through an alert dialog depicted in Figure 8 (c). Once the category is selected, users can proceed to enter the contact details, including the description and contact information. Figure 8 (e) illustrates the option to select a description from suggested options based on the chosen category. Furthermore, the utilities record as in Fig. 12 (a) allow user to record their monthly electric and water bills for quickly reference. The utilities record will be display in month calendar as in Fig. 12 (a). Users can add new utilities record as shown in Fig. 12 (b), view, edit and delete the utilities record as shown in Fig. 12 (c).



**Fig. 11** Record Keeping module interfaces (a) Record keeping category page (b) Contact record page; (c) Choose contact category; (d) Add contact page; (e) Add description from suggestion

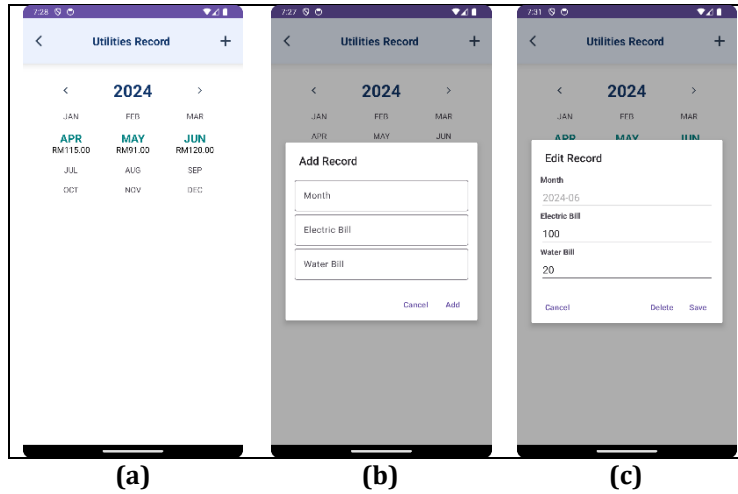


Fig. 12 Record Keeping module interfaces (a) Utilities record (a) Add record; (c) Record details;

### 4.2.5 Date and Event Module

Fig. 13 illustrates the Date and event module interfaces where users can organize their family event well. It displays a calendar view as shown in Fig. 13 (a) where a red indicator will appear on the date that had planned event. Users can add new event as shown in Fig 13 (b). Besides, user can click on the event on the particular date to view, update and delete the event as in Fig. 13 (c).

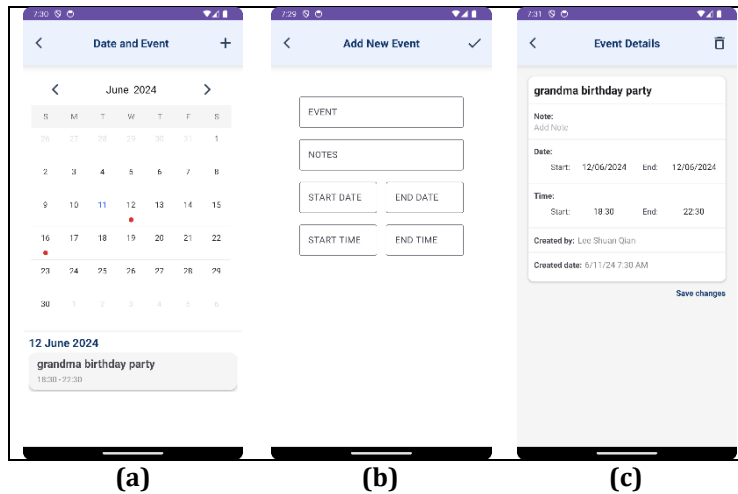


Fig. 13 Date and Event module interfaces (a) Date and event calendar view (a) Add event page; (c) Event details page;

## 4.3 System Testing

System testing is a level of software testing where a complete and integrated software system is tested. The purpose of system testing is to evaluate the system's compliance with the specified requirements. It ensures that the software behaves as expected in a production-like environment. Functional testing will be conducted in this project to evaluate the features and functionality of the proposed application.

