



SECJ2203: Software Engineering

System Documentation (SD)

DPCM

**(DENGUE PREVENTION AND CONTROL IN
MALAYSIA)**

Version 4.0

DATE : 20 JANUARY 2025

Faculty of Computing

Prepared by: THE RESEK

Revision Page

a. Overview

This software design document shows how the process to build the system in terms of design, plan and function of each part in the Dengue Prevention and Control system in Malaysia (DPCM) such as the latest information, real-time data and system login for user safety

b. Target Audience

For our target audience starting from children aged 12 years and also adults aged 30 years and above because in this age range they are smart in using the internet so it is easier for them to understand how to use the web application because it is a user friendly web application.

c. Project Team Members

| Member Name | Role | Task | Status |
|-------------------------------------|----------------------------------|--|----------|
| MUHAMMAD ALIF BIN AZMAN | Product Manager and Scrum Master | - Launch Phase - User Story Detail - Test Cases - TC001-TC009 | Complate |
| MUHAMMAD FIRDAUS BIN MD SHAHRUNNAR | Project Owner | - Introduction - System Features - Process View | Complete |
| MUHAMMAD FIRDAUS ALIF BIN ROSLAN | UX Designer | - Design Constraints - Specific Requirements - Activity Diagram - Component Diagram | Complete |
| MUHAMMAD ANGWIN SAYRESTIAN | Developer | - Performance - User Interface Design | Complete |
| SITI NUR SYAKIRAH BINTI MOHD NADZRI | Quality Assurance | - Activity Diagram - Logical View - Test Case TC010 - TC018 | Complete |

d. Version Control History

| Version | Primary Author(s) | Sprint/Iteration | Description of Version | Date Completed |
|----------------|-------------------------------------|-------------------------|--|-----------------------|
| 1.0 | MUHAMMAD FIRDAUS BIN MD SHAHRUNNHAR | Sprint 1 | Completed Chapter 1 and 2, Section 1.0 - Section 2.6 | 10 December 2024 |
| 2.0 | MUHAMMAD FIRDAUS BIN MD SHAHRUNNHAR | Sprint 2,3 | Completed Chapter 3 and 4, Section 3.0 - Section 4.5 | 28 December 2024 |
| 3.0 | MUHAMMAD FIRDAUS BIN MD SHAHRUNNHAR | Sprint 4,5 | Completed Chapter 5, 6, 7,8 and 9, Section 5.0 - Section 9.1 | 20 January 2025 |

Table of Contents

| | |
|--|-----------|
| 1. Introduction | 4 |
| 1.1 Purpose | 4 |
| 1.2 Scope | 5 |
| 1.3 Definitions, Acronyms and Abbreviation | 5 |
| 1.4 References | 6 |
| 1.5 Overview | 6 |
| 2. Specific Requirements | 8 |
| 2.1 Persona | 8 |
| 2.1.1 Persona 1: User (Registered User) | 8 |
| 2.1.2 Persona 2: Guest | 9 |
| 2.1.3 Persona 3: Admin | 9 |
| 2.2 System Features | 10 |
| Figure 2.1: Use Case Diagram for DPCM SYSTEM | 11 |
| Table 2.1: Description of Module and Functions for DPCM SYSTEM | 12 |
| Figure 2.2: Domain Model for DPCM SYSTEM | 13 |
| Figure 2.3: State Machine Diagram for DPCM SYSTEM | 14 |
| 2.3 Launch Phase | 15 |
| 2.4 User Story Details | 20 |
| 2.4.1 US01: User Story <Guest> | 20 |
| 2.4.2 US02: User Story <User> | 22 |
| 2.4.3 US09: User Story <Administrator> | 36 |
| 2.5 Performance and Other Requirements | 47 |
| Performance Requirements | 47 |
| Other Software System Attributes | 48 |
| 2.6 Design Constraints | 49 |
| 3. Architectural Rationale | 51 |
| 3.1 Architecture Style and Rationale | 51 |
| 3.1.1 Architectural Overview | 51 |
| 3.1.2 Architectural Components | 51 |
| 3.1.3 Architecture Rationale | 52 |
| 4. Architectural Views | 54 |
| 4.1 Use Case View | 56 |
| 4.2 Development View | 57 |
| 4.3 Logical View | 64 |
| 4.4 Process View | 69 |
| Figure 4.14: Shows SD001 Guest Registration | 69 |
| Figure 4.15: Shows SD002 User Login | 70 |
| Figure 4.16: Shows SD003 User Manage Profile | 70 |
| Figure 4.17: Shows SD004 User View Dengue Map Data | 71 |

