APPLIED INFORMATION TECHNOLOGY AND COMPUTER SCIENCE
e-ISSN: 2773-5141

AITCS

Vol. 6 No. 2 (2025) 396-415
https://publisher.uthm.edu.my/periodicals/index.php/aitcs

Neighbourly: A Mobile Application for Uniting Local Engagements in Bandar Springhill

Puteri Fadlina Aliah Megat Mahayudin[1], Hanayanti Hafit[1*]

[1] *Fakulti Sains Komputer dan Teknologi Maklumat,*
Universiti Tun Hussein Onn Malaysia, Parit Raja, Batu Pahat, 86400, MALAYSIA

*Corresponding Author: hana@uthm.edu.my
DOI: https://doi.org/10.30880/aitcs.2025.06.02.022

Article Info

Received: 13 June 2025
Accepted: 3 November 2025
Available online: 30 November 2025

Keywords

Community engagement, Skill-sharing platforms, social cohesion, Local services, Digital ecosystems, Prototype development, Mobile application, Flutter, Firebase integration, Neighbourhood empowerment

Abstract

The Bandar Springhill community, located between Port Dickson and Seremban, traditionally relied on unorganised platforms such as Facebook and WhatsApp groups to access local skills and services. This often resulted in inefficiencies and led residents to seek external resources, overlooking the talents available within their neighbourhood. The Neighbourly platform is developed to address this issue by facilitating structured community interaction and resource sharing. Adopting the Prototype Model, the application is designed and implemented using Android Studio and Flutter for the front end, with Firebase integrated as the real-time backend solution. The prototype features a Login and Registration, User Profile, Skill Listings, Chat, Review and Rating, Volunteering Events, and an Admin Dashboard accessible through unique login credentials. The Neighbourly platform aims to empower Bandar Springhill residents by improving access to local resources, encouraging collaboration, and fostering self-reliance. Ultimately, it seeks to contribute to building a more cohesive, sustainable, and connected neighbourhood.

1. Introduction

The concept of being a neighbour has transformed significantly in today's digital society. Traditionally, neighbours were seen not only as people living nearby but also as members of a shared community, offering support and mutual assistance [1][2]. However, with the emergence of digital communication platforms, face-to-face interactions have declined, often being replaced by general-purpose tools such as Facebook and WhatsApp. While these platforms support general messaging and group sharing, they are not designed to facilitate structured, skill-based, and trust-driven relationships within neighbourhoods [3].

This gap is evident in the context of Bandar Springhill; a residential community located between Port Dickson and Seremban. The neighbourhood is home to many skilled individuals, ranging from bakers and electricians to tutors and crafters, but their talents often remain unrecognised and underutilised. Without a dedicated platform tailored for local engagement, many residents often seek external services rather than utilising resources within their community. This not only results in missed opportunities for affordable help but also weakens the community's sense of trust, identity, and connection [4].

This project introduces Neighbourly, a mobile application designed to encourage community-based skill sharing and address these challenges. Unlike general social platforms, Neighbourly offers a structured environment where residents can showcase their skills, request assistance, and participate in volunteering initiatives. The application aims to revitalise neighbourly ties by building a local ecosystem centred around collaboration, recognition, and mutual support [2][5].

This paper presents the theoretical foundation for community engagement through digital platforms and justifies the development of Neighbourly as a tailored, scalable, and inclusive solution for the Bandar Springhill community.

2. Related Work

Several digital platforms have been developed to support community-based interaction and skill sharing. This section reviews three relevant systems, which are BarterChain, Neighbor, and SkillSwap, and evaluates their features, strengths, and limitations to inform the design of Neighbourly.

2.1 Studying Similar Platforms

The community relies heavily on informal communication channels, such as Facebook pages and unofficial WhatsApp groups, to share information and stay connected. The Facebook page primarily promotes programs and community services but does not provide structured support for service providers to showcase their offerings. Meanwhile, widely used WhatsApp groups often lead to scattered and unorganised communication. Posts about services and needs quickly become buried in lengthy chat threads, causing delays in responses and missed opportunities for service providers and residents.

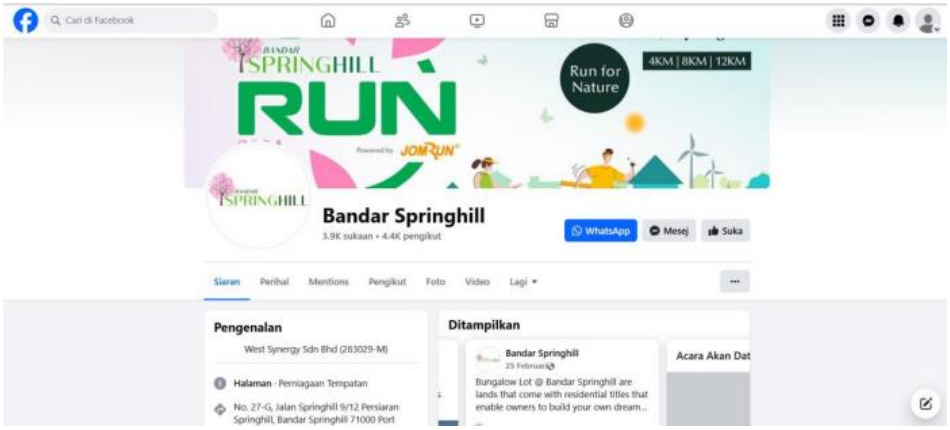


Fig. 1 *Bandar Springhill uses Facebook as a platform to spread announcements and engage with the community [6]*

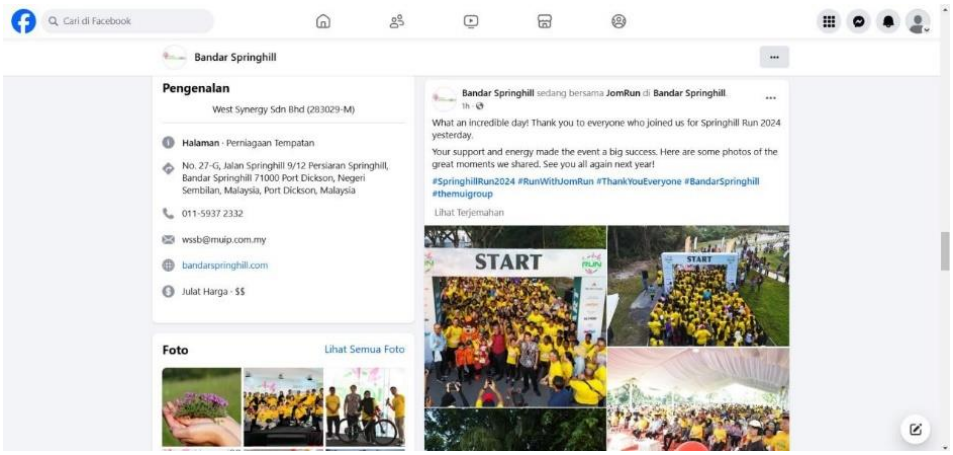


Fig. 2 *Bandar Springhill's Facebook shows updates from time to time about programs and advertisements [6]*

It is essential to examine existing platforms that facilitate local interaction and resource sharing to design an effective and context-aware solution. This study analyses three key systems BarterChain, Neighbor, and SkillSwap. Each offers valuable insights into community engagement, yet also presents limitations in scalability, usability, or localisation. Neighbourly seeks to address these gaps by combining its strengths into a unified, locally tailored platform that supports efficient and inclusive community collaboration.

2.1.1 BarterChain

BarterChain is a blockchain-based platform that facilitates peer-to-peer exchanges of skills and services, eliminating the need for monetary transactions. It utilises innovative contract technology to match users based on their available and required skills, fostering a cashless, trust-based community model [7]. While this approach encourages equitable participation, its reliance on reciprocal value exchange and the need for users to understand blockchain concepts can reduce accessibility for general users.

