Myanmar Institute of Information Technology

II Semester 2024-2025

PROJ 3002 Special Project II

“Lotus Shrine”

Mid-Term Report

Basic Project Details

Project Title	**: Lotus Shrine**
Project Area	**: Digital Platform for Buddhist Meditation and Prayer Utilizing Web Development and AI Integration.**
Project Group Name	**: Team PeaceTech**
Project Members	**:**
	1. Pai Min Thway (2022-MIIT-CSE-002) **2. Khant Nyar Thwin (2022-MIIT-CSE-018)** **3. Myat Mon Mon Zaw (2022-MIIT-CSE-027)**
Instructor-in-charge	**: Dr. Myat Thuzar Tun**
Project Supervisor	**: Dr. Phyu Myo Thwe**
Offered in	**: II Semester 2024-2025 (3th July 2025 – October 2025)**

Abstract

The Lotus Shrine project is an innovative web-based application designed to provide a virtual platform for Buddhist meditation and prayer, catering to users who may not be able to physically travel to pagodas or monasteries. It seamlessly blends ancient Buddhist practices with modern web development and AI integration to foster mindfulness, peace, and cultural connection.

This mid-term report outlines the significant progress made on the project. We are pleased to report that the project is currently halfway through its scheduled duration and is nearing completion. We have successfully implemented core features, including immersive pagoda viewing, guided meditation sessions, and importantly, the AI-based posture detection for meditation and other yoga poses. The primary remaining task is the completion of the "Koe Na Win Dashboard" for logged-in users. Following the upcoming mid-term seminar, we will focus on refining and upgrading the system based on the valuable feedback received from our supervisors and advisor.

Table of Contents

No.	Title	Page
1	Introduction	4
	1.1 Introduction to the Project	4
	1.2 Project Overview and Scope	4
2	Project Objectives	5
3	Background and Core Technologies	7
	3.1 Frontend Development	7
	3.2 Backend Development	7
	3.3 Database Management	8
4	Project Overview and Implementation	9
	4.1 Overall Project Structure and Flow	9
	4.1.1 Use Case Diagram	10
	4.1.2 Main Application Flowchart	11
	4.1.3 Services Flowchart	12
	4.1.4 Koe Na Win Flowchart	13
	4.1.5 About Us Flowchart	14
	4.1.6 Account Management Flowchart	15
	4.1.7 Entity-Relationship (ER) Diagram	16
	4.2 Key Feature Implementation	17
	4.3 Component Communication and State Management	18
	4.4 Build and Deployment Considerations	18
5	Remaining Work and Future Plans	19

5.1. Remaining Tasks 19

5.2. Future Enhancements and Considerations 20

6 Conclusion 21

List of Figures

No.	Figure Title	Page
4.1.1	Use Case Diagram	10
4.1.2	Main Application Flowchart	11
4.1.3	Services Section Flowchart	12
4.1.4	Koe Na Win Dashboard Flowchart	13
4.1.5	About Us Section Flowchart	14
4.1.6	Account Management Flowchart	15
4.1.7	Entity-Relationship (ER) Diagram	16

1. Introduction

1.1 Introduction to Lotus Shrine

The Lotus Shrine project is a web application designed to offer a virtual Buddha Pay Homage room, providing comprehensive virtual Buddhist meditation and prayer experiences. This initiative aims to bridge the gap for individuals who may be unable to physically travel to pagodas or monasteries. In an increasingly digital world, the demand for accessible spiritual practices has grown, and this project seeks to meet that need by thoughtfully combining ancient Buddhist traditions with modern web development and artificial intelligence. Our overarching goal is to create a peaceful, user-friendly, and technologically advanced digital space that supports and deepens individual spiritual journeys, promoting mindfulness and devotion irrespective of geographical limitations.

1.2 Project Overview and Scope

The Lotus Shrine app creates a virtual Buddha Pay Homage room. Here, users can access Buddha Teachings through audio and video. They can also meditate using a timer, with the option to play offering audio continuously during meditation or as background sound on the website. The app offers virtual pagoda visits. A special feature lets users see the pagoda from a specific "corner" (like the Monday corner) based on their birthday. Meditation sessions get better with AI-powered posture feedback in real-time. Users can also find traditional chants and daily Dhamma quotes.

A key part of the app is the "Koe Na Win Dashboard." This feature helps users in Myanmar with their traditional Buddhist bead-counting practice. It sends timely alarms and notifications (by message or email) and shows what tasks or prayers need to be done each day. This report shows our progress and what still needs to be done to finish the Lotus Shrine. Chapter 2 explains the project objectives in more detail, and Chapter 4 will cover implementation specifics.

2. Project Objectives and Our schedules

2.1 Project Objectives

The core objectives guiding the development of the Lotus Shrine web application are to:

- ***Promote mindfulness and spiritual connection***: To leverage accessible digital technology to foster mindfulness, devotion, and a deeper connection to Buddhist practices in the modern age, ultimately aiming for inner peace for all users.
- ***Provide a comprehensive virtual platform:*** To offer an accessible and inclusive digital space for Buddhist meditation and prayer experiences for users worldwide, regardless of their physical location.
- ***Develop the Koe Na Win Dashboard:*** To build and integrate the personalized "Koe Na Win Dashboard" for logged-in users, enabling them to track spiritual activities, manage meat-free day reminders, maintain a reflection journal, and fulfill traditional bead-counting vows with automated guidance and notifications.
- ***Create immersive virtual environments:*** To develop an immersive pagoda viewing experience that realistically simulates a physical visit, including the ability to view the pagoda from specific traditional "corners" (e.g., Monday corner) corresponding to a user's birthday, thereby enhancing the spiritual connection.
- ***Integrate intelligent posture detection:*** To implement guided meditation sessions with AI-powered posture detection using Google Teachable Machine, providing real-time feedback to users for improved alignment and practice.
- ***Ensure privacy and security:*** To prioritize user privacy and data security by requesting clear permissions for webcam access, ensuring no video data is stored or transmitted, providing options to turn posture detection on/off, and securely storing all logged-in user data accessible only to them.