### 4.3.1 Functional Testing

Functional testing involves validating a software system against its functional requirements. By providing suitable input and checking each function's output, functional tests ensure the software meets its specified requirements. This type of testing uses black-box techniques, ignoring the application's source code. Tables below illustrate the

functional test cases results of each module. Overall, all of the test case are pass and the functionality of the application works well as expected.

**Table 6** *Testing results of User Management Module*

No.	Test case	Expected Result	Actual Result	
1	Registration	Input name, email and password to register	Show message account created successfully and direct to main page.	PASS
		Input invalid data or empty fields	Show error message and require user to input data again.	PASS
2	Log in	Input correct email and password	Login successfully and access to the application.	PASS
		Input wrong email and password	Show error message and require user to input data again	PASS
3	Logout		Redirect to login page	PASS
4	Update profile	Update username at profile	Show update success message and display the updated name at profile	PASS
5	Join group	Input valid group code	Group joined successfully and display at the main page	PASS
		Input invalid group code	Show alert message "Invalid group code"	PASS
6	Create group	Input group name	Group created successfully and display at the main page	PASS
		Empty group name	Show alert message "group name is required"	PASS
7	Exit group		The main page updated; the exited group is not display	PASS

**Table 7** *Testing results of Daily Task Module*

No.	Test case	Expected Result	Actual Result
1	Add new task	Show task added successful message and display the task at the Daily Task page	PASS
2	Edit task	Display updated task details	PASS
3	Delete task	Show task deleted successful message and display updated task list without the deleted task	PASS
4	View task detail	Display the details of the particular task	PASS
5	Mark task as done	Display updated task list with the marked task at the completed task list	PASS

**Table 8** *Testing results of Meal Planning Module*

No.	Test case	Expected Result	Actual Result
1	Add meal plan	Show added successful message and display the it at the calendar in Meal Planning main page	PASS
2	Edit meal plan	Show update successfully and display updated meal plan details	PASS
3	Delete meal plan	Show deleted successful message and display updated meal plan at the calendar in Meal Planning main page without the deleted item	PASS
4	View meal plan detail	Display the details of the particular meal plan	PASS

**Table 8** Testing results of Meal Planning Module (continue)

No.	Test case	Expected Result	Actual Result
5	Add recipe	Show added successful message and display it at the recipe page at "Your recipe" section	PASS
6	Delete recipe	Show deleted successful message and display updated recipe page without the deleted one	PASS
7	Edit recipe	Show update successfully and display updated recipe details	PASS
8	View recipe details	Display the details of the particular recipe	PASS
9	Add recipe to meal plan	Show added successfully message and display at the calendar in Meal Planning main page	PASS
10	Add recipe's ingredients to grocery shopping task	Show added successfully message and display at the Daily Task page. The ingredient details will be imported to the description	PASS

**Table 9** Testing results of Record Keeping Module

No.	Test case	Expected Result	Actual Result
1	Add contact	Show added successful message and display the it at the contact main page	PASS
2	Edit contact	Show update successfully and display updated contact details	PASS
3	Delete contact	Show deleted successful message and display updated contact list at the contact page without the deleted item	PASS
4	View contact details	Display the details of the particular contact	PASS
5	Add utilities record	Show added successful message and display it at the utilities record page	PASS
6	Delete utilities record	Show deleted successful message and display updated utilities record without the deleted one	PASS
7	Edit utilities record	Show update successfully and display updated utilities record details	PASS
8	View utilities record	Display the details of the particular utilities record	PASS

**Table 10** Testing results of Date and Event Module

No.	Test case	Expected Result	Actual Result
1	Add event	Show added successful message and display the it at the module's main page	PASS
2	Edit event	Show update successfully and display updated event details	PASS
3	Delete event	Show deleted successful message and display updated event at the module's main page without the deleted item	PASS
4	View event details	Display the details of the particular event	PASS

## 5. Conclusion

In conclusion, the household management mobile application project has been successfully developed, achieving all its goals and objectives. This Android-based application marks a significant advancement in simplifying household organization by serving as a centralized hub for cataloging and structuring crucial household data, ultimately enhancing accessibility for all family members. The project underwent various crucial phases, including planning, analysis, design, implementation, and rigorous testing, ensuring its successful creation and functionality.