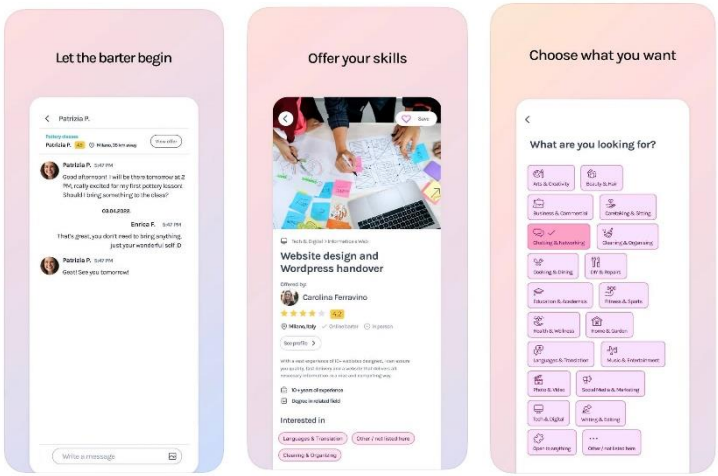


Fig. 3 *BarterChain Application Features [8]*

2.1.2 Neighbor

Neighbor is a platform that focuses on peer-to-peer storage rentals. It incorporates trust mechanisms such as verified user profiles, insurance-backed services, and a rating system to enhance transparency and safety [9]. Although its core function is limited to storage, the platform effectively utilises verification and feedback features to build user confidence. These mechanisms are applicable beyond storage and can inform the design of trust in broader service exchange platforms.

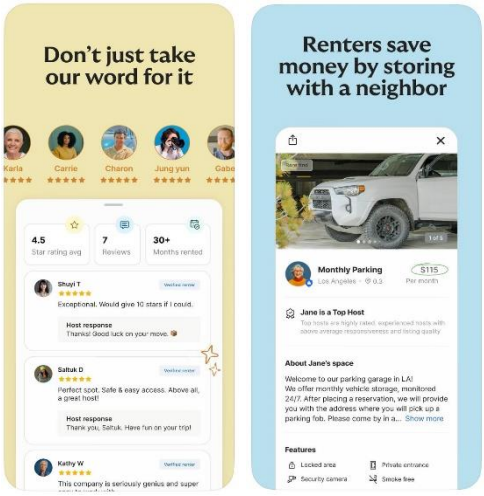


Fig. 4 *Neighbor Application Features [9]*

2.1.3 SkillSwap

SkillSwap enables users to exchange services and knowledge on a mutually beneficial basis, without involving financial transactions. It supports community participation and personal growth by encouraging collaborative learning and facilitating the sharing of expertise [10]. However, the platform's usability is limited by the need for exact skill matching, which may restrict participation when exchanges are not directly reciprocal.

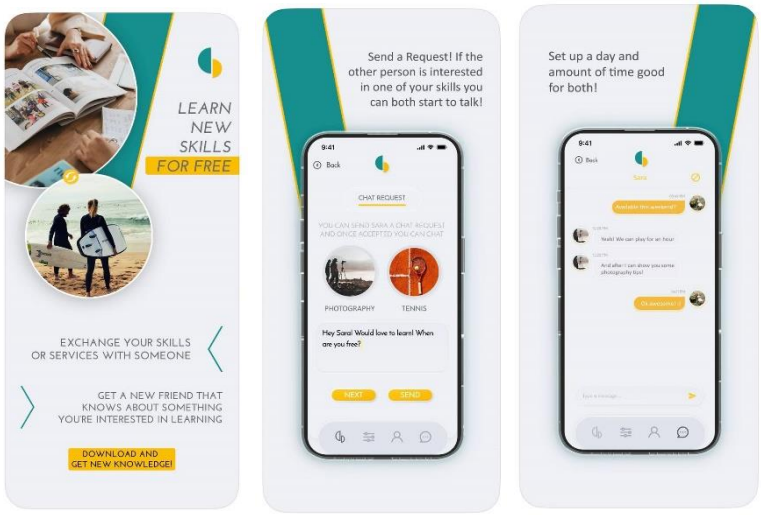


Fig. 5 *SkillSwap Application Features [10]*

2.1.4 Comparative Analysis

Neighbourly integrates selected strengths from these platforms while addressing their limitations. Table 1 provides a comparison of key features:

Table 1 *Comparison between Proposed Application and Related Work*

System/ Feature	BarterChain	Neighbour	SkillSwap	Neighbourly
User Verification	Blockchain identity (complex for general users)	Verified profiles	Basic or none	Manual approval & profile verification
Listings	Available (general)	Storage-specific listings	Skill-focused listings	Categorised & searchable service/skill listings
Matching Mechanism	Smart contract-based matching	Listing search & proximity-based discovery	Manual matching	Search filters, categories, and location-based discovery
In-App Communication	Limited (often external channels)	Limited (basic messaging)	Limited (basic messaging)	Real-time communication with integrated messaging
Rating & Review Mechanism	Not Available	Available	Available	Robust rating and review system to build trust
Community Focus	Decentralised global exchanges	General	Skill-Specific	Comprehensive focus, balancing general skill-specific offerings

3. Methodology

This study adopts the Prototype Model, an iterative development approach that enables early user feedback and continuous refinement. This model is suitable for community-based platforms where user needs evolve and require rapid adaptation. The process consists of five primary phases: requirement gathering, quick design, prototype development, prototype testing, and refinement.

3.1 Prototype Model

The Neighbourly platform addresses the challenges faced by the Bandar Springhill community in accessing local skills and services due to reliance on unorganised tools like WhatsApp and Facebook groups. This research employs the Prototype Model, a structured framework for iterative software development, to ensure that the system evolves dynamically in response to user feedback and community-specific needs [10]. The project integrates insights from previous chapters to foster inclusiveness, resource sharing, and collaboration among residents.

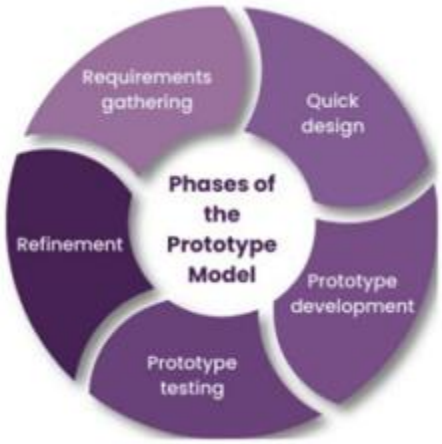


Fig. 6 *The phases of the Prototype Model [11]*

This diagram illustrates the cyclical nature of the Prototype Model, where requirements are gathered and quickly translated into a design and working prototype. Feedback from testing informs refinement before the cycle repeats or progresses to implementation.

Table 2 *Summary of Key Tasks and Outcomes by Phase*

Phase	Key Activities	Outcomes
Requirements Gathering	Interviewed a community resident and observed digital behaviours; created UML diagrams.	Identified functional (e.g., skill listings) and non-functional requirements; module list finalised.
Quick Design	Created mock-ups and wireframes using Figma, also outlined UI structure.	Produced blueprints for Login, Skill Listings, Profile, Admin Dashboard, and Event screens.
Prototype Development	Developed Login, Registration, Skill Listings, Chat, Review, Admin, and Volunteering Event modules using Flutter & Firebase.	First working version of Neighbourly mobile app with Firebase backend.
Prototype Testing	Conducted pilot testing with 10 users from Bandar Springhill; gathered feedback via surveys and observation.	Collected user feedback highlighting issues in navigation, categorisation, and filtering.
Refinement	Incorporated UAT feedback; added skill categorisation, enhanced UI layout	Finalised a stable, user-informed prototype ready for further testing or deployment.

4. Requirement Analysis

The requirement analysis phase is essential for determining what the system must do (functional requirements) and how it must perform (non-functional requirements). This section outlines the core functionality and performance expectations for the Neighbourly platform, informed by community interviews, behavioural analysis, and real-world use cases observed among Bandar Springhill residents.

