**MOUNT KENYA UNIVERSITY**

**SCHOOL OF PURE AND APPLIED SCIENCES**

**DEPARTMENT OF INFORMATION TECHNOLOGY**

**DAILY CHRISTIAN DEVOTION MOBILE APPLICATION**

**A case Study of Christian Fellowship Groups**

**BY: BENITE BUYANA**

**REG. NO.: BBICT/2023/40703**

**A Research Project Submitted in Partial Fulfilment of Requirement for the Mount Kenya University Award of**

**BACHELOR OF BUSINESS INFORMATION COMMUNICATION TECHNOLOGY**

**JULY 2025  
  
  
  
  
  
  
DECLARATION**

I hereby declare that this project report is based on my original work except for citations and quotations, which have been duly acknowledged. I also declare that it has not been previously and concurrently submitted for any other degree or award at Mount Kenya University.

Name: Benite BUYANA

REG No.: BBICT/2023/40703

Signature: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_Date: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

SUPERVISOR

I, the undersigned, do hereby certify that this is a true report for the project undertaken by the above-named student under my supervision and that it has been submitted to Mount Kenya University with my approval.   
Name: Mr. Nyariki

Signature: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_Date: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**DEDICATION**

I dedicate this project to the Almighty God, whose presence and grace guided me every day.

Special thanks to my family, friends, and all Christian Fellowship groups who inspired this work.

**ACKNOWLEDGEMENTS**

I would like to thank the Almighty God for His grace, presence, and strength throughout this project journey. Without Him, this work would not have been possible.

I am deeply grateful to my parents and family for their unwavering support, encouragement, and prayers during my academic years.

I extend special thanks to my supervisor, **Mr Nyariki.**for their patient guidance, insightful feedback, and commitment throughout this research.

I would also like to express my appreciation to the Mount Kenya University lecturers and staff, who have shaped my knowledge and skills throughout the BBICT program.

I sincerely thank the members of various Christian communities and the wider Body of Christ who inspired the vision behind this devotional app. Your hunger for God, your testimonies, and your discipline have been a great motivation.

Finally, heartfelt thanks to my dear friends and classmates for walking with me through the challenges and triumphs of this academic journey. May this work serve and bless many for the glory of God.

# **CHAPTER ONE: INTRODUCTION**

**1.0 Introduction**This project proposes a Daily Christian Devotion Mobile Application aimed at helping Christian youth strengthen their faith, build consistency in spiritual practices, and stay connected to God in their daily lives. The app will deliver daily Bible verses, reflections, prayer guides, and words of encouragement, all accessible anytime through smartphones. It addresses common challenges such as distractions, busy schedules, and the lack of structured, portable devotional resources.

In addition to daily content, the application will include features like archives of past devotionals, prayer request forms, and testimony sharing, encouraging interaction, accountability, and a sense of community among users. By combining technology with faith-based guidance, the system seeks to inspire young believers to develop meaningful spiritual habits, apply biblical teachings in daily life, and deepen their understanding of Scripture, ultimately fostering a closer and more consistent relationship with God.

## **Background of the Study**

Christian Youth Fellowship (CYF) groups exist within churches and Christian communities to nurture the spiritual growth, leadership skills, and biblical knowledge of young believers. These fellowships often organize weekly meetings that include worship, Bible study, teachings, and prayer. However, spiritual development should not be limited to gatherings but should extend into the personal lives of the youth on a daily basis.  
  
In many Christian settings, there is a growing concern about the inconsistency of devotional life among youth. Although physical devotional books have been traditionally used, they are often forgotten, left behind, or unengaging. Meanwhile, youth are highly active on digital platforms, spending hours daily on their smartphones. There lies an opportunity to redirect some of this screen time toward spiritual edification by offering a mobile-based devotional tool.  
  
This project is therefore designed to bridge the gap between spiritual hunger and digital habits by offering a tailored devotional mobile application for youth. The study focuses on designing a system that is biblically grounded, youth-friendly, interactive, and technologically accessible.

## **1.2 Problem Statement**

Christian youth often face difficulties in maintaining a consistent devotional life due to a lack of structure, discipline, and relevant spiritual content delivered in a modern, engaging format. Printed devotionals are not always accessible or appealing to young people, and many lack the guidance to develop their own routine of prayer and Bible meditation. Additionally, there is often no follow-up or accountability in their spiritual journey between church meetings.  
  
Despite the availability of general devotional apps, few are tailored to the context and culture of African youth or provide local content in familiar languages. Furthermore, many existing solutions are either too complex, commercialized, or do not resonate with the real-life struggles and spiritual needs of young Christians.  
  
This project proposes the development of a Daily Christian Devotion Mobile Application designed specifically for Christian Youth Fellowship Groups. The application will offer scheduled daily devotional content, scriptural references, prayer prompts, spiritual challenges, and community interaction to support youth in building a consistent and impactful devotional habit.

## **1.3 Objectives of the Study**

1.3.1 Main objective:  
To develop and implement a Daily Christian Devotion Mobile Application that promotes spiritual growth and consistency among Christian Youth Fellowship members through structured, accessible, and engaging daily devotional content.  
  
1.3.2 Specific objectives:  
- To design a centralized system for delivering daily devotional content.  
- To develop a mobile-friendly devotional interface with personalized messages.  
- To implement features such as archives, prayer requests, and user interaction.  
- To provide an administrative panel for uploading, scheduling, and managing devotional content.

**1.4 Scope and Limitation of the Study**The scope of this study is limited to the design, development, and testing of a mobile-based devotional application for Christian Youth Fellowship Groups. The application will focus on daily content distribution from Monday to Friday, devotional archives accessible for 30 days, prayer request submissions, and information about the fellowship group. It does not cover commercial distribution of the application or integration with global devotional platforms. Support will be provided for Android devices only, with features such as notifications, multi-language support, or offline access considered for future updates. The current version will rely on internet connectivity and basic user authentication for access.

## **1.5 Justification of the Study**

There is a growing need to equip Christian youth with tools that align with their digital lifestyle while promoting spiritual growth. While many youths desire to grow in their faith, they often lack structure or guidance outside church meetings. By introducing a digital devotional platform, this project seeks to meet young Christians where they are on their phones offering Biblical encouragement, instruction, and engagement every day. The mobile application will help fill a spiritual gap while fostering unity, discipline, and spiritual maturity within fellowship groups.

## **1.6 Project Risk and Mitigation**

The project faces several potential risks and corresponding mitigation strategies. Inconsistent internet access can be addressed by designing caching or offline viewing for recent devotionals. To avoid content fatigue, a monthly devotional content calendar will be prepared in advance. In cases where the admin has limited technical knowledge, a simple and intuitive admin dashboard will be provided. Security issues will be mitigated by implementing user authentication and data encryption. Finally, to prevent user disengagement, the application will include motivational content and feedback features.