2.2 Project Schedule

Weeks	Description
Weeks 1-2	Initiate the project by organizing the team and collecting detailed requirements. Define and finalize the project goals, deliverables, and overall system specifications.
Weeks 3-4	Create the system's architecture and design the database structure. Begin UI/UX design using Figma by developing wireframes and interactive prototypes. Start front-end development using React.js along with HTML, CSS, and JavaScript.
Weeks 5-6	Enhance the UI/UX designs and commence back-end development with PHP. Establish the MySQL database and implement connections between front-end and back-end components.
Week 7	Integrate AI-driven posture detection using Google Teachable Machine and test its functionality within the meditation feature.
Week 8	Prepare and deliver the Mid-Semester Seminar and report. Showcase the project progress, including UI/UX prototypes and AI integration demos.

Weeks 9-11	Apply feedback from the seminar to improve UI elements and AI features. Conduct thorough debugging and optimize system performance.
Weeks 12-14	Perform comprehensive system testing with real data and user acceptance testing (UAT). Refine the application interface and features based on test results.
Weeks 15-16	Complete final documentation and reporting. Prepare presentation materials and perform last rounds of testing and UI/UX improvements.
Week 17	Submit the final report and conduct the final seminar presentation, demonstrating the fully functional Lotus Shrine web application.

We have passed the journey of 8 weeks.

3. Background and Core Technologies

The development of the **Lotus Shrine** web application is built upon a robust and modern technical stack, carefully selected to ensure a responsive, interactive, and secure user experience. This section outlines the core technologies utilized across the frontend, backend, and database components of the project.

3.1. Frontend Development

The client-side of the application is primarily developed using **React.js**, leveraging its component-based architecture for building a dynamic and user-friendly interface. Components are organized to handle specific features and views, allowing for modular development and maintainability.

- **HTML**: Provides the fundamental page structure for all web interfaces.
- **CSS (with Tailwind CSS):** Manages the styling and layout, ensuring a modern and responsive design across various devices. Components from ShadCN/UI are integrated to provide consistent, accessible, and aesthetically pleasing UI elements, leveraging Tailwind CSS for utility-first styling.
- **JavaScript / TypeScript:** Serves as the core logic and interactivity layer. TypeScript is used throughout the project for enhanced code quality and type safety, facilitating robust development of features.
- **Vite:** This build tool is employed to provide a fast development environment and efficient bundling of frontend assets.
- **Asset Management:** Various assets, including images for features (e.g., virtual pagoda, meditation), custom fonts, and background audio for immersive experiences, are systematically managed and integrated into the frontend.
- **Animation Libraries:** Framer Motion is used for smooth page transitions and micro-interactions, while GSAP (GreenSock Animation Platform) is utilized for more complex, high-performance animations, adding rich visual effects and enhancing user engagement. Consideration for Three.js / Panolens.js (for advanced 3D or 360-degree pagoda views) is also noted.

3.2. Backend Development

The server-side logic and API management for Lotus Shrine are handled by **PHP**. The backend is structured to support user interactions and data retrieval for the frontend.

- **User Management:** Functions are implemented to handle user registration, email verification, login authentication, password updates, and account recovery. These functions connect securely with the database to manage user credentials.
- **Content Fetching**: APIs are in place to dynamically retrieve content, such as daily Dhamma quotes, which are then delivered to the frontend for display.
- **Communication:** Backend scripts manage contact form submissions, allowing users to send inquiries or feedback.
- **API Development:** PHP forms the backbone for building various API endpoints that facilitate seamless communication and data exchange between the React.js frontend and the MySQL database.

3.3. Database Management

MySQL serves as the relational database for the application, responsible for securely storing and managing all project data.

- **Schema:** The database schema is defined using SQL, outlining the structure for storing different types of data, including user information, prayer logs, meditation records, and Dhamma quotes.
- **Data Operations**: Custom PHP classes are developed to interact with the MySQL database, enabling efficient CRUD (Create, Read, Update, Delete) operations. This allows the application to save user-specific data (like prayer history and reminders), manage user accounts, and retrieve content as needed.
- **Security Measures:** Mechanisms are in place within the database interaction layer to handle user data securely, including aspects like rate limiting for certain operations and protecting sensitive user information.

  By combining these technologies, the Lotus Shrine project aims to deliver a seamless, feature-rich, and spiritually enriching experience.

4. Project Overview and Implementation

The Lotus Shrine web application is structured to deliver a rich, interactive, and spiritually enriching experience through a well-defined architecture and systematic implementation. This section details the overall project structure, key features, and how various components integrate to achieve the stated objectives.

4.1. Overall Project Structure and Flow

The project follows a standard modern web application structure, separating frontend and backend concerns for efficient development and scalability.

- **Frontend (User Interface):** Built with React.js and TypeScript, the frontend handles all user interactions, visual presentation, and client-side logic. The application's entry point renders the main application component, which manages global providers (like the music player context) and client-side routing. Users navigate through various pages such as the home screen, meditation sessions, Dhamma teachings, traditional chant sections, the Koe Na Win dashboard, and authentication flows (login and sign-in).
- **Backend (Server-side Logic):** Developed in PHP, the backend is responsible for managing server-side operations, handling API requests from the frontend, and interacting with the database. It provides functionalities for user authentication, data retrieval (e.g., Dhamma quotes), and communication (e.g., contact forms).
- **Database (Data Storage):** MySQL serves as the data repository, storing all persistent information, including user details, prayer logs, meditation records, and content like Dhamma quotes and mantra data.