4.1 Functional Requirement Analysis

The Neighbourly platform's functional requirements encompass the basic capabilities necessary to ensure the achievement of both user and system goals. This platform enables users to easily register, view, and engage with other service listings while maintaining genuine interactions. Table 3 outlines the functional requirements:

Table 3 *Functional Requirement Analysis*

Functional Requirement	Description
1. User Account Management	Users shall register and log in using their email and password. Authentication is validated through Firebase.
2. Admin Authentication	Admins shall log in using predefined credentials that grant elevated privileges and access to events and user moderation tools.
3. Skill Listing Creation	Users shall be able to create, update, and delete skill listings, including descriptions, categories, and availability.
4. Service Request System	Users shall be able to initiate service requests, view responses, and communicate with service providers via an integrated chat.
5. Profile Management	Users shall manage their profiles, including contact information, display name, profile photo, and skill categories.
6. Ratings and Reviews	After service completion, users are required to leave a star rating and provide optional written feedback, which will be visible to other users.
7. Volunteering Participation	Users shall browse upcoming events, register to join, and cancel registration if needed.
8. Admin Posting Control	Admins should create and manage community announcements, events, and user reports within a secure mobile admin interface.

4.2 Non-Functional Requirements

The non-functional requirements describe the other characteristics that make the platform good, reliable, and easy to use. Table 4 outlines these requirements:

Table 4 *Non-Functional Requirement Analysis*

Non-Functional Requirement	Description
Platform Portability	The application should be accessible on all Android devices, with responsive and adaptable screen functionality.
Real-Time Interaction	Firebase backends support real-time synchronisation for chat, listing updates, and reviews.
Usability and Accessibility	UI shall be user-friendly and straightforward, using accessible fonts, sizes, and contrast suitable for users aged 18–60+.
Performance Efficiency	The app shall load its core interface in under 3 seconds on mid-range devices.

Security and Privacy	Authentication shall use secure Firebase protocols. User data shall be encrypted and not publicly accessible.
Maintainability	Codebase shall use a modular structure to allow easy updates and future enhancements.

4.3 Unified Modelling Language (UML)

There are two major categories of UML diagrams: behavioural diagrams and structural diagrams. Behavioural diagrams use case, sequence, and activity diagrams, while structural diagrams are class diagrams.

4.3.1 Use Case Diagram

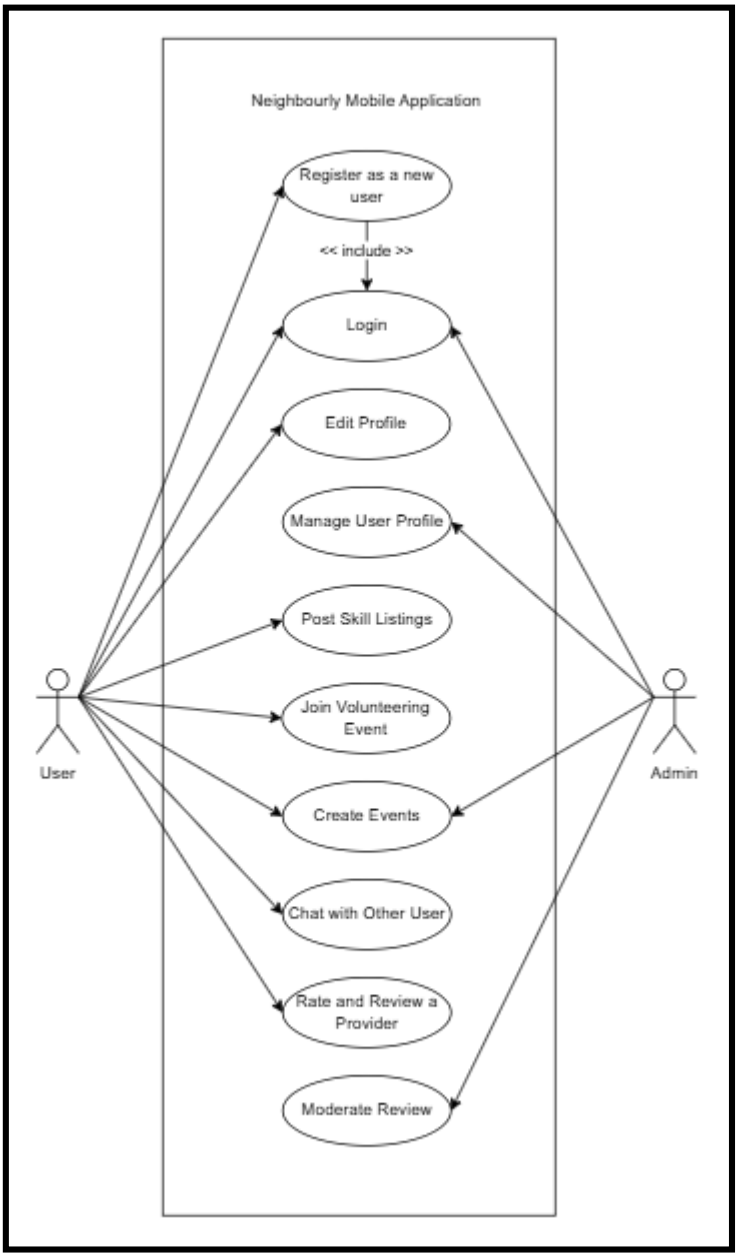


Fig. 7 *Neighbourly Use Case Diagram for User and Admin*

Applied Information Technology and Computer Science Vol. 6 No. 2 (2025) p. 396-415

The Use Case Diagram for Neighbourly outlines the primary interactions between the user and the system, identifying the core functionalities the application offers to its users. It highlights the roles of two leading actors: Users and Admins. Users can register, log in, manage their profiles, browse and create skill listings, engage in chat, participate in volunteering events, and submit ratings and reviews. Each use case represents a specific goal that users aim to achieve through the application, ensuring that all essential community-based interactions are captured and accounted for.

On the other hand, Admins have additional privileges, such as logging in with administrative access, managing community events, and moderating user activities. The diagram helps in visualising system boundaries and clarifying the responsibilities of each actor. It also supports the validation of functional requirements by ensuring that all intended user interactions, such as service browsing, communication, and profile editing, are accounted for in the early design phase. Overall, the Use Case Diagram serves as a foundational reference for guiding the system's development and ensuring alignment with user expectations.

4.3.2 Sequence Diagram

These diagrams illustrate how different system components communicate to accomplish specific tasks for users and administrators. Each diagram highlights the order of operations, the method calls involved, and the system's response, ensuring a clearer understanding of the process flow and backend integration logic.

4.3.2.1 Sequence Diagram for User Registration

This diagram models the interaction between the User, the Registration Page, and Firebase services. When a user registers, the system captures input data (e.g., email, password, name), validates the input through Firebase Authentication, and stores additional profile data in Cloud Firestore. Upon successful registration, the user is redirected to the dashboard. This diagram reflects the authentication and onboarding flow for new users.