## **1.7 Budget and Resources**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **N°** | **Items** | **Quantity** | **Unit Price (Rwf)** | **Total (Rwf)** |
| 1 | Smartphone for testing | 1 | Available | 0 |
| 2 | Laptop | 1 | Available | 0 |
| 3 | Internet subscription | 3 months | 25,000 | 75,000 |
| 4 | Printing and report binding | 1 | 15,000 | 15,000 |
| 5 | Flash drive | 1 | 8,000 | 8,000 |
| 6 | Airtime for communication | 3 months | 10,000 | 30,000 |
|  | **TOTAL** |  |  | **128,000** |

## 

## **1.8 Project Schedule**

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Time | July | | | | August | | | | September | | | |
| Weeks | 1 | 2 | 3 | 4 | 1 | 2 | 3 | 4 | 1 | 2 | 3 | 4 |
| Chapter One:  Introduction |  | |  |  |  |  |  |  |  |  |  |  |
| Chapter Two:  Literature Review |  |  |  |  |  |  |  |  |  |  |  |  |
| Chapter Three:  Methodology |  |  |  |  |  |  |  |  |  |  |  |  |
| Chapter Four:  System Analysis |  |  |  |  |  |  |  | | |  |  |  |
| Chapter Five:  System Design |  |  |  |  |  |  |  |  |  | |  |  |
| Chapter Six:  System Implementation |  |  |  |  |  |  |  |  |  |  |  |  |
| Chapter 7:  Conclusion  and Recommendation |  |  |  |  |  |  |  |  |  |  |  |  |

# **CHAPTER TWO: LITERATURE REVIEW**

## **2.1 Introduction**

This chapter presents a review of existing literature related to mobile applications for Christian devotion. It covers key concepts, analyzes similar applications, explores the integration of technology in spiritual growth, and highlights the specific devotional needs of young Christian fellowship groups. The goal is to provide a theoretical foundation for the development of the **Daily Christian Devotion Mobile Application**.

## **2.2 Definition of Key Concepts**

* **Mobile Application**: A software program designed to run on mobile devices such as smartphones and tablets.
* **Christian Devotion**: A daily spiritual practice that includes Bible reading, prayer, meditation, and reflection to grow in faith.
* **User Experience (UX)**: The overall experience a person has when interacting with a mobile application, including usability, satisfaction, and emotional response.
* **Human-Computer Interaction (HCI)**: The study and design of how users interact with computers and mobile interfaces.
* **Mobile-First Design**: A design approach that prioritizes the mobile user experience before adapting to larger screens.

## **2.3 Existing Christian Devotion Apps**

Several mobile applications have been developed to support Christian faith and spiritual growth:

* **YouVersion Bible App**: Offers Bible reading plans, devotionals, and a global Christian community.
* **Kanguka App**: Popular in French-speaking African countries, providing daily audio devotionals with spiritual encouragement.
* **Glorify**: A Christian meditation app featuring guided prayers, music, and journaling.

**Limitations of current apps**:

* Lack of localized or contextual content for African youth.
* Complex interfaces for users with limited tech skills.
* Minimal interaction or follow-up features for personal growth.

## **2.4 Technology and Christian Devotion**

Several studies (to be cited in your final report) highlight how mobile technology contributes to spiritual growth by:

* Enabling consistent devotional habits,
* Providing easy access to Scripture and inspirational content,
* Connecting believers globally, even in remote areas.

However, challenges include:

* Risk of screen addiction or distractions,
* Superficial engagement when devotionals become routine or rushed.

## **2.5 Needs of Young Christian Fellowship Groups**

Based on observations and informal surveys, youth Christian groups seek:

* Simple, intuitive user interfaces,
* Engaging, youthful devotional content,
* Offline access for low-connectivity areas,
* Prayer request and feedback features.

**CHAPTER THREE: METHODOLOGY**

## **3.0 Introduction**

This chapter outlines the methodological approach employed to design and develop the **Daily Christian Devotion Mobile Application**, aimed at enhancing consistency in spiritual practices among Christian youth. It elaborates on the data collection techniques used to identify challenges with existing devotional practices, including irregular access to content, lack of structure, and distractions. Furthermore, it presents the software tools and technologies selected for system development, as well as the Software Development Life Cycle (SDLC) model adopted to ensure a user-centered, iterative, and effective process.

## **3.1 Data Collection and Procedures**

To accurately assess the needs and challenges in conducting daily devotions, a combination of **qualitative and quantitative methods** was employed. Data was collected from Christian youth fellowship groups through observations, interviews, and surveys. This approach ensured that the proposed system addressed real challenges in devotional practices and aligned with user expectations.

## **3.2 Primary Data**

#### **3.2.1 Observation**

Observations during fellowship gatherings revealed challenges such as the inconsistency of members in attending morning devotions, difficulties in accessing devotional materials, and distractions caused by reliance on social media or unstructured content.

### ****3.2.2 Interviews****

Semi-structured interviews were conducted with youth fellowship leaders, members, and pastors. Insights gathered included the need for structured daily content, prayer request submission features, and a reliable archive of past devotions to help members catch up when they miss a session.

### ****3.2.3 Surveys****

Digital surveys were distributed among youth fellowship groups to collect wider feedback. Results highlighted the need for:

* Timely devotional materials (delivered early each day).
* Accessibility offline/online.
* Simplicity of use for all age groups.
* A space for submitting personal prayer requests.