| | |
|---|------------|
| Figure 4.18: Shows SD005 User Report Breeding Sites | 71 |
| Figure 4.19: Shows SD006 User Take Quizzes And View Quizzes Result | 72 |
| Figure 4.20: Shows SD007 User View Educational Video | 72 |
| Figure 4.21: Shows SD008 Admin Login To Admin Dashboard | 73 |
| Figure 4.22: Shows SD009 Admin Verify User Reports | 74 |
| Figure 4.23: Shows SD010 Admin Update Dengue Map Data | 75 |
| Figure 4.24: Shows SD011 Admin Monitor System Logs | 76 |
| Figure 4.25: Shows SD012 Admin Manage Quizzes | 77 |
| 4.5 Physical View | 78 |
| 5. Data Design | 80 |
| 5.1 Data Dictionary | 81 |
| 5.1.1 Entity: <GUEST> | 81 |
| 5.1.2 Entity: <USER> | 81 |
| 5.1.3 Entity: <VIDEO> | 81 |
| 5.1.4 Entity: <DENGUE MAP> | 82 |
| 5.1.5 Entity: <QUIZ> | 82 |
| 5.1.6 Entity: <QUIZ RESULT> | 82 |
| 5.1.7 Entity: <ADMIN> | 83 |
| 5.1.8 Entity: <REPORT CASES> | 83 |
| 6. User Interface Design | 84 |
| 6.1 Guest Registration Subsystem Interface | 84 |
| 6.2 Customer Login Subsystem Interface | 87 |
| 6.3 Email Verification Subsystem Interface | 89 |
| 6.4 Customer Logout Subsystem Interface | 92 |
| 6.5 Viewing Homepage Subsystem Interface | 94 |
| 6.6 Viewing Menu Subsystem Interface | 95 |
| 6.7 Forgot Password Subsystem Interface | 96 |
| 6.8 Changing Password Subsystem | 98 |
| 6.9 Viewing Profile Subsystem Interface | 101 |
| 6.10 Editing Profile Subsystem Interface | 102 |
| 6.11 Admin Login Subsystem Interface | 103 |
| 6.12 Admin Logout Subsystem Interface | 105 |
| 6.13 Dengue Map Subsystem Interface | 107 |
| 6.14 Reporting Mosquito Breeding Sites Subsystem Interface | 109 |
| 6.15 Viewing Educational Videos Subsystem Interface | 111 |
| 6.16 Taking Quizzes Subsystem Interface | 112 |
| 6.17 Admin Managing Reports Subsystem Interface | 114 |
| 6.18 Admin Managing Quizzes Subsystem Interface | 117 |
| 7. Traceability | 119 |
| Table 7.1: Traceability | 119 |
| 8. Test Cases | 120 |
| 8.1 TC001: Test Package User Management Subsystem: Guest Registration (US001) | 120 |
| 8.2 TC002: Test Package User Management Subsystem: Customer Login (US002) | 122 |