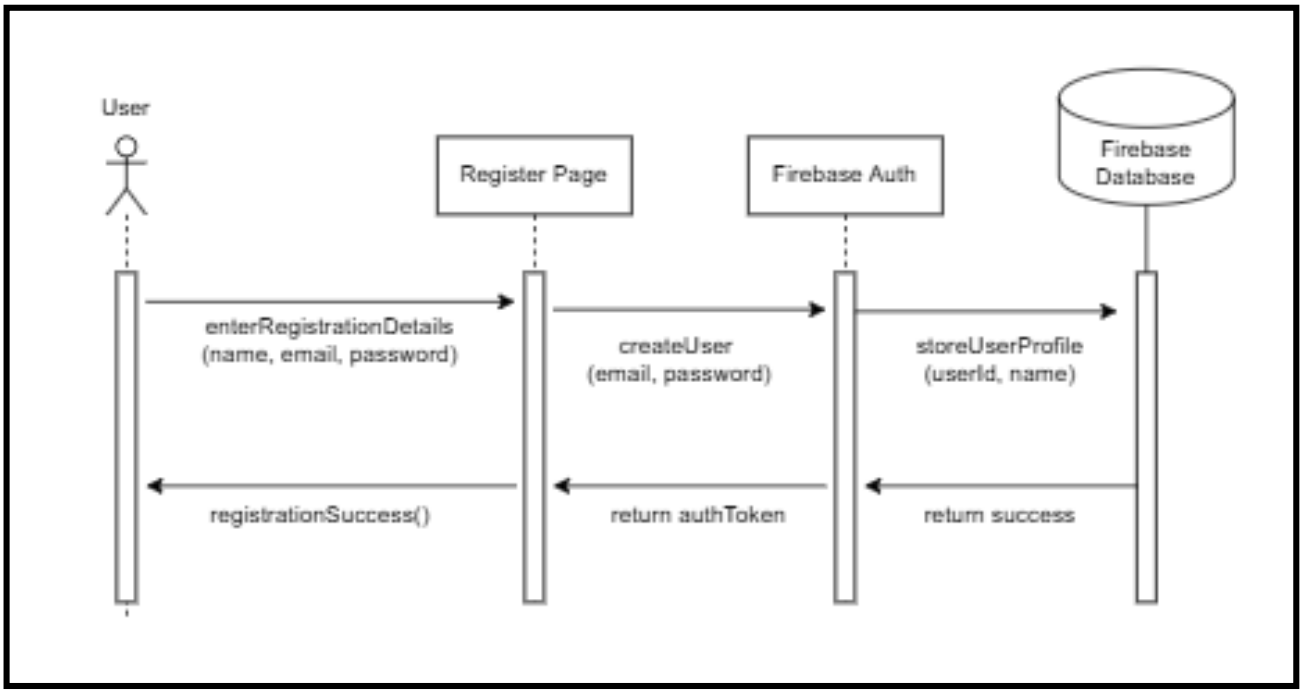


Fig. 8 *Sequence Diagram for User Registration*

4.3.2.2 Sequence Diagram for Admin Posting a Volunteering Event

This diagram illustrates the admin flow where a user with exceptional credentials logs in, accesses the event form, and submits a volunteering event. After authentication via Firebase Auth, the EventForm collects the details and stores them in the Firebase Database. Once saved, the event feed is refreshed for users. This diagram illustrates the extended privileges of admin users, highlighting event moderation and announcement broadcasting functionality.

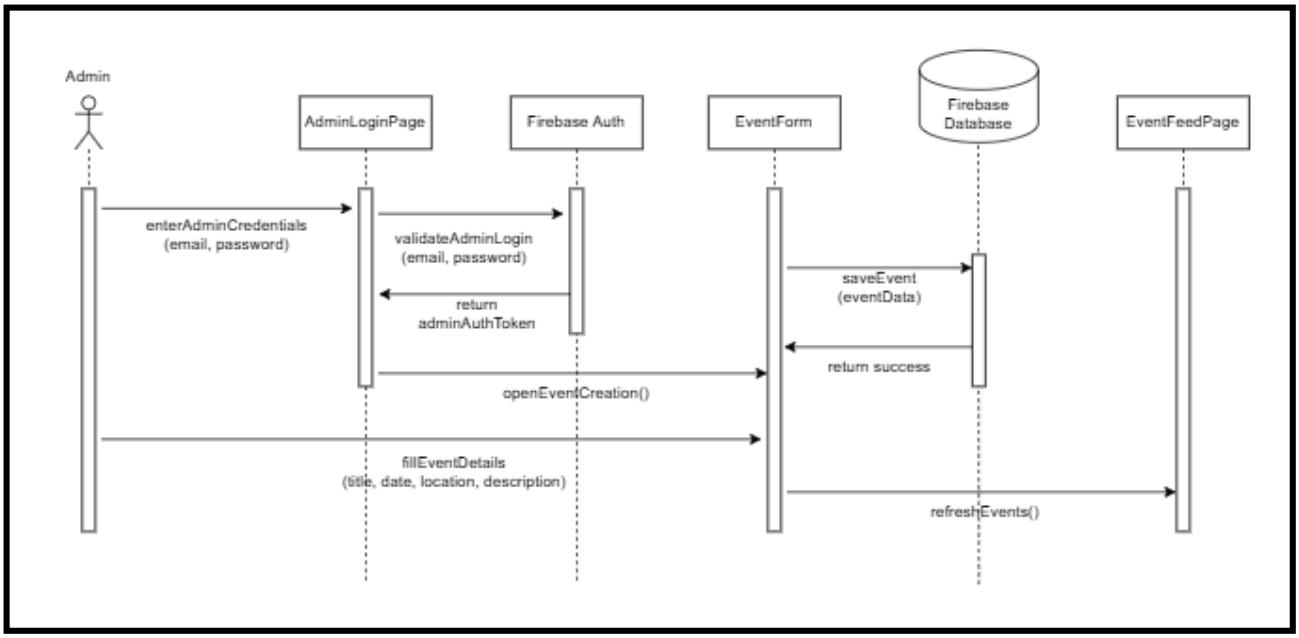


Fig. 9 *Sequence Diagram for Skill Listing Submission*

4.3.3 Activity Diagram

4.3.3.1 User Activity Diagram

The User Activity Diagram for the Neighbourly application illustrates the typical flow of actions a user can perform, from login to logout. The process begins with the user either registering or logging into the system by entering their credentials. If the credentials are invalid, an error message is displayed, and the user is prompted to retry. Once authenticated, users are directed to the main menu, which serves as the central navigation point for accessing various features of the app, including skill listings, volunteering opportunities, chat functionality, and profile management.

From the main menu, users can create or search for skill listings, initiate chats with other residents, participate in community events, and manage their profiles. Each of these features follows a clear and structured path, offering options to submit reviews, communicate in real-time, or update user details. At the end of each activity, users are prompted to either return to the main menu or log out. The complete User Activity Diagram is presented in Appendix A to accommodate its detailed structure and layout.

4.3.3.2 Admin Activity Diagram

The Admin Activity Diagram illustrates the sequence of actions available to administrators within the Neighbourly mobile application. The process begins with the admin logging in using authorised credentials, after which they are granted access to the admin dashboard. From this central interface, admins can perform several key tasks to support the platform's operation and community engagement. These tasks include managing user-generated content, such as skill listings and event postings; moderating user behaviour; reviewing reports or flagged content; and posting official announcements or volunteering opportunities. Each activity is designed to help maintain the platform's safety, relevance, and trustworthiness for all users.

The diagram also shows decision points where the admin may choose to continue managing other modules or exiting the system. After completing their tasks, admins are prompted to either return to the dashboard or log out of the system. This activity flow ensures that administrative tasks are streamlined and clearly defined, reducing ambiguity and improving system governance. Overall, the Admin Activity Diagram supports the enforcement of platform policies and the smooth operation of Neighbourly's backend processes. For visual reference, the complete diagram is included in Appendix B of this document.

Applied Information Technology and Computer Science Vol. 6 No. 2 (2025) p. 396-415

4.3.4 Class Diagram

The class diagram for the "Neighbourly" app illustrates the relationships between key modules, such as Admin Dashboard, Login & Registration, User Profile, Skill Listings, Service Exchange, Volunteering Events, and Review & Rating. The AdminDashboard class interacts with UserProfile, Skill, ServiceRequest, and Event, as the admin manages users, services, skills, and events. The UserAuth class links to the UserProfile class, handling user login and authentication. The UserProfile class interacts with Skill, ServiceRequest, Event, and UserRating, allowing users to create and manage their profiles, skills, services, events, and ratings. Additionally, Skill interacts with UserProfile to manage skill listings, while ServiceRequest and Event are both linked to UserProfile for service requests and event registration.

Furthermore, the ServiceRequest and UserRating classes form critical parts of the service exchange and review system, with ServiceRequest allowing users to initiate, approve, and track service requests, and UserRating enabling users to rate both skills and services. Skill and User Rating are connected, as skills can be rated by users, which influences their visibility and appeal. These interactions are illustrated in the diagram using appropriate arrows to indicate dependencies and associations. Solid lines represent associations, while hollow diamonds denote composition, and dashed lines indicate dependencies. This structure ensures that all classes and modules in the app work together efficiently to manage user interactions, services, and feedback.