The communication flow is primarily Frontend (React) ←→ Backend (PHP) ←→ Database (MySQL), with data exchanged via a RESTful API using JSON.

4.1.1 Use Case Diagram

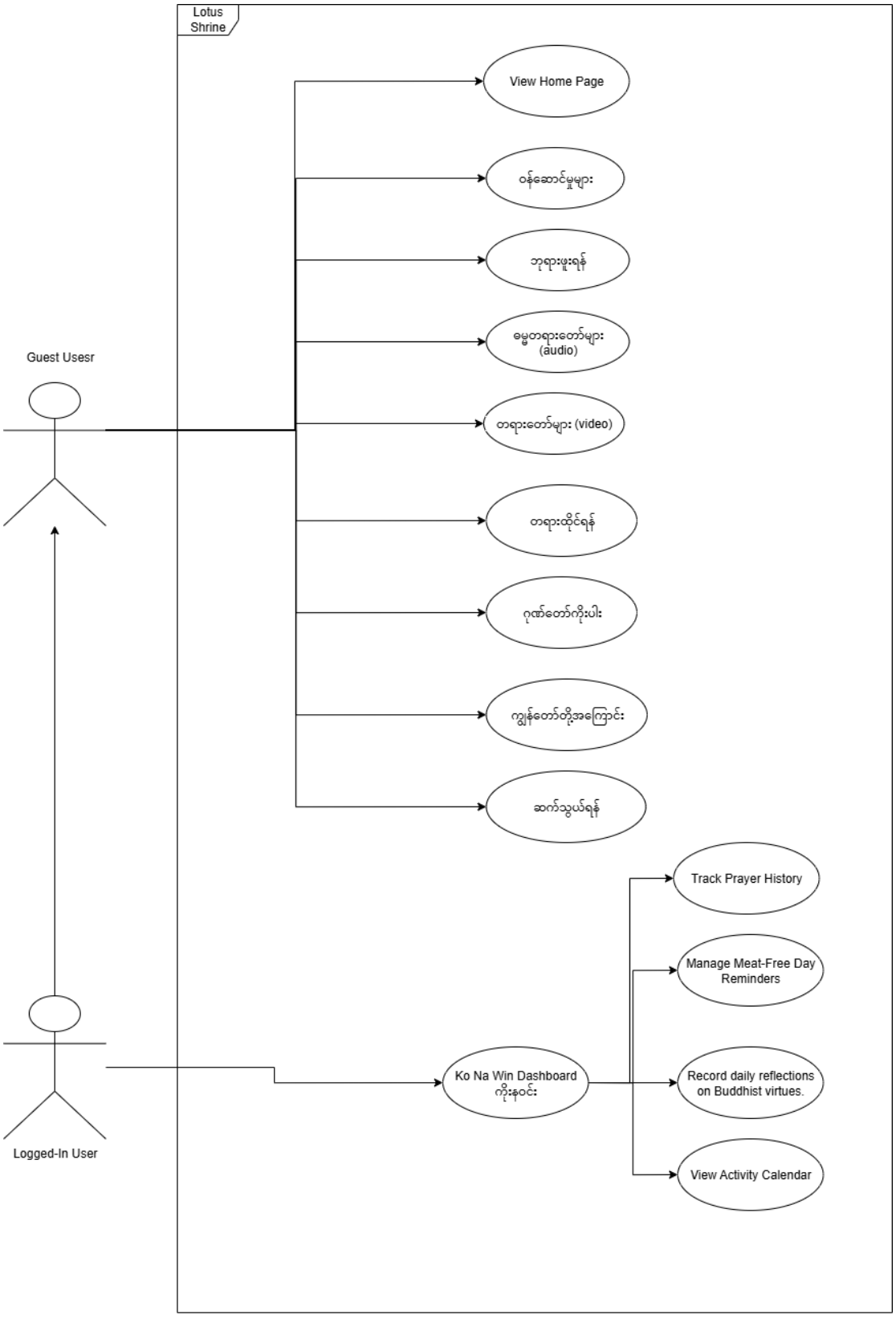

Figure 4.1: Use Case Diagram – Lotus Shrine Web Application

4.1.2 Main Application Flowchart

This flowchart outlines the primary navigation paths starting from the application's entry point, showing how users move from the home page to the main feature sections.