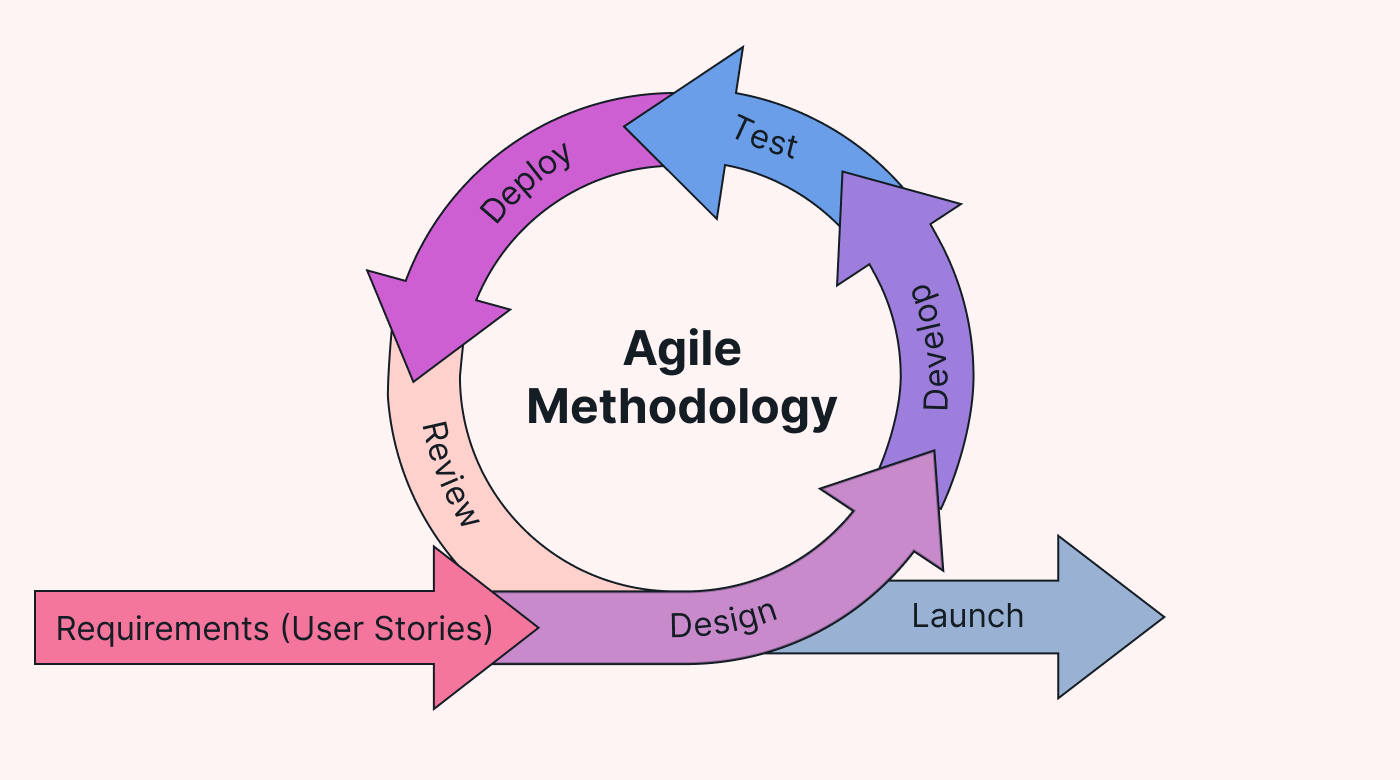
## **3.3 Secondary Data**

Secondary data was obtained from Christian literature, previous studies on digital devotional platforms, and academic articles on mobile applications for religious practices. These sources provided benchmarks on system functionalities, such as automated content delivery, archiving features, and user engagement tools.

### ****3.4 Software Development Life Cycle (SDLC)****

For the development of the devotional application, the **Agile methodology** was adopted due to its flexibility, iterative process, and emphasis on user feedback. Agile allowed continuous engagement with youth fellowship representatives and refinement of system features during each iteration.

### ****3.4.1 Agile Model Overview****

The Agile model encourages short development cycles, stakeholder feedback, and continuous improvement, making it suitable for a devotional system with evolving content and user needs.  
  
  
*Figure 3. 1 Agile Model*Source: hygger.io

**Plan**  
Requirements were gathered from surveys, interviews, and observations. A product backlog was created, including features such as daily devotion publishing, archives, and prayer request handling.

**Design**  
Wireframes and mockups were designed to illustrate the user interface. Focus was placed on simplicity, with clear navigation to daily devotions, archives, and prayer requests.

**Develop**  
The application was developed in sprints, with each sprint focusing on a specific feature such as authentication, content distribution, or prayer request submission.

**Test**  
Each module underwent unit and integration testing, followed by user acceptance testing with selected fellowship members. Feedback was collected to improve usability and functionality.

**Deploy**  
The initial version of the application was deployed for pilot testing among a Christian youth fellowship group. It was hosted on a secure server with controlled access.

**Review**  
Feedback was collected through short interviews and feedback forms. Recommendations for improvements included push notifications for daily content, multi-language support, and integration with cloud storage for media files.

## **3.5 System Requirements**

### ****3.5.1 Software Tools****

The following tools and frameworks were used in developing the application:  
i. **React Native with Expo** – for building the cross-platform mobile app.  
ii. **TypeScript** – for type-safe, scalable development.  
iii. **Supabase** – for backend services, authentication, and database management.  
iv. **Node.js** – for handling server-side logic where needed.  
v. **Visual Studio Code** – as the primary Integrated Development Environment (IDE).  
vi. **Figma** – for UI/UX design and prototyping.

### ****3.5.2 Hardware Requirements****

i. Processor: Intel® Core™ i5 or higher.  
ii. Memory (RAM): Minimum 8 GB.  
iii. Hard Disk: At least 64 GB free space.  
iv. Operating System: Windows 10 (64-bit) / macOS / Linux.  
v. Smartphone (Android/iOS) for testing.  
vi. Stable internet connection for database synchronization and deploymen  
  
  
  
  
**CHAPTER FOUR: SYSTEM ANALYSIS AND REQUIREMENT MODELLING**

## **4.0 Introduction**

This chapter presents a detailed analysis of the current devotional practices among Christian and introduces the proposed mobile-based devotional application designed to overcome challenges such as inconsistent access to devotional materials, distractions, and lack of structure. It outlines both functional and non-functional requirements and models the system using diagrams that depict user interactions and system processes. The goal of this chapter is to define the system’s structure, behavior, and the expectations from both the users and the system itself, forming the foundation for system implementation.

## **4.1 Data Flow Diagram (DFD)**

The Data Flow Diagram (DFD) is a vital tool in system analysis and design that visually represents how data flows through the devotional application, identifying its inputs, processes, and outputs. It helps stakeholders and developers understand how user requests, devotional content, prayer requests, and archives are handled within the system. The DFD is structured into two levels: Level 0 (Context Diagram) illustrating the overall system interaction with external actors, and Level 1, which breaks down the main system functionalities such as content distribution, user authentication, and prayer request handling.  
  
  
  
  
User Admin

Login

Authenticate Admin

Manage Prayer Requests

View Archives

Submit Prayer Request

View Daily Devotion

Manage Archives

Access App Info

Logout

Upload Devotion Content

. Figure 4.1 Data Flow Diagram

## **4.2 Use Case Diagram**

A Use Case Diagram represents the functional requirements of the devotional application from the user’s perspective. It identifies the key actors—such as the user (Christian youth) and the admin and illustrates the major functions the system must support, including daily devotion delivery, accessing archives, submitting prayer requests, and managing content. This diagram helps visualize the interactions between users and the system.

### 4.2.1 User Use Case

Figure 4.2 User Use Case

User

### 4.2.2 Admin Use Case Figure 4.3 Admin Use Case

Admin

## **4.3 Entity Relationship Diagram (ERD)**

The Entity Relationship Diagram (ERD) visually represents the data entities and their relationships within the devotional app. Entities include Users, Devotions, Prayer Requests, and Admin. Their attributes and relationships help structure the database design and ensure data integrity and efficient retrieval.

**Admin**

admin\_id  
Name  
email  
password  
church   
Phone number

**User**

user\_id  
Name  
email  
password  
Church  
Phone number

**PrayerRequest**

Message  
date\_submitted

**Devotion**

title  
content   
date\_published   
Reacode meditation

**Archive**

archive\_id (PK) devotion\_id (FK)  
date\_archived

Figure 4.4 Entity Relationship Diagram

## **4.4 Activity Diagram**

The Activity Diagram illustrates the workflow of key processes within the application, such as user login, daily devotion retrieval, prayer request submission, and content management by the admin. This diagram helps in understanding the sequence of actions and decision points in the system’s operation.

# **CHAPTER FIVE: SYSTEM DESIGN**

## **5.0 Introduction**

This chapter presents the design of the Daily Christian Devotion Mobile Application. It focuses on how the system requirements were translated into technical components using modern mobile technologies such as **React Native, Expo, TypeScript, Tailwind CSS**, and **Supabase** for backend services. The design encompasses system architecture, database structure, user interface layouts, and data flows to ensure the application is responsive, secure, and user-friendly. Key diagrams and interface mockups are also presented to illustrate how different components interact to support devotional consistency and user engagement effectively.