| | |
|---|------------|
| 8.3 TC003: Test Package User Management Subsystem: Email Verification (US002) | 123 |
| 8.4 TC004: TestPackage User Management Subsystem: Customer Logout (US002) | 125 |
| 8.5 TC005: Test Package User Management Subsystem: Viewing Homepage (US002) | |
| 125 | |
| 8.6 TC006: Test Package User Management Subsystem: Viewing Menu (US002) | 126 |
| 8.7 TC007: Test Package User Management Subsystem: Forgot Password (US002) | 127 |
| 8.8 TC008: Test Package User Management Subsystem: Changing Password (US002) | 127 |
| 8.9 TC009: Test Package User Management Subsystem: Viewing Profile (US003) | 128 |
| 8.10 TC010: Test Package User Management Subsystem: Editing Profile (US003) | 129 |
| 8.11 TC011 : Test Package Admin Subsystem : Admin Login (US009) | 130 |
| 8.12 TC012 : Test Package Admin Subsystem : Admin Logout (US011) | 131 |
| 8.13 TC013: Test Package Dengue Information Subsystem : Dengue Map (US004) | 131 |
| 8.14 TC014: Test Package Dengue Information Subsystem : Reporting Mosquito Breeding Sites (US005) | 132 |
| 8.15 TC015: Test Package Learning Engagement Subsystem : Viewing Educational Videos (US008) | 133 |
| 8.16 TC016: Test Package Learning Engagement Subsystem : Taking Quizzes (US006) | |
| 134 | |
| 8.17 TC017 : Test Package Admin Subsystem : Admin Managing Reports (US0012) | 135 |
| 8.18 TC018: Test Package Admin Subsystem : Manage Quiz (US018) | 136 |
| 9. Traceability Matrix | 138 |
| Table 9.1: Traceability Matrix for <DPCM System> | 138 |
| Appendix | 143 |

1. Introduction

Our project is about A Dynamic Application for Dengue Awareness, Prevention, and Control, DPCM. This application is a dynamic, interactive application designed to promote dengue awareness, prevention, and control. This application also features engaging tools such as quizzes, real-time data integration and engaging learning resources to educate users of all ages. Key elements include dengue outbreak notifications to keep users informed of the latest outbreaks and trends, teaching resources that offer a variety of educational materials from prevention tips to understanding symptoms and treatments, and gamified challenges that allow users to participate in fun interactive activities to reinforce learning and improve awareness in a fun way.

1.1 Purpose

Dengue Prevention and Control in Malaysia (DPCM), with this system, we are aiming to make an engaging system to educate people, to raise awareness about dengue here in Malaysia by creating a comprehensive, user-friendly, and interactive platform. In our system, we will integrate real-time dengue situations in Malaysia, educational and instructional resources that are easy to access, and community engagement features to encourage individuals and communities to learn and prevent dengue effectively.

Key Features

1. Real-time Data Integration:
 - a. Real-time dengue fever situation mapping: Visualize dengue spreadness in Malaysia
 - b. Dengue epidemic notification: Receive notification about the dengue outbreak in our surrounding
2. Interactive Learning:
 - a. Gamified learning: create an engaging learning process in the system through interactive quizzes, challenges, etc.

- b. Educational Videos: Informative video for dengue prevention and control
- 3. Community Engagement Platform:
 - a. Social Features: Users can connect with each other and share experiences about dengue for a real experience education
 - b. Community Forum: Facilitate a forum discussing and sharing about dengue
 - c. Reporting system: users can report potential breeding sites or dengue cases in their area and contribute to real-time data in the system

1.2 Scope

The scope for DPCM project aims are to create an interactive, user-friendly dynamic application designed to promote awareness, prevention, and control of dengue fever. This website will feature education resources, real-time epidemic updates, and engaging tools such as quizzes, etc. This project includes the design, development, testing, deployment, and maintenance of the website, ensuring it is accessible and effective for users of all ages.

1.3 Definitions, Acronyms and Abbreviation

DPCM - Dengue Prevention and Control in Malaysia

SD - System Documentation

NFR - Non-functional requirements

UX Designer - User Experience Designer

1.4 References

Blog - Draw a UML state machine diagram. (n.d.).

<https://www.drawio.com/blog/uml-state-diagrams>

Use Case Diagram For Purchase Order Processing and Management [classic] | Creately.

(n.d.).

<https://creately.com/diagram/example/ikoychpw/use-case-diagram-for-purchase-order-processing-and-management-classic>

Editor. (2023, December 30). Nonfunctional Requirements in Software Engineering:

Examples, Types, Best Practices. *AltexSoft*.

<https://www.altexsoft.com/blog/non-functional-requirements/>

1.5 Overview

This Software Design Document (SD) describes the design and functionality of the Dengue Prevention and Control in Malaysia (DPCM) system, providing a defined framework for development and testing.

1. **Organisation of the Document:** The introduction provides an overview of the project's aim, features, and target users. The purpose of this document is to describe the DPCM system's aims for raising awareness and promoting dengue prevention.
2. **Scope:** Determines the project's bounds, including features and target consumers.
3. **Definitions, Acronyms, and Abbreviations:** Clarifies essential terminology and abbreviations used throughout the work.
4. **References:** A list of publications and standards used throughout development.