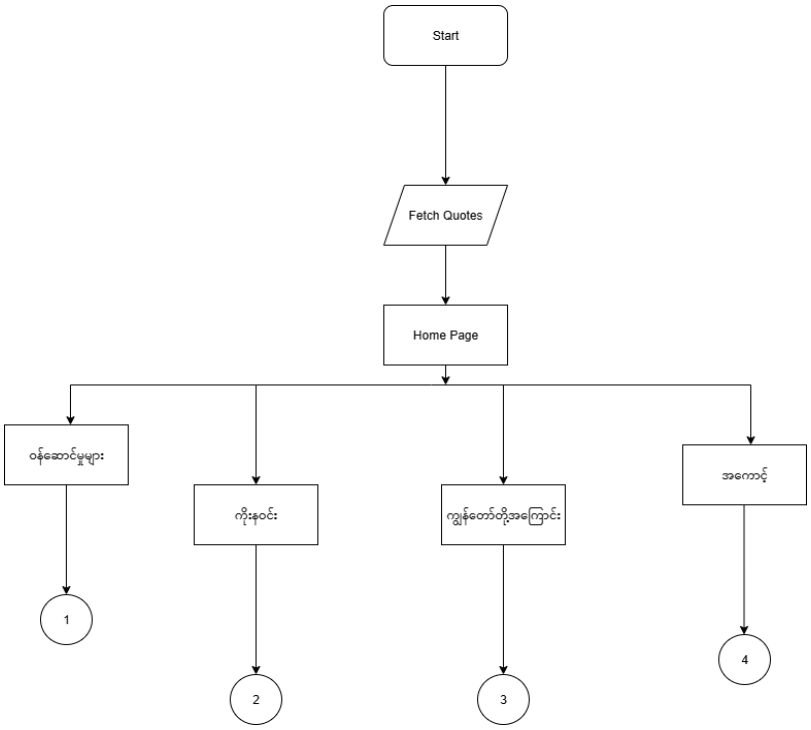

Figure 4.2: Main Application Flowchart

Description: This diagram illustrates the initial user journey from the application's start, through fetching daily quotes and landing on the home page. It then branches out to the main sections of the application: Services, Koe Na Win, About Us, and Account Management, each leading to their respective detailed sub-flows.

4.1.3 Services Flowchart

This flowchart details the functionalities available within the "ဝန်ဆောင်မှုများ" (Services) section, which users can access from the main application flow.

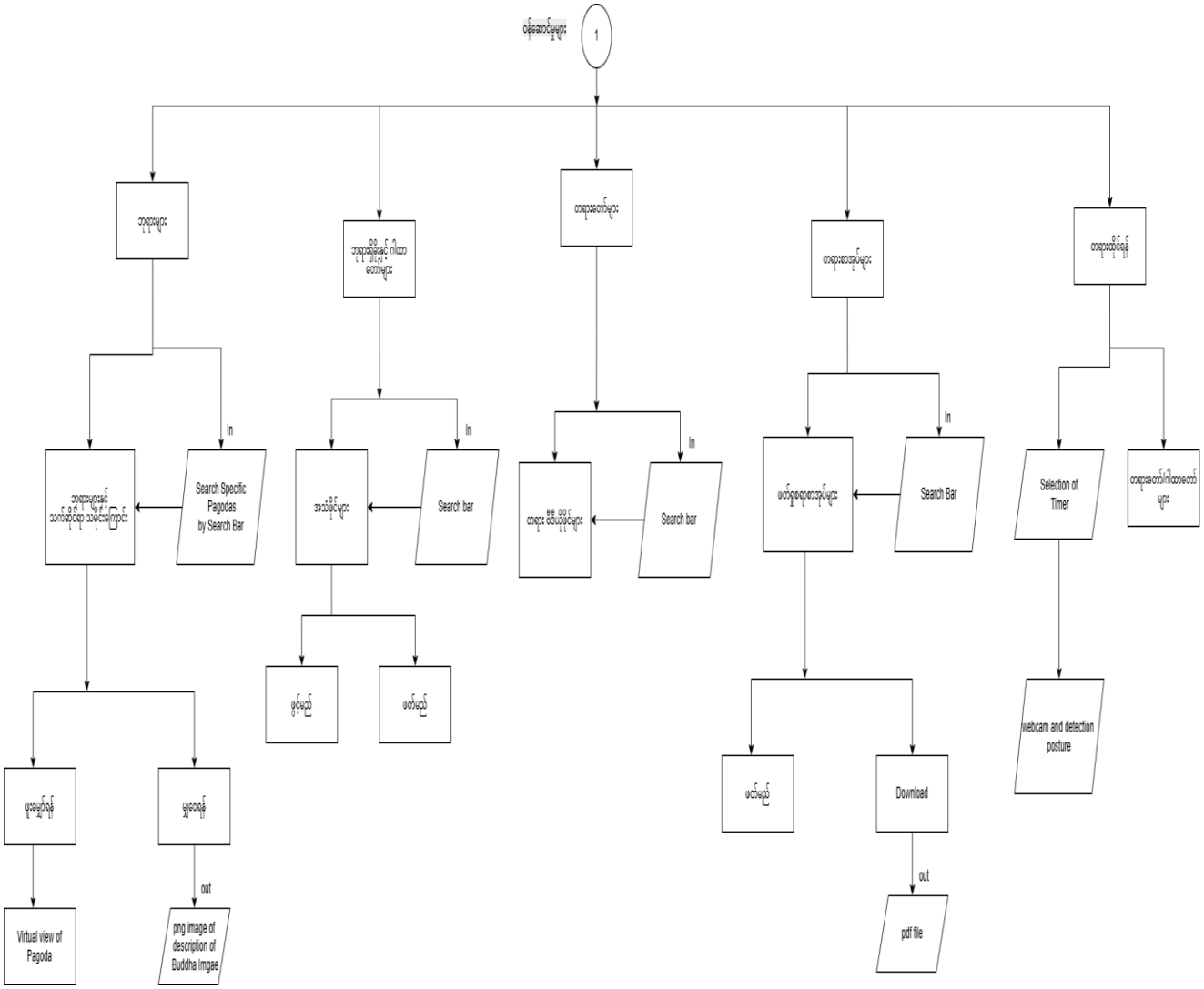

Figure 4.3: Services Section Flowchart

Description: This diagram maps out the various features accessible from the Services page. It includes pathways for viewing pagodas, accessing Buddha teaching audio and video, initiating meditation with AI posture detection, and navigating to explanations of the Nine Attributes of the Buddha. It also shows interaction with search functionalities for specific content.

4.1.4 Koe Na Win Flowchart

This flowchart provides a detailed look into the user journey and functionalities within the "ကိုးနဝင်း" (Koe Na Win) dashboard.