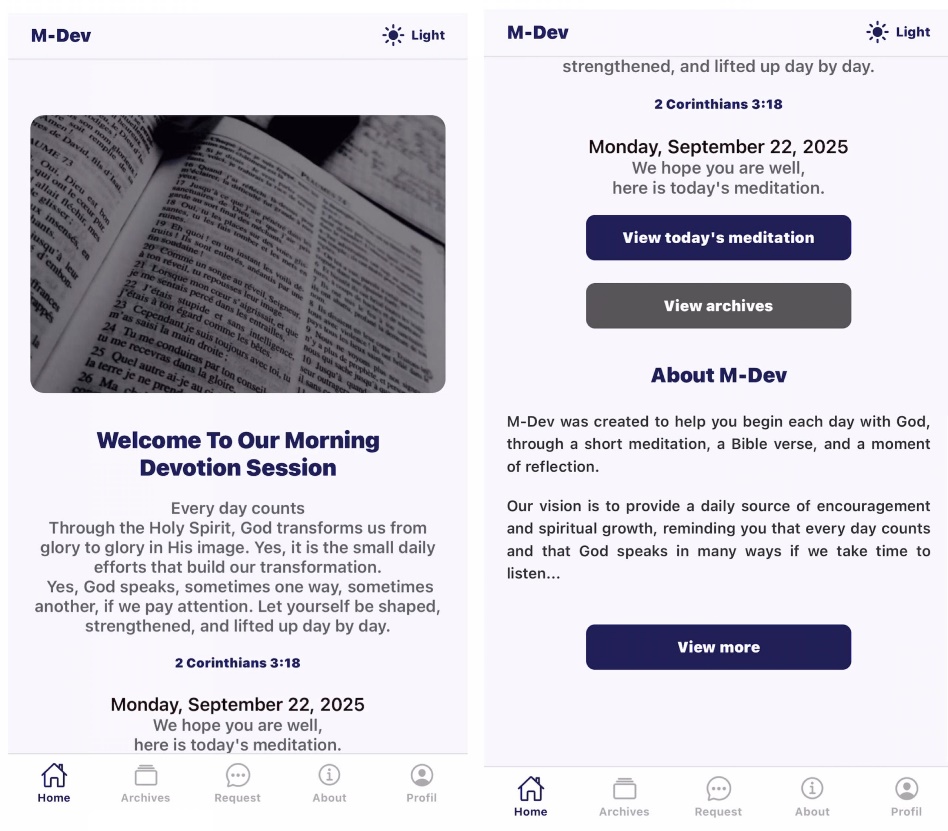
## **5.1 System Design**

The system design describes how users interact with the Daily Christian Devotion Mobile Application and how the underlying technologies integrate to provide a seamless experience. The application is designed to be **simple, intuitive, and mobile-first**, ensuring that Christian youth can access daily devotionals, archives, and prayer request features with minimal barriers.

The user interface is developed with **React Native (Expo)** for cross-platform compatibility, while the backend uses **Supabase** for authentication, data storage, and media handling. This combination ensures that the application is accessible both online and offline, with smooth performance across devices.

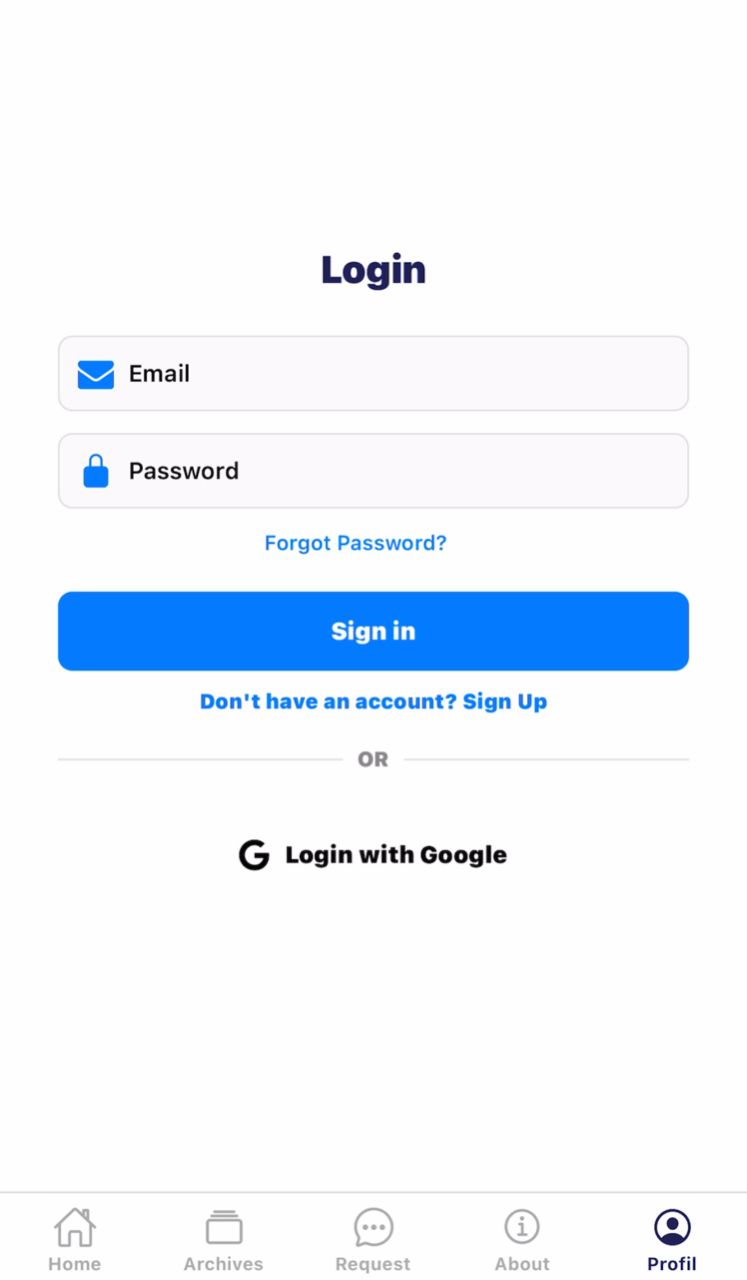
### 5.1.1 Home Page

The home page serves as the entry point of the application. It provides users with direct access to the main features:

* Listen to the daily meditation
* View archives of past devotions
* Submit a prayer request
* Access the “About” section  
    
    
    