5. **System Overview:** A summary of the main components, functionality, and user responsibilities.
6. **Specific Requirements:** Use cases and user stories provide detailed functional and non-functional requirements.
7. **System Design:** Consists of use case diagrams, domain models, and workflow diagrams.
8. **Product Backlog and Sprints:** Outlines the iterative development process and user stories for each sprint.
9. **Non Functional Need:** Include performance, security, and compatibility.
10. **Design Constraint:** Include environmental, hardware, security, and performance considerations.

The SD document serves as a systematic guide to the DPCM system, outlining the project's goal, requirements, design, and restrictions, providing clear comprehension and successful development.

2. Specific Requirements

This section contains all the software requirements that will be used for the Dengue Prevention and Control system in Malaysia where all the requirements will be explained in detail to make it easier for designers to make a neater design and can attract users to see this dengue application. In addition, it can make it easier for testers to test the Dengue Prevention and Control system in Malaysia because if the tester gives satisfactory results, then the Dengue Prevention and Control system in Malaysia is very suitable and user-friendly for other users. So this section will explain every need of users who will use this system.

2.1 Persona

Each persona represents a type of user interacting with the system, emphasizing their specific goals, needs, and interactions.

2.1.1 Persona 1: User (Registered User)

- User Need:**

Registered users need access to full functionality, including personalized notifications, real-time dengue information, and interactive learning features.

- User Stories:**

- As a user, I want to log in to the app so that I can access personalized features such as notifications and learning progress tracking.
- As a user, I want to view a real-time dengue map so that I can identify high-risk areas.
- As a user, I want to complete quizzes and track my progress to learn about dengue prevention in an engaging way.
- As a user, I want to report potential mosquito breeding sites so that I can contribute to dengue prevention efforts in my community.

2.1.2 Persona 2: Guest

- **User Need:**

Guests need to register as a user to get full access on the system.

- **User Stories:**

- As a guest, I want to register as an authorized user so that I can get full access to the system features.

2.1.3 Persona 3: Admin

- **User Need:**

Admins need tools to manage system data, review user reports, and maintain the system's integrity and accuracy.

- **User Stories:**

- As an admin, I want to log in to an admin dashboard so that I can manage system data securely.
 - As an admin, I want to verify and approve user-reported mosquito breeding sites to ensure accurate data.
 - As an admin, I want to update real-time dengue case statistics to provide users with the most accurate and timely information.
 - As an admin, I want to monitor system logs so that I can ensure the app's security and performance.

2.2 System Features

This section will explain the features of the system found in Dengue Prevention and Control in Malaysia. Among the features found in Dengue Prevention and Control in Malaysia is a security feature where users need to register their information and perform verification when using the Dengue Prevention and Control in Malaysia application to maintain and store user data. In addition, Dengue Prevention and Control in Malaysia also has a real time data feature where users can see the latest dengue status according to their area and users can find out the rate of increase in dengue cases in Malaysia. Next, the Dengue Prevention and Control in Malaysia website also has interactive learning where users can play several mini games related to dengue information and users can also watch a short video about the explanation of dengue fever. So this can make it easier for users to understand faster and users don't feel bored. Below we include a use case diagram regarding our Dengue Prevention and Control in Malaysia system and a description of the functions in this system.