This class diagram will be included in Appendix C of the report.

4.4 User Interface Design

The UI for the *Neighbourly* app is designed to be simple, clean, and community focused. It features a calming green theme that reflects harmony and trust among neighbours. The login and registration screens are straightforward, with clear input fields for user details and intuitive call-to-action buttons. The home page welcomes the user by name, features a search bar for easy navigation, and includes a bottom navigation bar with icons for Home, Listings, Join, and Profile. This layout ensures ease of use for users of all ages and tech backgrounds, promoting engagement within the community.

Fig. 10 *The UI design for Login Page, Register Page, and User Home Page.*

The second set of screens in the Neighbourly app showcases the main features that support community interaction. The first screen displays a list of nearby users offering various skills and services, such as cleaning, baking, and plumbing, with an option to add new listings. The second screen highlights local events and volunteering opportunities organised by different groups, encouraging users to participate in community activities. The third screen is the user profile page, which displays the user's name, contact information, and profile picture, along with options to edit the profile, change the password, and toggle between light and dark mode. The interface maintains a consistent green theme and uses a bottom navigation bar for easy access across all sections.

Fig. 11 *The UI design for Skill Listings Page, Events & Volunteering Page, and User Profile Page.*

The final UI screens display the review and rating page alongside the chat page. The review and rating page enables users to view and provide feedback on services or events, fostering trust and accountability within the community. Users can rate experiences using a star system and add written comments. The chat page provides a direct messaging feature, enabling users to communicate in real-time for coordination or inquiries. Both screens follow the app's clean and consistent design, with a green color scheme and intuitive layout to ensure a smooth user experience.

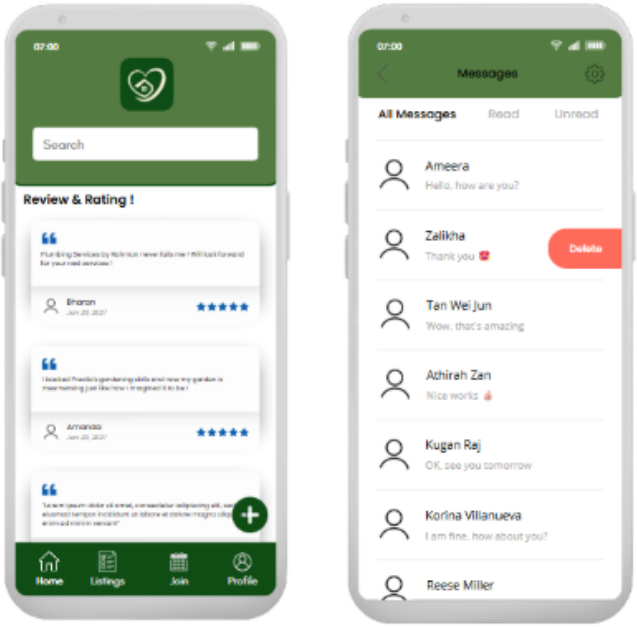


Fig. 12 *The UI design for Review & Rating Page and Chat Page.*

Applied Information Technology and Computer Science Vol. 6 No. 2 (2025) p. 396-415

5. Implementation and Testing

This section will discuss the implementation and testing of the Neighbourly mobile application.

5.1 Implementation

The Neighbourly mobile application was developed using Flutter for the frontend UI and Firebase for backend services, including authentication, real-time data storage, and role-based access control. The final deployment was made available on Android devices and tested by selected users from Bandar Springhill.

Following internal testing and refinement, a final prototype was launched. A user guide was prepared to assist first-time users, and Firebase Analytics was integrated to monitor user activity, identify errors, and evaluate feature engagement.

5.1.1 Login and Registration Module

The image displays two screens from the "Neighbourly" mobile application. The first screen is the login page, where existing users are prompted to enter their email and password to access the app. Below the login fields, a "Login" button is available, allowing users to proceed. Additionally, a link is provided for users who do not have an account, directing them to the registration page.

The second screen is for creating a new account. It includes fields for the user's full name, email, phone number, and password. After entering the required information, users can click the "Register" button to create their account. A link is also available for those who already have an account, taking them back to the login page. The overall design uses a gradient background of green and pink, providing a warm and welcoming atmosphere.

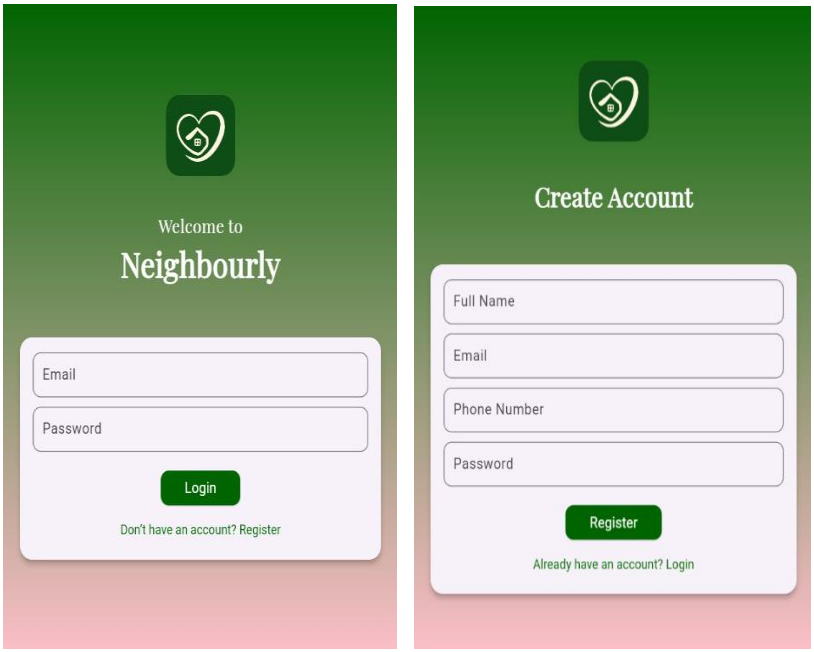


Fig. 13 *The Login Page and Register Page.*

5.1.2 User Profile Module

The image showcases the "User Profile Page" of the "Neighbourly" mobile application, which includes two features: Edit Profile and Change Password. On the left side, the profile page displays the user's name, "Puteri Hana", along with their phone number. The user can access options to Edit Their Profile and Change Their Password. There is also a Sign Out button to log out of the account. The Edit Profile feature enables users to modify their name and phone number, with a "Save" button to confirm the changes. On the right side, the Change Password page allows users to update their password by entering their current password, a new password, and confirming the new password. A "Change Password" button is provided to save the changes.

The interface is clean and user-friendly, offering a straightforward way for users to manage their personal information and security settings. The layout is designed to offer simple navigation between the profile, edit, and password change features, enhancing the user experience.