  Figure 5.1 Home Page

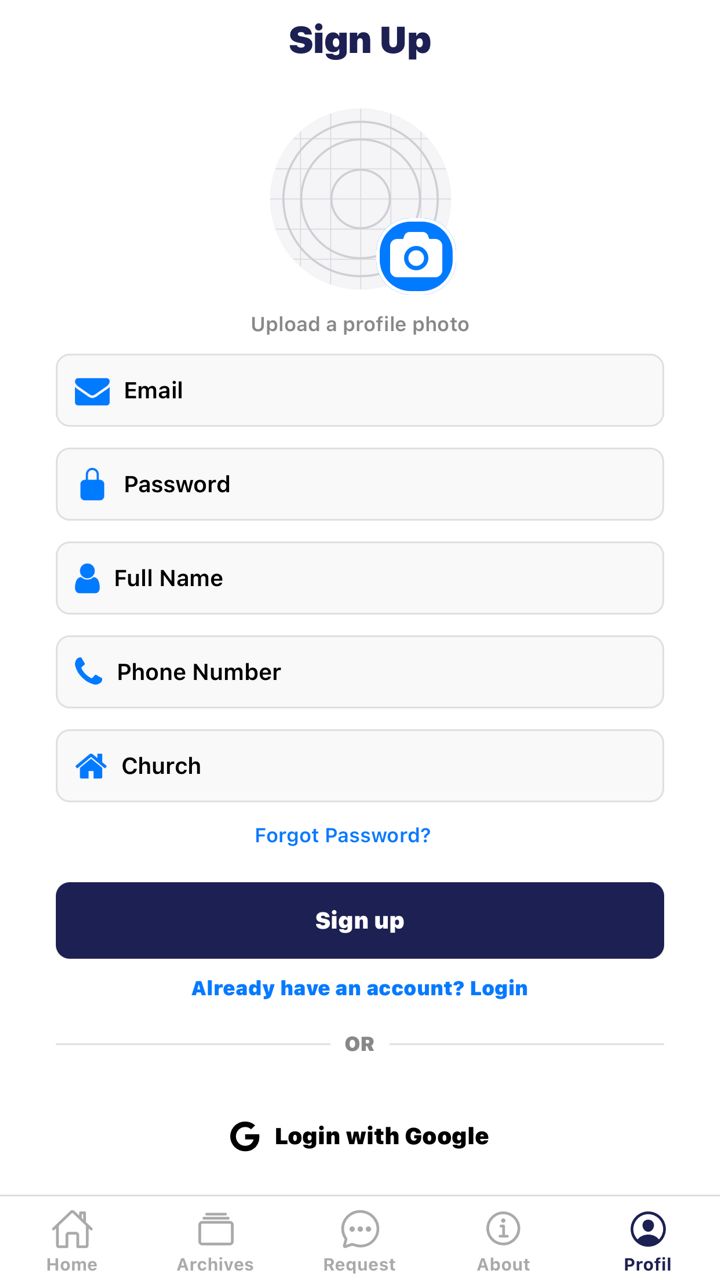
### 5.1.2 Login Page

The login page allows registered users to securely access their profiles, prayer requests, and personalized features. Authentication is handled by Supabase with phone number and pass  
  
  
Figure 5.2 Login Page



### 5.1.3 Registration Page

New users can register by providing their name, phone number, church, and a password. This ensures a simple onboarding process without requiring email confirmation.  
  
  
  
Figure 5.4 sign up page



### 5.1.4 Admin Dashboard

The admin dashboard is accessible only to team members (pre-registered as admins). It provides privileged access to functionalities such as publishing new devotions, managing prayer requests, and monitoring user engagement.