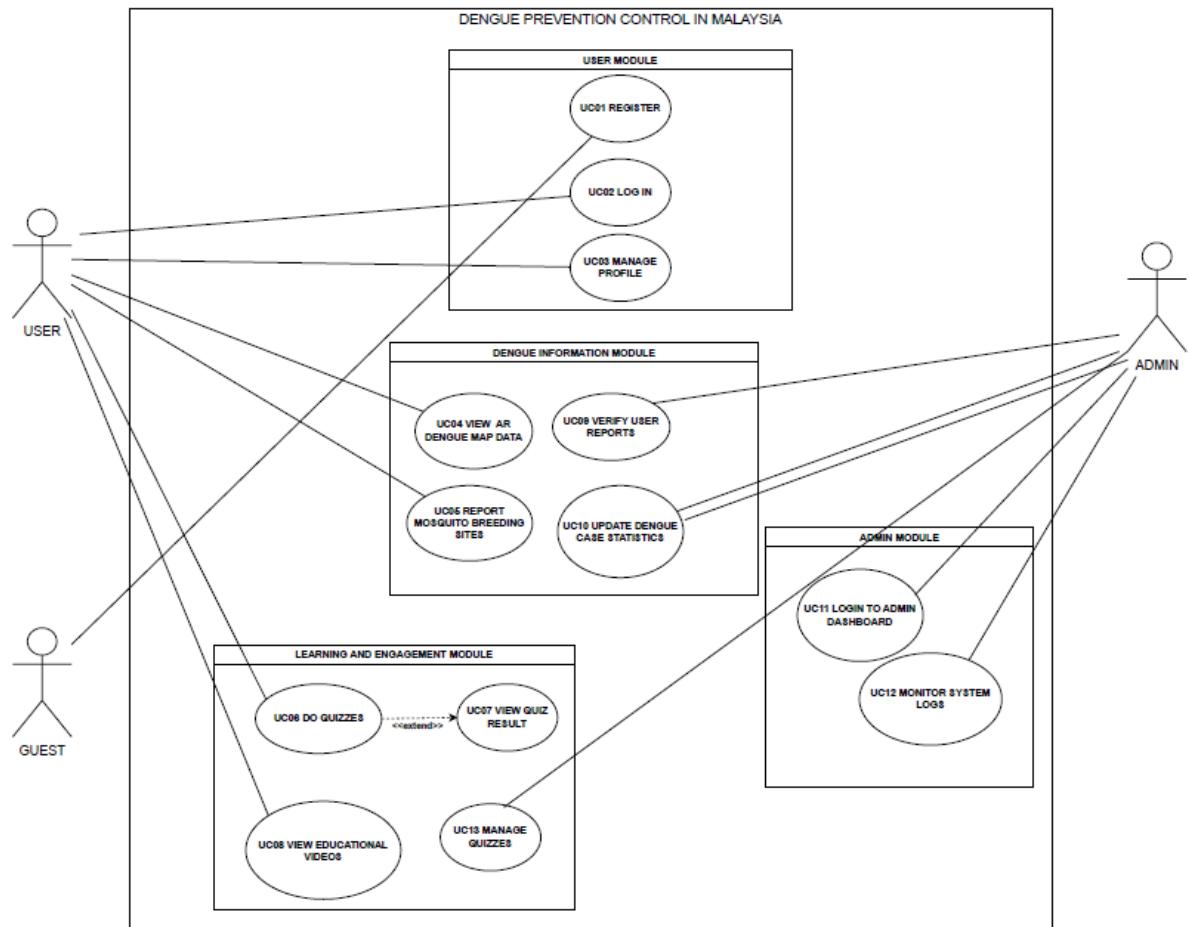


Figure 2.1: Use Case Diagram for DPCM SYSTEM

Table 2.1: Description of Module and Functions for DPCM SYSTEM

| Module | Function | Description |
|--------------------------------|---------------------------------------|--|
| User Module | UC01-Register | This use case allows guests to sign up as a user for the application. |
| User Module | UC02-Login | This use case allows registered users to log in to the application. |
| User Module | UC03-Manage Profile | This use case allows users to view and edit their personal profile information. |
| Dengue Information Module | UC04-View Dengue Map Data | This use case allows users to view Dengue Map Data. |
| Dengue Information Module | UC05-Report Mosquitoes Breeding Sites | This use case allows users to report potential mosquito breeding sites in their area. |
| Learning and Engagement Module | UC06-Do Quizzes | This use case allows users to take quizzes to test their knowledge about dengue. |
| Learning and Engagement Module | UC07-View Quizzes Result | This use case allows users to view their scores on quizzes. |
| Learning and Engagement Module | UC08-View Educational Videos | This use case allows users to access educational videos about dengue prevention and control. |
| Dengue Information Module | UC09-Verify User Reports | This use case allows administrators to verify and approve or reject user-reported mosquito breeding sites. |
| Dengue Information Module | UC10-Update Dengues Case Statistics | This use case allows administrators to update real-time dengue case statistics. |
| Admin Module | UC11-Login to Admin Dashboard | This use case allows administrators to log in to the admin dashboard. |
| Admin Module | UC12-Monitor System Logs | This use case allows administrators to monitor system logs to identify and resolve issues. |
| Admin Module | UC13 - Manage Quizzes | This use case allows administrators to create, update, and delete quizzes to ensure up-to-date and accurate content for users. |