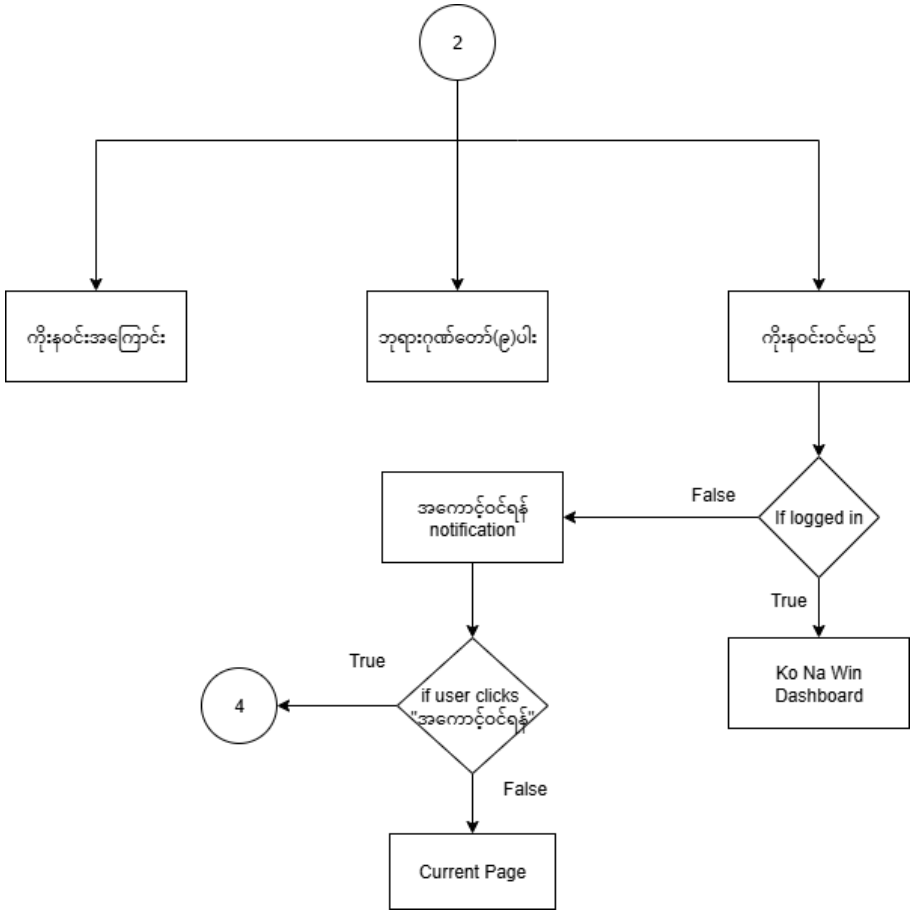

Figure 4.4: Koe Na Win Dashboard Flowchart

Description: This diagram illustrates the flow for accessing the Koe Na Win dashboard. It includes a decision point for user login status, leading to either a login notification or direct access to the dashboard. It also outlines sub-paths for activities related to the Nine Attributes of the Buddha and additional Koe Na Win functionalities.

4.1.5 About Us Flowchart

This flowchart outlines the navigation and interactions within the "ကျွန်တော်တို့အကြောင်း" (About Us) section, including its sub-pages.