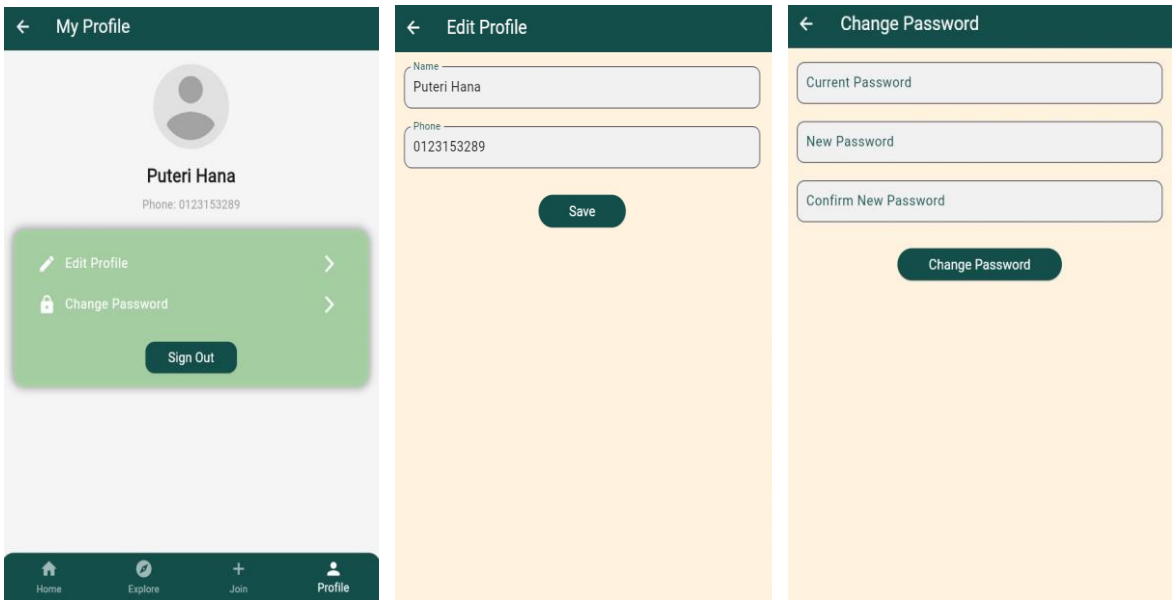


Fig. 14 *The User Profile Page, including Edit Profile and Change Password features.*

5.1.3 Skill Listings Module

This module allows users to offer services to the community by creating skill listings. Each listing includes the user's name, skill category, and a brief description. Users can add new listings, view more details, or delete their posts. The layout emphasizes clarity and accessibility.

5.1.4 Chat Module

The Chat module supports real-time messaging between users after a service request has been made. It uses Firebase for instant updates, allowing smooth and timely communication. Messages are clearly distinguished between sender and receiver, and users are identified by their profile names.

5.1.5 Review and Rating Module

After a service is completed, users can submit feedback through this module. It includes a star rating (1–5) and an optional written comment. Only users who have interacted with a service can leave a review, helping build trust and accountability in the community.

5.1.6 Volunteering Events Module

This module lists upcoming community events and volunteering opportunities. While admins can publish official events, users can also offer their programs for others to join. Each event includes details such as title, date, location, and description. Users can register or unregister for events, encouraging real-world participation. Attendance is tracked to support engagement and accountability.

5.1.7 Admin Dashboard Module

Accessible only to admin users, this module allows monitoring and management of platform activities. Admins can oversee users, approve or remove listings, and create events or announcements. It ensures the platform remains well-regulated and aligned with community standards.

5.2 Functional Testing

A series of black-box functional test cases was conducted to ensure that features performed as intended. These test cases were derived directly from the functional requirements defined in Section 4.1.

Table 5 *Functional Testing Results by Module*

Module	Test Cases	Expected Output	Actual Output
Login & Registration	Register, log in, reset password.	New user created, login works, password reset email received	Pass
Admin Dashboard	Admin login, event creation, and user moderation	Admin accesses dashboard, posts announcements, and manages users	Pass
Skill Listings	Create, update, and delete skill listings.	Listings appear or update in the feed accordingly.	Pass
Service Request & Chat	Initiate service requests, exchange messages	Request sent successfully, messages shown in real time	Pass
Profile Management	Edit profile info, update photo, and skill categories	Profile updated, skill tags saved	Pass
Review & Rating	Leave a review after the service.	Review appears on the user's listing.	Pass
Volunteering Events	View events, register/unregister	Registration recorded, cancellation confirmed	Pass

All essential modules passed functionality tests, though minor layout bugs and text overflow issues were resolved.

5.3 User Acceptance Testing (UAT)

A User Acceptance Testing (UAT) session was conducted with ten participants from the Bandar Springhill community to evaluate the usability and effectiveness of the Neighbourly application. These individuals represented diverse demographics, including age, occupation, and familiarity with community-based applications.

Participants interacted with core modules, including login and registration, profile management, skill listing, service request via chat, review system, and volunteer registration. Feedback was gathered using a structured Google Form that included both closed-ended and open-ended questions. Quantitative results were analysed to assess ease of use, navigation, and satisfaction, while qualitative responses were used to identify potential areas for improvement.

5.3.1 Quantitative Findings

Overall, participants responded positively to the usability of the Neighbourly application. The results can be summarised as follows:

- The average score for registration ease was 4.2 out of 5, indicating that most users found the account setup process smooth and intuitive.
- The navigation experience scored an average of 4.0 out of 5. While most users navigated the system effectively, two respondents experienced minor issues related to button responsiveness and text readability.
- The overall satisfaction rating averaged 4.3 out of 5, demonstrating a high level of user satisfaction. Eight out of ten participants rated the system 4 or 5 stars.
- Regarding platform adoption, 80% of users confirmed they would recommend the application to others in their community.