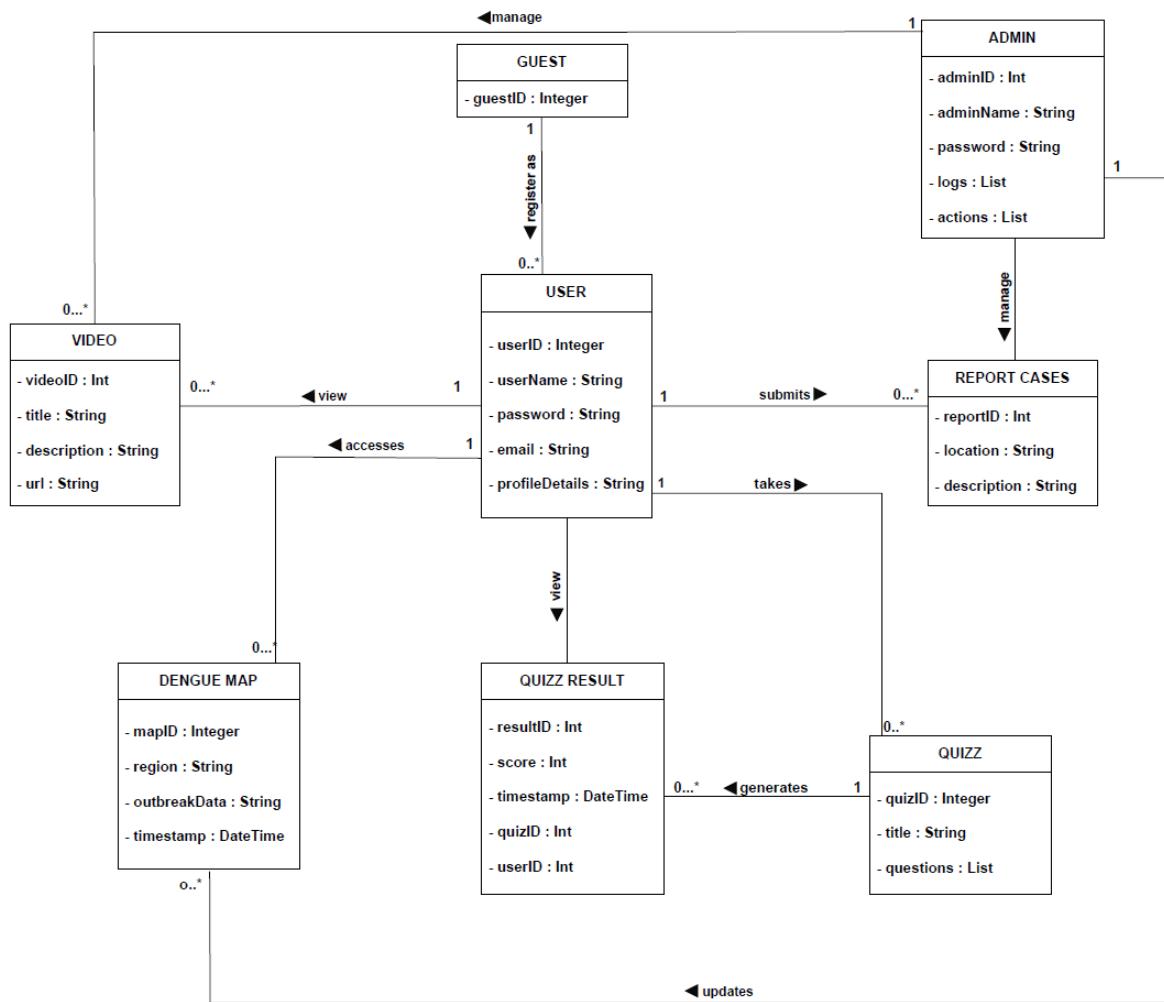


Figure 2.2: Domain Model for DPCM SYSTEM

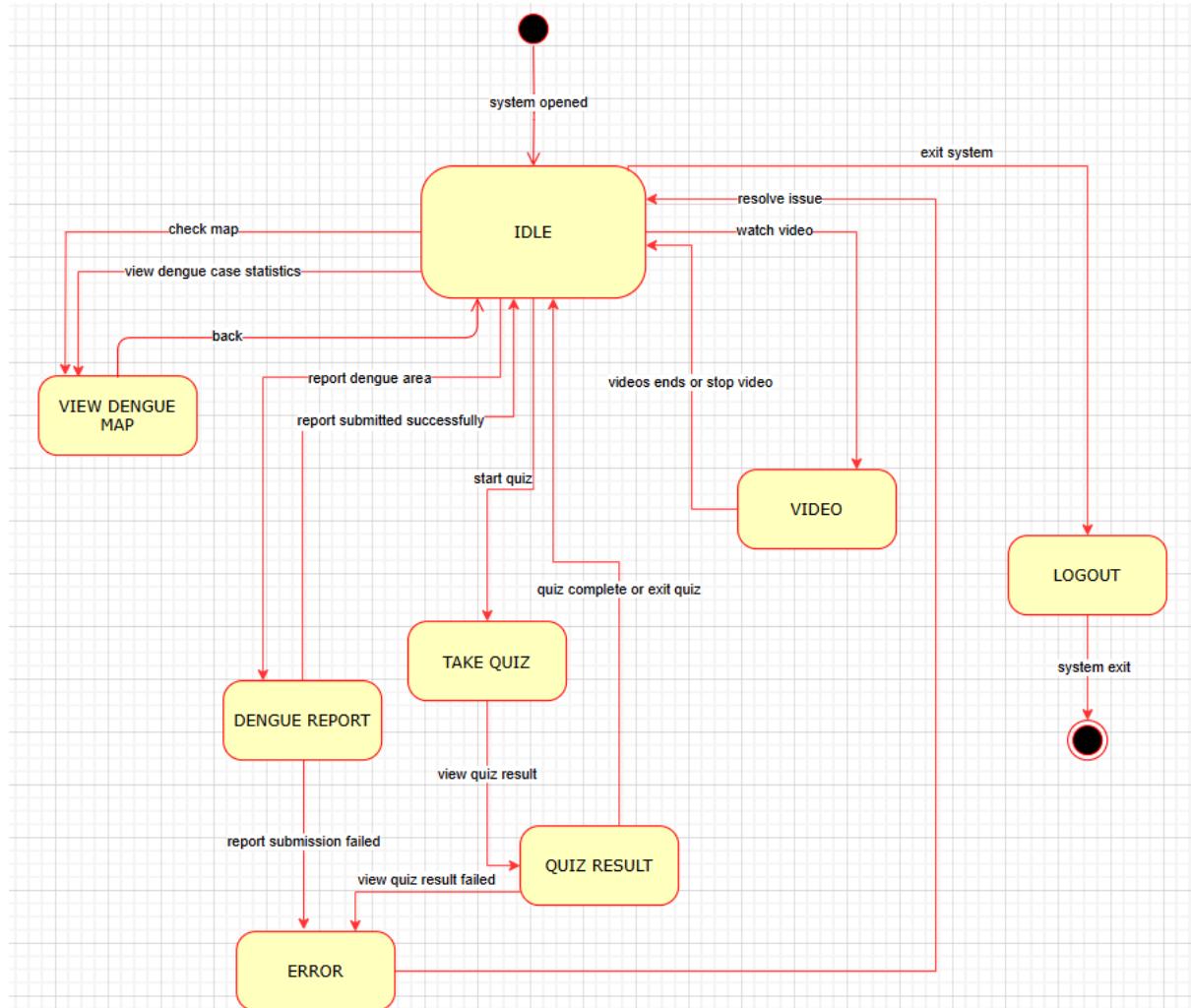


Figure 2.3: State Machine Diagram for DPCM SYSTEM

2.3 Launch Phase

| Sprint | Team member assigned |
|---|---|
| <p>Sprint #1 (2 Week)</p> <p>User Module</p> <p>-UC01-Register</p> <ul style="list-style-type: none"> • Design and implement a user registration form (username, password, email, etc.). • Verify user inputs (e.g., password strength, email format). • Store user data securely in the database. • Integrate email verification for account activation. • Create error handling for failed registration attempts. • Notify the user of successful registration <p>-UC02-Login</p> <ul style="list-style-type: none"> • Create a login form (username/email and password). • Implement authentication logic. • Support "Remember Me" functionality. • Add account lockout for multiple failed login attempts. • Provide a "Forgot Password" feature (password recovery). • Secure session handling (e.g., session timeout, secure cookies). <p>-UC03-Manage Profile</p> <ul style="list-style-type: none"> • Build a user profile view page. • Develop functionality for editing personal information (e.g., name, email, profile picture). • Validate updated profile data. • Implement password update/change feature. • Integrate with storage for profile picture uploads (if applicable). • Add notifications for successful profile updates. • Ensure secure handling of sensitive data. | 1. Project manager 2. Scrum Master 3. Product Owner 4. Developer 5. UX designer 6. Quality Assurance |

| | |
|--|--|