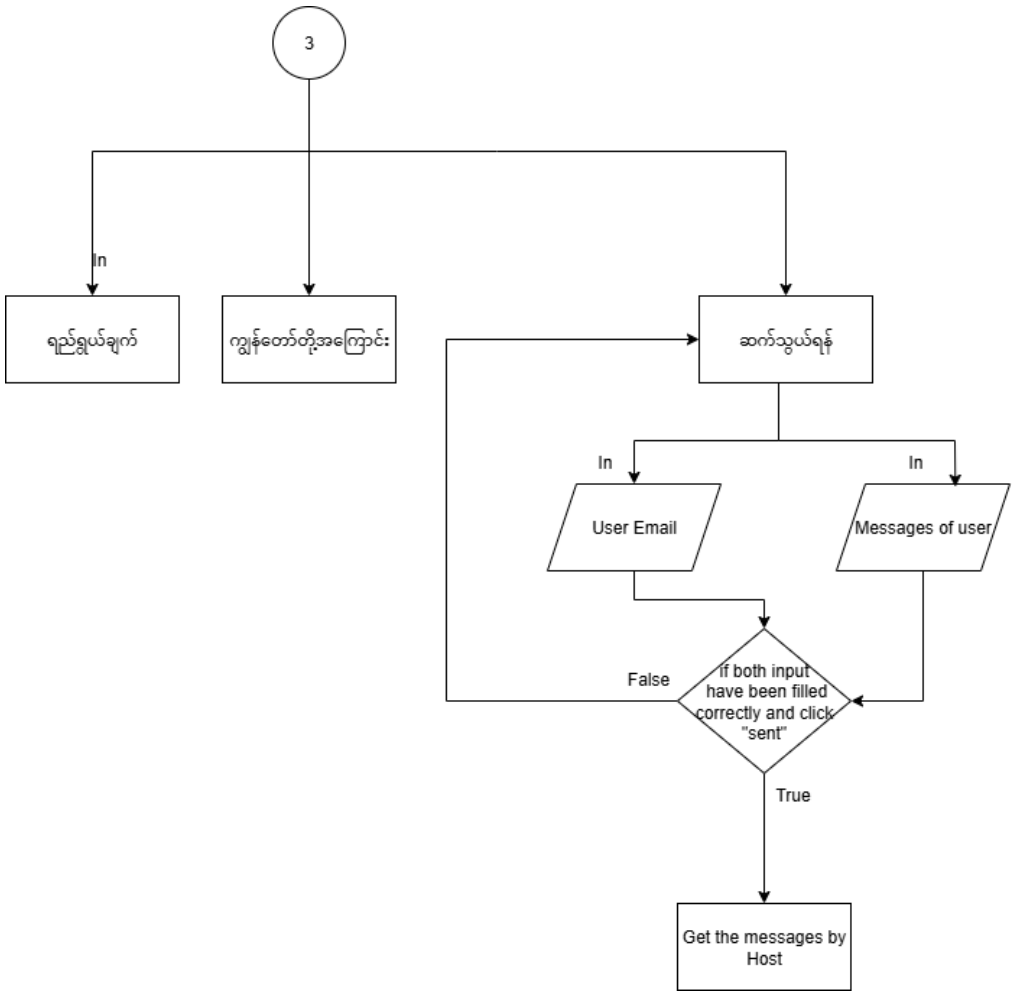

Figure 4.5: About Us Section Flowchart

Description: This diagram details the navigation within the About Us section, showing paths to the Mission (ရည်ရွယ်ချက်), About Us (ကျွန်တော်တို့အကြောင်း), and Contact Us (ဆက်သွယ်ရန်) pages. It also illustrates the process for submitting user inquiries via email through the Contact Us form, including input validation and message delivery.

4.1.6 Account Management Flowchart

This flowchart depicts the various processes involved in user account management, including login, registration, and password handling.

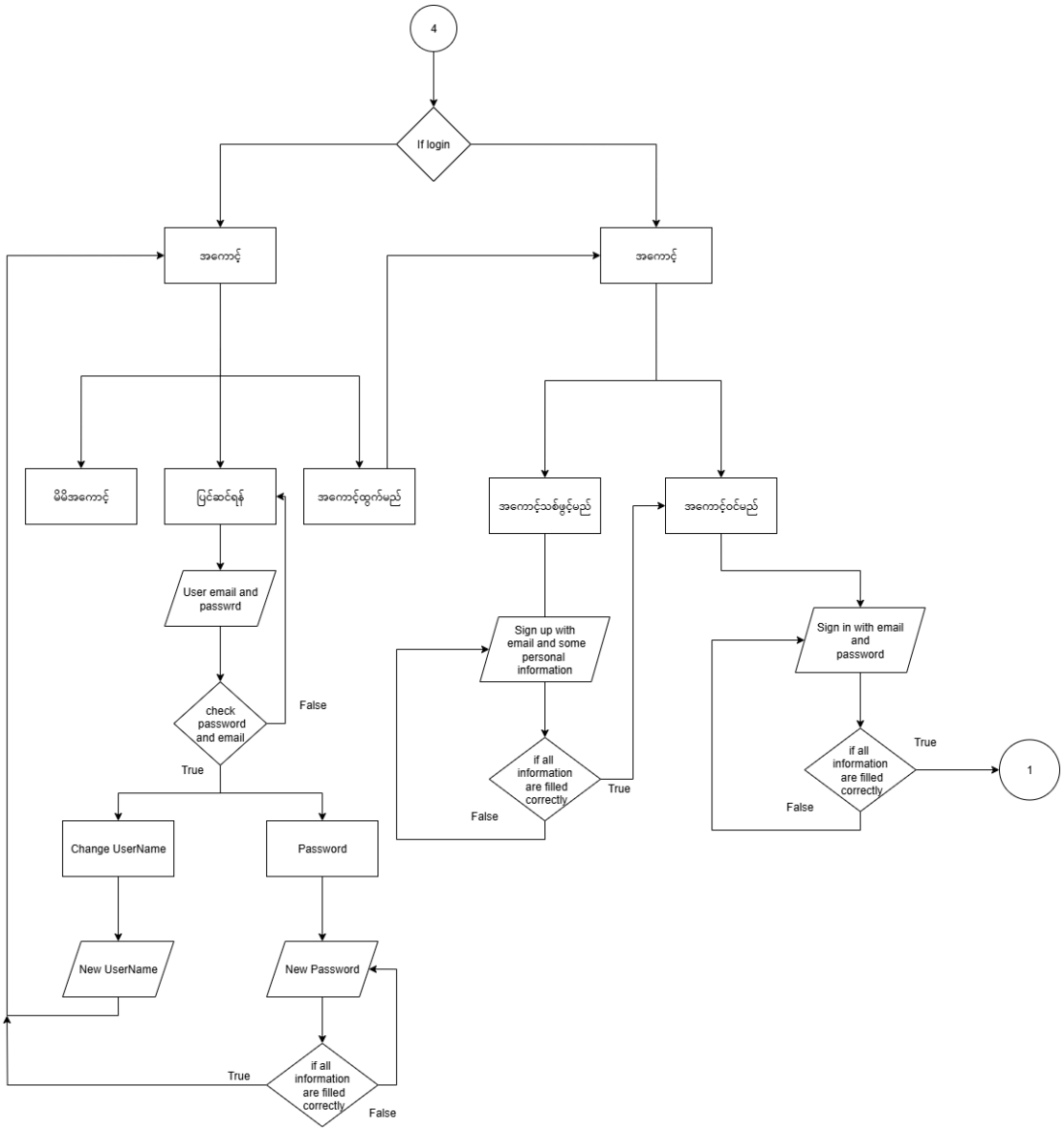

Figure 4.6: Account Management Flowchart

Description: This diagram illustrates the different account-related functionalities, including user login authentication, password management (change and reset), and profile updates. It shows decision points for login success/failure and correct information submission, guiding users through account-related actions.

4.1.7 Entity-Relationship (ER) Diagram

This diagram visually represents the database structure, showing the entities (tables) within the Lotus Shrine system and the relationships between them.