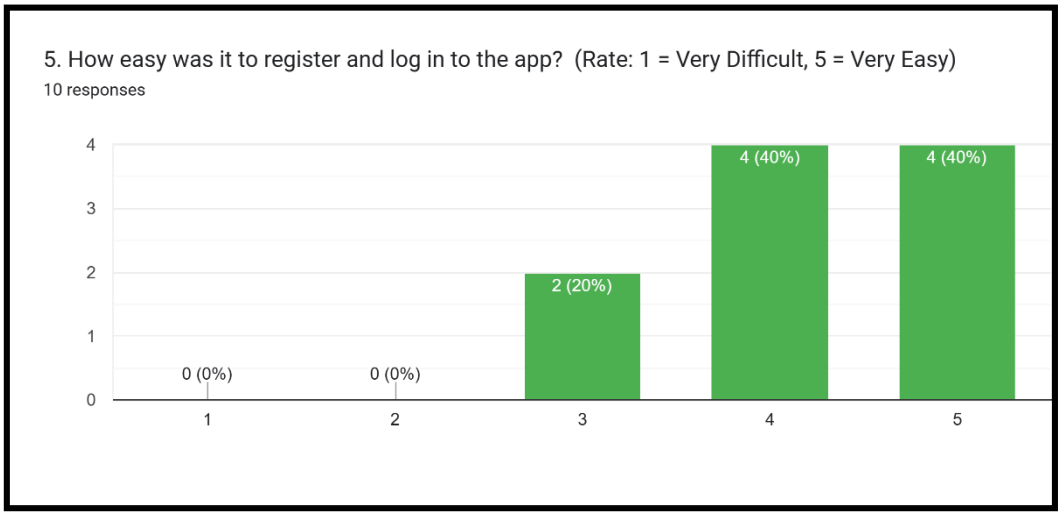


Fig. 15 *The result for user registration experience.*

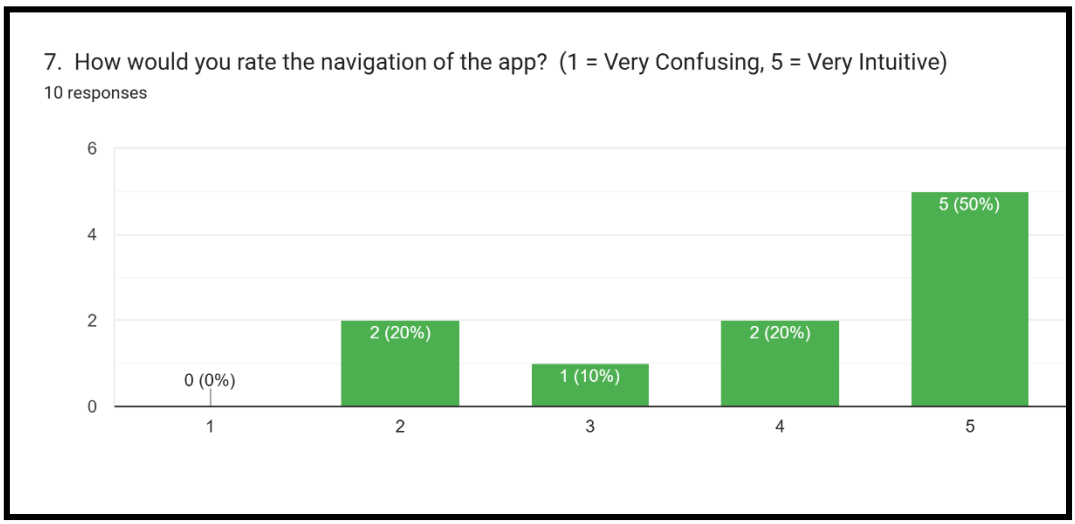


Fig. 16 *The result for navigation experience.*

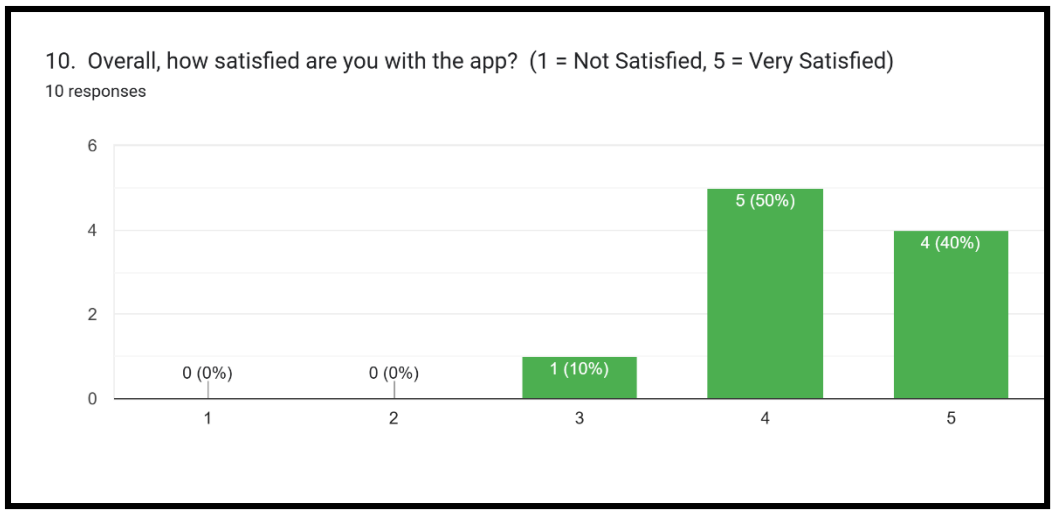


Fig. 17 *The user satisfaction result.*

5.3.2 Qualitative Findings

Participants provided open-ended responses to reflect their impressions and offer suggestions for improvement. The most frequently mentioned positive aspects of the platform were:

- A clean and user-friendly interface
- Clear and accessible skill listing features
- Empowerment of local service providers to promote their offerings

In terms of suggestions for improvement, users recommended:

- Implementing a dark mode for accessibility and aesthetic preference
- Introducing advanced filtering and search capabilities for listings
- Improving button responsiveness and UI layout on lower-end devices
- Enabling richer chat functionalities such as media sharing and reading indicators

5.3.3 Summary

The results of the UAT affirm that the Neighbourly application is functionally robust and well-received by the intended user base. The feedback obtained has been instrumental in guiding final refinements to the user interface and interaction flow. Remaining issues, such as the addition of media support and notification features, are addressed in the recommendations in Chapter 6.

6. Conclusion

The Neighbourly platform is developed to enhance community participation and skill sharing within the Bandar Springhill area. The system was successfully implemented and evaluated through real-user feedback, achieving its intended objectives. The development process followed the Prototype Model, ensuring continuous refinement based on user needs and local context.

6.1 Achievement of the Overall Objectives

The Neighbourly project successfully achieved its primary objectives. First, community needs were identified through interviews and observations of digital interactions on platforms such as WhatsApp and Facebook. Second, a functional mobile application was developed that integrated seven essential modules, each addressing a specific aspect of local skill-sharing and interaction. Third, the system was validated through user testing and feedback, which confirmed its effectiveness in promoting hyperlocal resource sharing and neighbourly engagement.

6.2 System Advantages

The platform provides a structured, accessible alternative to unorganised communication tools commonly used in neighbourhoods. It enhances local visibility of skills, simplifies service requests, and fosters community collaboration. By incorporating a real-time chat system and a public review mechanism, the platform fosters trust and accountability among users. Admin users benefit from streamlined event posting and content moderation tools embedded within the same application interface.

6.3 System Disadvantages

Despite its successful implementation, the system has a few limitations. Push notifications for new events or messages are not yet implemented, limiting user engagement. Security is currently limited to standard Firebase authentication, without multi-factor authentication. Additionally, the admin dashboard lacks multimedia support for posts, which limits the expressiveness of content.

6.4 Recommendations

It is recommended to integrate Firebase Cloud Messaging for real-time notifications to enhance user experience and scalability. Implementing multi-factor authentication would enhance account security, particularly for administrative users. The admin dashboard should also be expanded to support image uploads and formatted content. Optimisations should be made for older devices, and improvements in chat functionality, such as image sharing and message status indicators, would provide users with a more complete and interactive experience.

Acknowledgement

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

Conflict of Interest

The authors declare that there is no conflict of interest regarding the publication of this 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:*** *Puteri Fadlina Aliah Binti Megat Mahayudin, Hanayanti Binti Hafit;* ***data collection:*** *Puteri Fadlina Aliah Binti Megat Mahayudin;* ***analysis and interpretation of results:*** *Puteri Fadlina Aliah Binti Megat Mahayudin, Hanayanti Binti Hafit;* ***draft manuscript preparation:*** *Puteri Fadlina Aliah Binti Megat Mahayudin, Hanayanti Binti Hafit. All authors reviewed the results and approved the final version of the manuscript.*

References

[1] R. Williams, *Neighbourhoods and Community: A Critique of Theories of the Local*, 2nd ed. Oxford, U.K.: Oxford University Press, 2005.

[2] R. D. Putnam, *Bowling Alone: The Collapse and Revival of American Community*. New York, NY, USA: Simon & Schuster, 2000.

[3] D. Boyd and N. B. Ellison, "Social network sites: Definition, history, and scholarship," *J. Comput.-Mediat. Commun.*, vol. 13, no. 1, pp. 210–230, 2007. [Online]. Available: https://doi.org/10.1111/j.1083-6101.2007.00393.x

[4] D. W. McMillan and D. M. Chavis, "Sense of community: A definition and theory," *J. Community Psychol.*, vol. 14, no. 1, pp. 6–23, 1986. [Online]. Available: https://doi.org/10.1002/1520-6629(198601)14:1<6::AID-JCOP2290140102>3.0.CO;2-I

[5] P. Parigi and R. Henson, "Neighborhoods in the digital age: Social networks and community engagement," *J. Urban Affairs*, vol. 36, no. 2, pp. 219–232, 2014. [Online]. Available: https://doi.org/10.1111/juaf.12035

[6] Bandar Springhill Seremban, *Bandar Springhill Seremban [Facebook page]*. Facebook. [Online]. Available: https://www.facebook.com/bandarspringhillseremban [Accessed: Dec. 9, 2024].

[7] F6S, "BarterChain: The platform for trading skills and services," 2022. [Online]. Available: https://www.f6s.com/barterchain [Accessed: Dec. 10, 2024].

[8] BarterChain, *BarterChain Appstore*. [Online].