| <p>Sprint #2 (2 Week)</p> <p>Dengue Information Module</p> <p>-UC04-View Dengue Map Data</p> <ul style="list-style-type: none"> ● Integrate a map API (e.g., Google Maps, Leaflet) for displaying data. ● Design a user-friendly interface for the dengue map. ● Fetch and display dengue hotspot data from the database or an external API. ● Implement filters (e.g., date range, severity level, location). ● Include map legends and tooltips to explain data points. ● Optimize performance for real-time map updates. ● Add accessibility features (e.g., descriptive labels, screen reader compatibility). <p>-UC05-Report Mosquitoes Breeding Sites</p> <ul style="list-style-type: none"> ● Create a form for users to report breeding sites (e.g., location, description, photo upload). ● Implement a geolocation feature to allow users to pinpoint exact locations. ● Validate submitted reports (e.g., required fields, appropriate file uploads). ● Save reports in the database with proper categorization (e.g., urgent, under review). ● Notify the user of successful submission. ● Provide admins/moderators with tools to review and act on reports. ● Include feedback mechanisms to inform users about the status of their report (e.g., resolved, in progress). | 1.Project manager 2. Scrum Master 3.product Owner 4.Developer 5.UX designer 6.Quality Assurance |
| <p>Sprint #3</p> <p>Learning and Engagement Module</p> <p>-UC06-Do Quizzes</p> <ul style="list-style-type: none"> ● Design and develop a quiz interface for users. ● Create a database to store quiz questions, answers, and categories. | 1.Project manager 2. Scrum Master 3.product Owner 4.Developer 5.UX designer 6.Quality Assurance |

| | |
|---|--|