The application was designed to centralize household management, enhance task consistency, and promote family collaboration, successfully achieving these objectives. By consolidating scattered information into a single

platform, it improves efficiency and communication within the family. Its advantages include centralized data access, enhanced collaboration, and better organization of household tasks, leading to a more harmonious home environment.

However, the application does face limitations such as dependency on technology and concerns about privacy and security. To address these issues, recommendations include providing comprehensive user training, incorporating offline functionality, and enhancing security measures. These improvements will increase the application's reliability, usability, and user trust, ultimately better serving families in managing their household activities. Despite its limitations, the application has met its primary goals, with further enhancements suggested to optimize its effectiveness.

## Acknowledgement

The authors would like to thank the Faculty of Computer Science and Information Technology, Universiti Tun Hussein Onn Malaysia for its support.

## Conflict of Interest

Authors declare that there is no conflict of interests regarding the publication of the paper.

## Author Contribution

This journal requires that all authors take public responsibility for the content of the work submitted for review. The contributions of all authors must be described in the following manner:

*The authors confirm contribution to the paper as follows: **study conception and design:** Lee Shuan Qian, Firkhan Ali Hamid Ali; **data collection:** Lee Shuan Qian; **analysis and interpretation of results:** Lee Shuan Qian, Firkhan Ali Hamid Ali; **draft manuscript preparation:** Lee Shuan Qian, Firkhan Ali Hamid Ali. All authors reviewed the results and approved the final version of the manuscript.*

An author name can appear multiple times, and each author name must appear at least once. For single authors, use the following wording:

*The author confirms sole responsibility for the following: study conception and design, data collection, analysis and interpretation of results, and manuscript preparation.*

## References

- [1] Thomas, P. A., Liu, H., & Umberson, D. (2017). Family Relationships and Well-Being. *Innovation in Aging*, 1(3). <https://doi.org/10.1093/geroni/igx025>
- [2] Kapur, Radhika. (2021). Management of Household Responsibilities is Indispensable to Enhance one's Living Conditions.
- [3] Mulyaningsih, M., Eti Kusmiati, Ieke Sartika Iriany, & R. Pasciana. (2023). Smartphone Technology for Household Management. *Jurnal Organisasi Dan Manajemen*, 19(1). <https://doi.org/10.33830/jom.v19i1.2022.2023>
- [4] Moschin, M. (2023). Italian Lockdown and Consumption: The Rise of Flour during Forced Domestic Isolation from a Digital Perspective. *Gastronomy*, 1(1), 3–17. <https://doi.org/10.3390/gastronomy1010002>
- [5] Maduhu, Mary Shija., & Makori, Francis M. (2019). Food Waste Management in Coastal Hotels in Tanzania. *International Journal of Academic Research in Business and Social Sciences*. <http://doi.org/10.6007/ijarbss/v9-i11/6541>
- [6] "Homer | Helping You Be a Better Homeowner." <https://www.homer.co/>
- [7] Cozi Family Organizer | Must-Have App For Families," Cozi Family Organizer. <https://www.cozi.com/>
- [8] "FamilyWall | Happy Family Organization," Familywall.com. <https://www.familywall.com/index.html>

- [9] Rozhnova, T., Tomachynska, V., & Korsun, D. (2022). Life cycle models, principles and methodologies of software development. *InterConf*, 28(137). <https://doi.org/10.51582/interconf.19-20.12.2022.040>
- [10] Gbaranwi, Precious, B., Ojekudo, & Akpofure, N. (2021). A Comparative Analysis of Software Development Methodologies. *International Journal of Research and Innovation in Applied Science*, 06(05). <https://doi.org/10.51584/ijrias.2021.6513>
- [11] Ghahrai, A. (2017). What is Prototyping Model? In TechTarget.
- [12] Dennis, A., Wixom, B.H., & Roth R.M. (2012). System Analysis And Design(5<sup>th</sup> ed.). Hoboken, New Jersey John Wiley & Sons. Retrieved from [http://www.uoitc.edu.iq/images/informatics-institute/Competitive\\_exam/Systemanalysisanddesign.pdf](http://www.uoitc.edu.iq/images/informatics-institute/Competitive_exam/Systemanalysisanddesign.pdf)