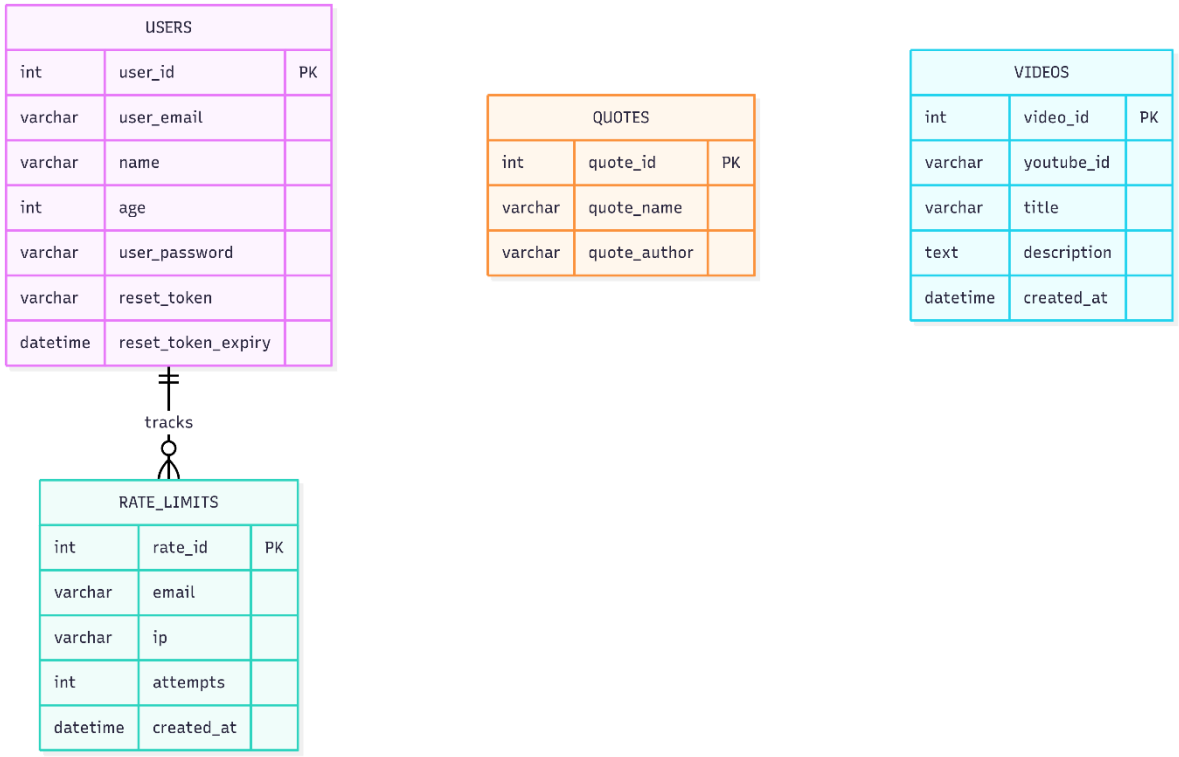

Figure 4.7: Entity–Relationship Diagram for User, Rate Limit, Quote, and YouTube Video Table

4.2. Key Feature Implementation

The core functionalities of the Lotus Shrine application are implemented as follows:

- **Virtual Buddha Pay Homage Room:** This central feature encompasses several integrated elements. Users can virtually visit pagodas, with the capability to view specific traditional "corners" of the pagoda (e.g., Monday corner) based on their birthday, enhancing a personalized spiritual connection. This visual experience is complemented by background offering audio that can be continuously played.
- **Buddha Teachings (Audio & Video):** The application integrates a dedicated section for accessing various Buddha Teachings.
    - **Dhamma Teachings (Audio):** Accessed by clicking on the "ဓမ္မတရားတော်များ" square on the services page.
    - **Buddha Teaching (Video):** Accessed by clicking on the "တရားတော်များ" square on the services page. These sections provide users with comprehensive resources for learning and spiritual growth.
- **Meditation Timer & Guided Sessions:** Users can engage in meditation with a customizable timer. This feature includes calming visuals and guided audio sessions. The integration of AI-powered posture detection using Google Teachable Machine provides real-time feedback during meditation, helping users maintain correct posture and reduce injury risk. This functionality is accessible by clicking the "တရားထိုင်ရန်" square on the services page.
- **Traditional Chants (Paritta Sutta Audio Player):** An in-app audio player allows users to listen to an organized playlist of Suttas. This player includes standard controls and optional subtitle display, enabling users to listen while browsing other sections of the app.
- **Daily Dhamma Quote:** A backend API is used to dynamically fetch and display a new Dhamma quote daily, which is stored in the MySQL database. This is prominently displayed on the homepage. *
- **Virtual Pagoda View:** Users can click on the "ဘုရားများ" square on the services page to access the virtual pagoda viewing feature. (This will be presented in a subsequent figure.) *
- **Koe Na Win Dashboard (Pending Completion):** This personalized dashboard, crucial for supporting the traditional Buddhist bead-counting practice in Myanmar, is a major focus for the remaining development. It is accessible by clicking the "ကိုးနဝင်း" square on the services page. It will provide:
    - **Activity Tracking:** Logging of prayer history (pagoda, date, notes).
    - **Reminders:** Automated notifications (via messages or email) for meat-free days and daily tasks related to Koe Na Win vows.
    - **Reflection Journal:** A space for daily reflections on the Nine Virtues of the Buddha.
    - **Activity Calendar:** A visual tracker to overview spiritual activities.
- **Explanation of Nine Attributes of the Buddha:** Users can access an explanation of the Nine Attributes of the Buddha by clicking on the "ဂုဏ်တော်ကိုးပါး" square on the services page. (This will be presented in a subsequent figure.)