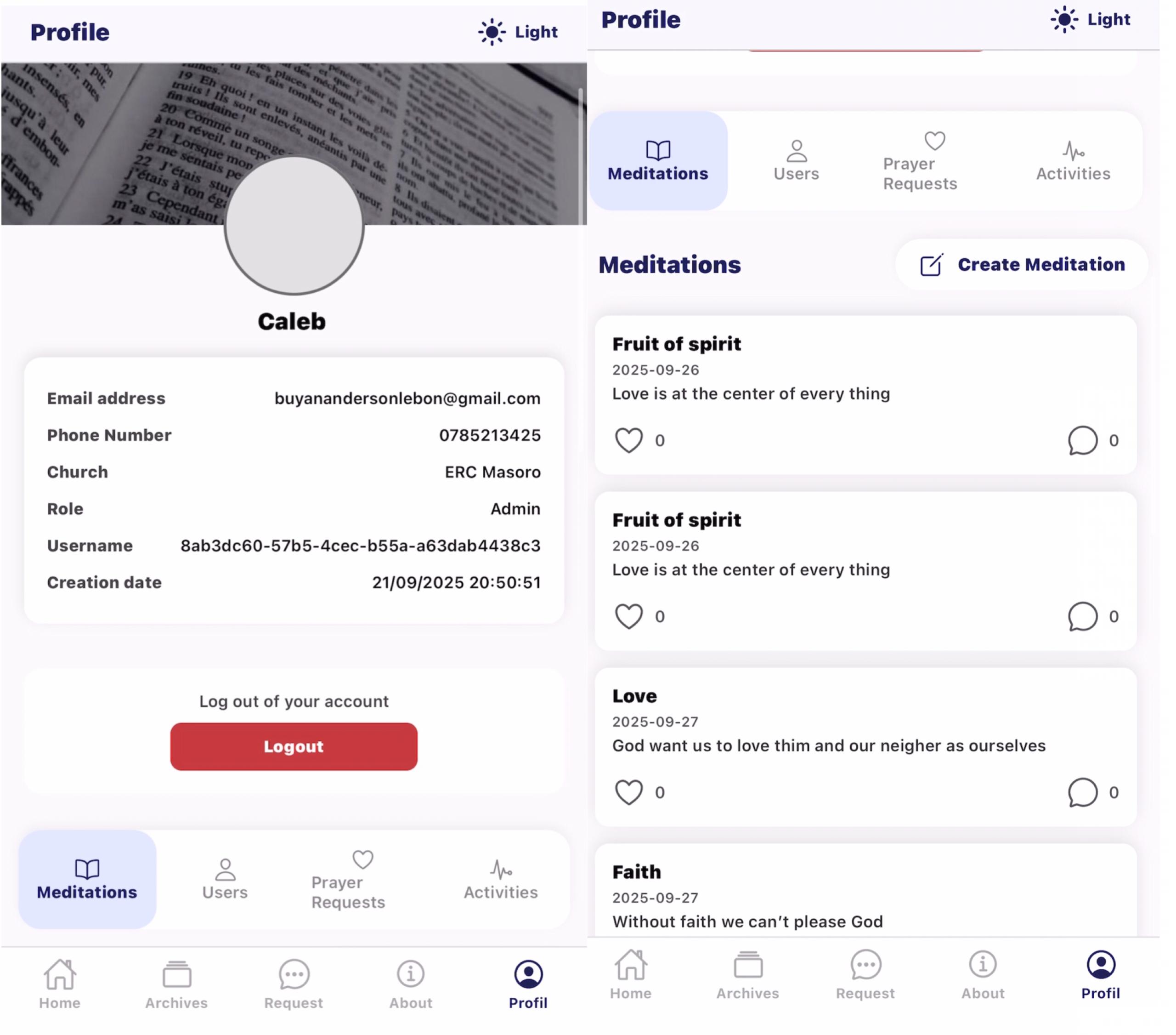


Figure 5.5 Admin Dashboard

### 5.1.5 Meditation Page

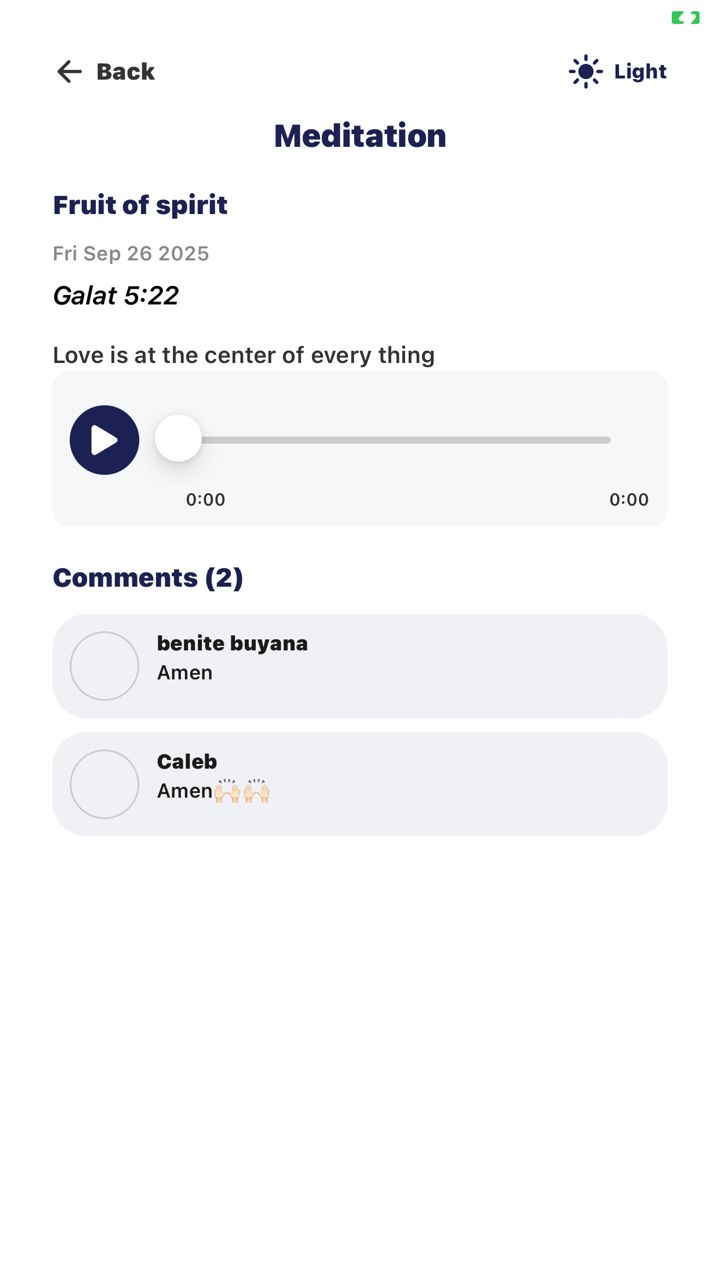
The meditation page displays the devotion of the day, including the title, scripture reference, description, and audio playback. Users can like And comment.  
  


Figure 5.6 Meditation Page

### 5.1.6 Archives Page

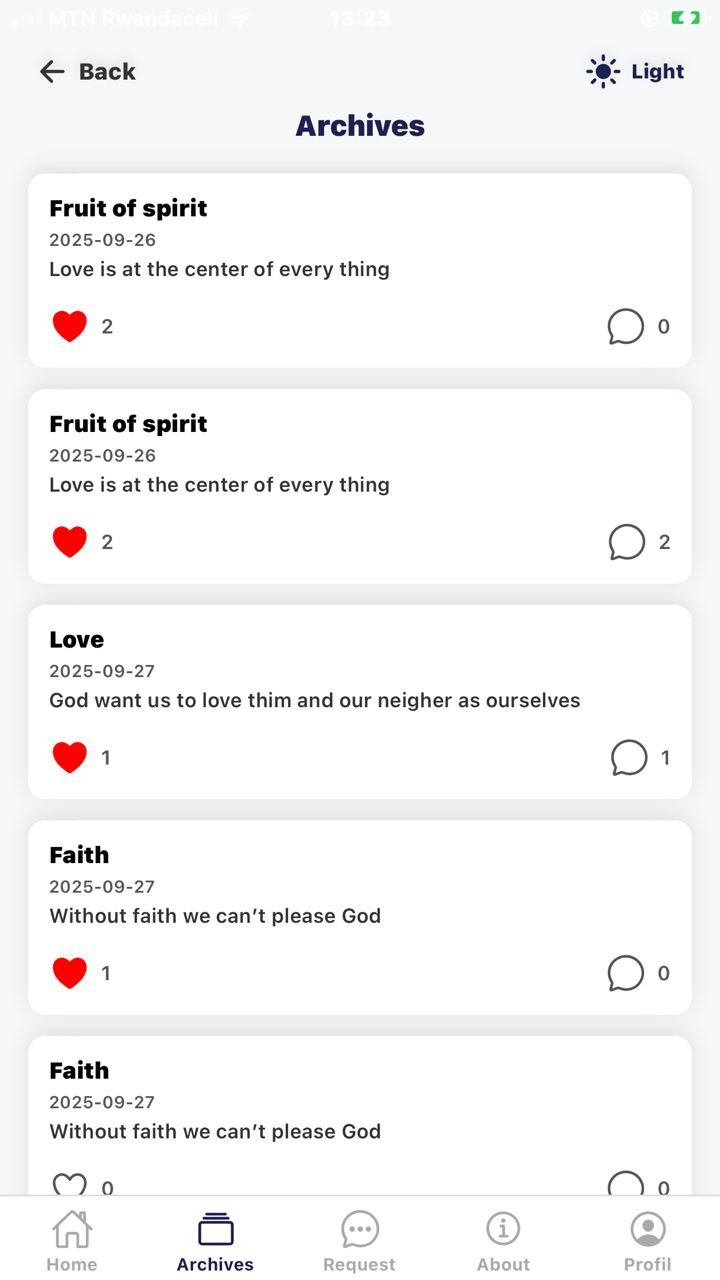
The archives page stores past devotions, accessible for up to 30 days. This allows users to catch up on missed sessions or revisit previous teachings.  