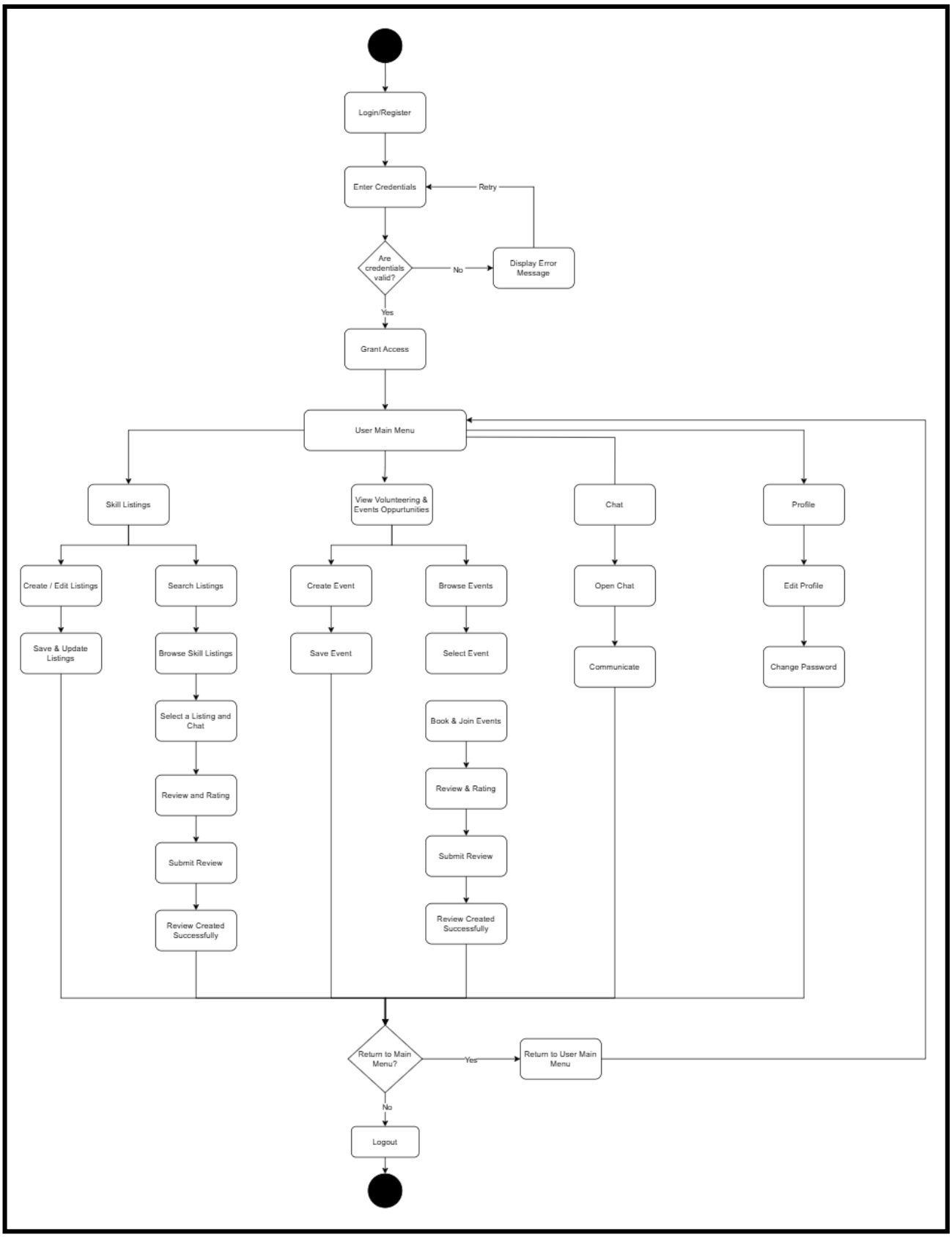
[9] Neighbor, *Neighbor Storage App on the App Store*. Apple. [Online]. Available: https://apps.apple.com/us/app/neighbor-storage/id1438651538 [Accessed: Dec. 13, 2024].

[10] SkillSwap, "SkillSwap: Exchange skills and services locally." [Online]. Available: https://www.skillswap.org/ [Accessed: Jun. 10, 2025].

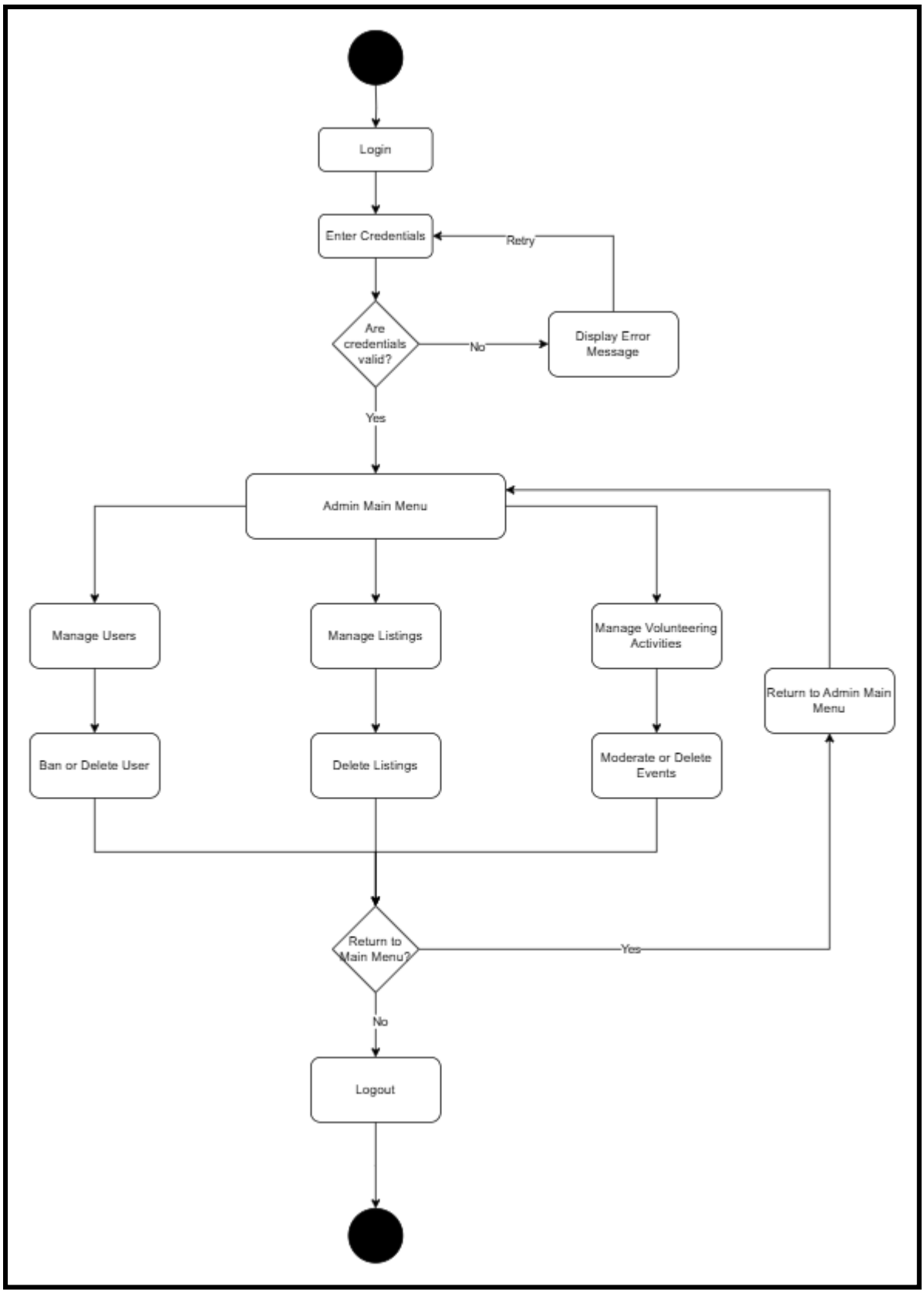
[11] The Knowledge Academy, "Prototype model in software engineering." [Online]. Available: https://www.theknowledgeacademy.com/blog/prototype-model-in-software-engineering/ [Accessed: Jun. 10, 2025].

Applied Information Technology and Computer Science Vol. 6 No. 2 (2025) p. 396-415

Appendix A: User Activity Diagram for Neighbourly



Appendix B: Admin Activity Diagram for Neighbourly



Applied Information Technology and Computer Science Vol. 6 No. 2 (2025) p. 396-415

Appendix C: Class Diagram for Neighbourly