- **Contact Us:** The application includes a "ဆက်သွယ်ရန်" (Contact Us) webpage, which is a sub-page of the "ကျွန်တော်တို့အကြောင်း" (About Us) section. This "About Us" section further includes "ရည်ရွယ်ချက်" (Mission) and "ကျွန်တော်တို့အကြောင်း" (About Us) itself. This page allows users to send inquiries or feedback. *

4.3. Component Communication and State Management

- **Global State:** A context manages the global music player state, ensuring a seamless audio experience across different pages. User authentication data is persisted using Local Storage.
- **Data Flow:** Components primarily communicate via props (parent to child data passing) and the React Context API for global state sharing (MusicPlayer, Authentication). React Router manages navigation and URL-based state changes.
- **Form Handling:** Forms throughout the application utilize React Hook Form for efficient state management and validation.
- **API Integration:** The frontend communicates with the PHP backend using the Fetch API, sending and receiving JSON data via RESTful principles. Backend endpoints handle specific requests like user authentication, registration, and contact form submissions.

4.4. Build and Deployment Considerations

- **Development Environment:** Vite serves as the build tool and development server, facilitating rapid development and hot module reloading.
- **Build Process:** The build command compiles TypeScript, bundles React components, and optimizes assets (images, fonts, sounds, CSS) for production deployment.
- **Responsive Design:** Tailwind CSS's responsive classes and custom hooks are employed to ensure the application adapts seamlessly to various devices, including mobile.
- **Animations and User Experience:** The project utilizes **shadcn/ui** components, which are built with Tailwind CSS, to provide a consistent and accessible user interface. For rich and complex visual effects, **GSAP (GreenSock Animation Platform)** is employed for high-performance animations, further enhancing user engagement and overall aesthetic quality. Additionally, Framer Motion is used for smooth page transitions and micro-interactions. Error handling is implemented with `try-catch` blocks, and user feedback is provided via toast notifications and form validation.

This comprehensive approach ensures that Lotus Shrine is not only functional but also provides an engaging and meaningful spiritual experience.

5. Remaining Work and Future Plans

The Lotus Shrine project has made significant progress, with core functionalities largely implemented. As we approach the final phase of development, our focus shifts to completing the remaining features, refining the existing system, and preparing for deployment.

5.1. Remaining Tasks

The primary tasks to be completed include:

- **Koe Na Win Dashboard Completion**: This is the most critical remaining feature. We need to fully implement all sub-functionalities of the personalized dashboard, including:
    - **Personal Prayer Log**: Developing the interface and backend logic for users to log their prayers, specify visited pagodas, and add personal notes.
    - **Meat-Free Day Reminders:** Implementing the system to allow users to set and manage their vegetarian observance days and receive automated notifications (via messages or email).
    - **Reflection Journal:** Creating the interface and data storage for users to record daily reflections on the Nine Virtues of the Buddha, with options to mark virtues as "reflected" or "chanted."
    - **Activity Calendar:** Developing the visual calendar tracker to display prayer, meditation, and vegetarian days, enabling users to view detailed activity and notes for specific dates.
- Refinement and Optimization: Based on feedback from supervisors and advisors received during the upcoming mid-term seminar, we will perform:
    - Comprehensive Debugging: Identifying and resolving any bugs or performance bottlenecks across the frontend and backend.
    - System Optimization: Improving the efficiency and responsiveness of the application, particularly for data loading and AI processing.
    - UI/UX Enhancements: Making further refinements to the user interface and overall user experience based on usability testing and feedback, ensuring cultural appropriateness and ease of use.

5.2. Future Enhancements and Considerations

Beyond the current scope, we envision several potential enhancements that could further enrich the Lotus Shrine experience:

- Donation System: Considering the implementation of a secure donation system for monasteries or related causes, if aligned with project goals.
- Augmented Reality (AR) Mode: Investigating an AR mode for an even more immersive pagoda experience, allowing users to superimpose virtual pagodas into their real-world environment.
- **Expanded Content:** Continuously adding more Buddha Teachings (audio/video), Paritta Suttas, and Dhamma quotes.
- Enhanced AI Meditation Posture Recognition: Expanding the AI's capabilities to recognize and provide feedback on a wider array of specific meditation postures. This enhancement would include:

- Posture Selection: Giving users the option to specify their desired meditation posture, such as:
    - Full Lotus (Padmasana)
    - Half Lotus
    - Burmese Position
    - Easy Pose (Sukhasana)
    - Seiza (kneeling)
    - Chair Sitting
    - Corpse Pose (Savasana, lying flat)
    - Reclined Supported Pose (lying with bolster support).

By diligently addressing these remaining tasks and planning for future enhancements, we are confident in delivering a comprehensive and impactful Lotus Shrine web application.

6. Conclusion

The Lotus Shrine web application project has achieved significant milestones at its mid-term stage, successfully combining ancient Buddhist practices with modern technology to create a peaceful and accessible spiritual experience for users worldwide. Our dedicated team has made substantial progress in implementing core features, including immersive virtual pagoda views, guided meditation sessions with AI-powered posture detection, and various Buddha Teachings in both audio and video formats. The successful integration of these complex functionalities demonstrates the project's robust technical foundation and its potential to bridge the gap between tradition and digital innovation.

While nearing completion, the remaining focus is on finalizing the personalized "Koe Na Win Dashboard," which will significantly enhance the user's ability to track and deepen their spiritual practices through automated guidance and notifications. We are committed to refining and optimizing the existing system based on feedback from our upcoming mid-term seminar, ensuring a polished and high-performance application.

Overall, Lotus Shrine is on track to deliver a meaningful and user-friendly platform that not only fosters mindfulness and devotion but also provides a unique way for individuals to stay connected with their faith in the digital age. We are confident that the continued development will result in a comprehensive and impactful spiritual tool.