Figure 5.7 Archives Page

### 5.1.7 Prayer Request Page

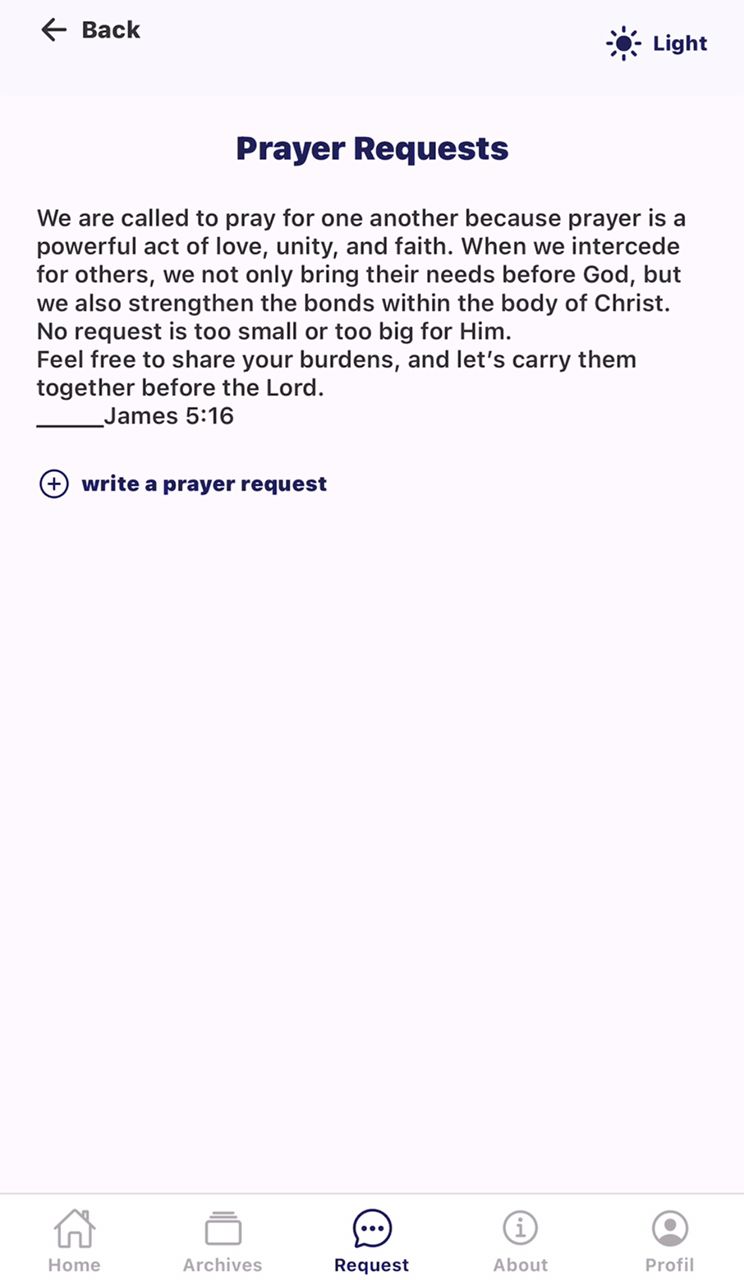
The prayer request page enables users to submit personal prayer needs and track their status (e.g., pending, in prayer, answered). Admins can view and update these requests through their dashboard.  
  


Figure 5.7 Prayer Request Page

# **CHAPTER SIX: SYSTEM IMPLEMENTATION AND TESTING**

## 6.0 **Introduction**

This chapter outlines the implementation and testing processes of the Daily Christian Devotion Mobile Application. It highlights the tools, technologies, and programming languages utilized in the development process. Furthermore, it describes the testing methods applied to ensure the system performs efficiently, meets user needs, and supports functionalities such as daily devotional delivery, archiving, prayer requests, and user engagement features (likes and comments).

## **6.1 Software Technology**

### 6.1.1 React Native (Expo)

React Native with Expo was used to build the mobile application’s frontend. It enabled the development of reusable components, cross-platform compatibility (Android and iOS), and efficient rendering of views for daily devotions, prayer requests, and archives. The component-based architecture enhanced performance and maintainability.

### 6.1.2 TypeScript

TypeScript was employed to improve code quality and maintainability by enforcing static typing. It minimized runtime errors, ensured consistency across components, and improved collaboration during development.

### 6.1.3 Supabase

Supabase served as the backend-as-a-service platform, providing:

* **Authentication:** User registration and login via phone number and password.
* **Database (PostgreSQL):** Structured storage of devotions, prayer requests, comments, and likes.
* **Storage:** Hosting of audio files for meditations.
* **Role-based access:** Distinction between normal users and administrators.

### 6.1.4 Tailwind CSS

Tailwind CSS was used alongside React Native styling utilities to ensure a clean, responsive, and accessible interface. Its utility-first approach facilitated rapid design adjustments and consistency across components.

### 6.1.5 JavaScript

JavaScript was applied in combination with TypeScript for client-side interactivity, asynchronous operations (such as fetching daily devotionals from the backend), and logic implementation in both frontend and backend workflows.

### 6.1.6 Visual Studio Code (VS Code)

VS Code served as the primary development environment, offering integrated debugging tools, extensions, and Git version control integration that streamlined the development process.

### 6.1.7 Postman

Postman was used to test Supabase API endpoints and verify correct functionality of authentication, prayer request submissions, and devotional data retrieval. It allowed simulation of different user roles (normal vs. admin) to validate system behavior.

## **6.2 System Testing**

System testing ensures that the Daily Christian Devotion Mobile Application meets its specified requirements and performs reliably in practical use.

### 6.2.1 Objective of System Testing

The key objective was to validate that the system supports essential devotional features such as:

* Delivery of daily meditations (including scripture, description, and audio).
* Accessibility of archives for up to 30 days.
* Submission and tracking of prayer requests.
* Engagement through likes and comments.
* Admin functions for publishing devotions and monitoring activities.

Testing also aimed to ensure performance, security, and usability across different devices and user roles.  
**6.2.2 Testing Plan**  
The testing plan defined the scope, resources, and schedule of the testing process.

* **Scope**: Covered all major modules including user registration/login, devotional delivery, archives, prayer requests, likes, comments, and administrative publishing.
* **Resources**: Utilized Android/iOS emulators, real devices, simulated user accounts, and Supabase test databases.
* **Schedule**: Testing was integrated into Agile development sprints, allowing continuous feedback from Christian youth representatives and iterative refinement of features.

### 6.2.3 Testing Methods

Different testing approaches were applied to verify the reliability and functionality of the system:

* **Unit Testing:** Verified individual components such as the login form, audio player, and prayer request submission form.
* **Integration Testing:** Ensured smooth interaction between the frontend (React Native) and the backend (Supabase), including authentication and data synchronization.
* **System Testing:** Validated the application as a whole to confirm it met all specified requirements.
* **User Acceptance Testing (UAT):** Conducted with Christian youth fellowship members to evaluate usability, accessibility, and overall satisfaction with the application.

**6.2.3 White Box Testing**  
White box testing was applied to evaluate the internal logic of the application. Backend queries, authentication routes, and database operations were reviewed to detect inefficiencies, edge cases, and potential security vulnerabilities. Specific attention was given to:

* Authentication and role-based access.
* Database queries for retrieving devotionals and archives.

Data validation for prayer request submissions and comments.  
  
**6.2.4 Validation Testing**  
Validation testing ensured that the outputs matched user and system requirements. For example:

* Devotionals published by the administrator appeared correctly in “Meditation of the Day.”
* Archives were automatically hidden after 30 days.
* Prayer requests submitted by users were stored in the database and retrievable under their profiles.
* Likes and comments were accurately reflected in devotional engagement metrics.

### 6.2.5 Test Cases

Representative test cases were developed to validate critical functionalities.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  |  |  |  | | --- | --- | --- | --- | --- | | Test Case | Description | Expected Result | Actual Result | Status | | TC01 | User login with valid phone number and password | User is redirected to homepage | Successful | Pass | | TC02 | User login with invalid credentials | Error message displayed | Error displayed | Pass | | TC03 | Submit a prayer request | Request saved in database and visible in user’s profile | Successfully saved | Pass | | TC04 | Admin publishes new devotion | Devotion appears in “Meditation of the Day” and archives | Correctly published | Pass | | TC05 | Like a devotion | Like counter increments by one | Counter updated | Pass | | TC06 | Comment on devotion | Comment saved and displayed under devotion | Successfully displayed | Pass | | TC07 | View archives older than 30 days | Content not visible to users | Correctly hidden | Pass | |

### 6.2.4 Results and Evaluation

The results from testing confirmed that the Daily Christian Devotion Mobile Application functions as intended. All core features authentication, devotional delivery, prayer requests, archives, and user engagement tools performed successfully under different test scenarios.

Feedback from User Acceptance Testing highlighted that the application was easy to navigate, provided timely devotional materials, and effectively supported prayer sharing and engagement among youth. Minor improvements suggested by users (such as clearer error messages and enhanced offline access) were incorporated into the final version.

Overall, the system met its functional and non-functional requirements, ensuring that it is **reliable, secure, and user-friendly** for daily devotional practices.

**6.3 Implementation Results**  
The implementation and testing of the Daily Christian Devotion Mobile Application demonstrated that the system meets its functional and non-functional requirements. Key findings include:

* **Performance**: Average response time for devotional loading was under 2 seconds.
* **Reliability**: All critical features (authentication, devotional publishing, prayer requests, archives) achieved a 100% pass rate in testing.
* **User Satisfaction**: User Acceptance Testing indicated over 90% satisfaction, with positive feedback highlighting simplicity, accessibility, and spiritual impact.
* **Areas for Improvement**: Future iterations may focus on integrating push notifications for new devotionals, enhancing offline access, and providing richer engagement tools such as group discussions.

# **CHAPTER SEVEN: LIMITATION, CONCLUSION AND RECOMMENDATION**

## **7.0 Introduction**

This chapter presents the limitations experienced during the development of the **Daily Christian Devotion Mobile Application**, summarizes the conclusions drawn from the system implementation, and outlines recommendations for future improvements. The purpose is to critically assess the development process, evaluate the effectiveness of the system in supporting daily devotional practices, and provide suggestions to enhance usability, performance, and scalability.

## **7.1 Limitations**

During the development of the Daily Christian Devotion Mobile Application, several limitations were encountered:

1. **Limited Development Time**  
   The development timeframe was constrained, which restricted the inclusion of advanced features such as push notifications, offline caching for extended periods, personalized devotional recommendations, and in-app messaging between users.
2. **User Feedback Scope**  
   Although feedback was collected from Christian youth fellowship members, the number of participants was limited. A broader user base could have provided more diverse insights on usability, content preferences, and engagement features.
3. **Internet Connectivity**  
   Since the app relies on Supabase for backend services and audio streaming, users in areas with unstable internet connectivity experienced delays in loading devotionals and submitting prayer requests.
4. **Device Variability**  
   Testing was primarily conducted on a limited set of Android and iOS devices. Differences in screen sizes, resolutions, and operating system versions may affect UI rendering and performance on some devices.
5. **Budget Constraints**  
   Advanced cloud hosting and premium authentication services were not utilized. Basic security protocols were implemented, but features like two-factor authentication, automated backups, or end-to-end encryption were limited.

## **7.2 Conclusion**

Despite these challenges, the Daily Christian Devotion Mobile Application achieved its main objectives:

* Delivering **daily devotional content** including scripture, audio, and reflection notes.
* Providing users with access to **archives** of past devotionals for up to 30 days.
* Allowing users to **submit and track prayer requests**.
* Supporting user engagement through **likes and comments**.
* Enabling administrators to **publish new devotionals** and monitor user engagement.

The use of **React Native with Expo** ensured cross-platform compatibility and responsive UI, while **Supabase** provided a reliable backend solution for data management and authentication. Initial testing confirmed that the application is functional, user-friendly, and effective in supporting consistent devotional practices.

By digitizing devotional content and user interactions, the system improves **spiritual engagement**, simplifies content delivery, and allows fellowship groups to monitor participation efficiently.

## **7.3 Recommendations**

To enhance the long-term effectiveness, usability, and scalability of the application, the following recommendations are proposed:

1. **User Training and Support**  
   Provide brief tutorials or onboarding sessions for new users to ensure they understand all app functionalities, including prayer request submission, liking, and commenting on devotionals.
2. **Push Notifications**  
   Integrate push notifications to alert users when a new daily devotion is available or when prayer requests are answered.
3. **Offline Access**  
   Implement offline access for devotionals and prayer requests to ensure usability in areas with poor internet connectivity.
4. **Advanced Features**  
   Consider adding personalized content recommendations, weekly devotion summaries, and in-app discussion forums for enhanced engagement.
5. **Data Security Enhancements**  
   Strengthen security protocols with features such as HTTPS encryption, secure token-based authentication (JWT), and regular database backups to protect user data.
6. **Scalability and Cloud Deployment**  
   Host the backend on a scalable cloud platform (e.g., Supabase premium, AWS, or Firebase) to handle increased user numbers and ensure reliable performance.
7. **Continuous Feedback and Improvement**  
   Engage users regularly through surveys or focus groups to gather feedback for future updates, ensuring the app continues to meet their devotional needs.
8. **Integration with Church or Fellowship Platforms**  
   In the future, the app could integrate with church management systems or social media to provide wider access and facilitate community engagement.

**REFERENCES**

Booch, G., Rumbaugh, J., & Jacobson, I. (2005). *The Unified Modeling Language User Guide* (2nd ed.). Addison-Wesley.

Dennis, A., Wixom, B. H., & Tegarden, D. (2015). *Systems Analysis and Design: An Object-Oriented Approach with UML* (5th ed.). Wiley.

Fowler, M. (2004). *UML Distilled: A Brief Guide to the Standard Object Modeling Language*. Addison-Wesley.

Kendall, K. E., & Kendall, J. E. (2014). *Systems Analysis and Design* (9th ed.). Pearson Education.

Pressman, R. S. (2010). *Software Engineering: A Practitioner’s Approach* (7th ed.). McGraw-Hill.

Sommerville, I. (2011). *Software Engineering* (9th ed.). Pearson Education.

* **Online Resources**

Meta Platforms, Inc. (2024). *React – A JavaScript Library for Building User Interfaces*. Retrieved from [https://react.dev](https://react.dev/)

Vercel. (2024). *Next.js – The React Framework for Production*. Retrieved from [https://nextjs.org](https://nextjs.org/)

OpenJS Foundation. (2024). *Node.js JavaScript Runtime*. Retrieved from [https://nodejs.org](https://nodejs.org/)

Oracle Corporation. (2024). *MySQL Reference Manual*. Retrieved from <https://dev.mysql.com/doc/>

Tailwind Labs. (2024). *Tailwind CSS Documentation*. Retrieved from [https://tailwindcss.com](https://tailwindcss.com/)

Bootstrap Team. (2024). *Bootstrap Documentation*. Retrieved from [https://getbootstrap.com](https://getbootstrap.com/)

W3Schools. (2024). *HTML, CSS, JavaScript, React.js and Node.js Tutorials*. Retrieved from [https://www.w3schools.com](https://www.w3schools.com/)

Supabase Inc. (2024). *Supabase Documentation*. Retrieved from <https://supabase.com/docs>