| <ul style="list-style-type: none"> • Implement randomization or topic-based selection of quiz questions. • Include a timer for time-limited quizzes (optional). • Provide immediate feedback for each question or overall feedback after the quiz. • Add scoring logic to calculate results. • Notify users of quiz completion and encourage retry if they score below a certain threshold. • Ensure accessibility for all users, including those with disabilities. <p>-UC07-View Quizzes Result</p> <ul style="list-style-type: none"> • Create a page or dashboard to display quiz results. • Store historical quiz scores in the database for user tracking. • Include statistics like average score, best score, and number of attempts. • Allow users to view correct answers for quizzes they've completed. • Implement motivational features (e.g., badges or progress indicators for high scores or improvement). <p>-UC08-View Educational Videos</p> <ul style="list-style-type: none"> • Design a video library interface with categories and search functionality. • Integrate video hosting platforms (e.g., YouTube, Vimeo) or embed locally hosted videos. • Include a description and tags for each video to aid user understanding. • Track video completion status to encourage users to watch all content. • Implement video playback controls (e.g., pause, rewind, subtitles). • Add a "favorite" or "watch later" feature for users to save videos. • Ensure all videos have captions or transcripts for accessibility compliance. | |
| <p><u>Sprint #4</u></p> <p>Dengue Information Module</p> <p>-UC09: Verify User Reports</p> <ul style="list-style-type: none"> • Develop an admin dashboard to list all user-reported mosquito breeding sites. | <ol style="list-style-type: none"> 1.Project manager 2. Scrum Master 3.product Owner 4.Developer 5.UX designer 6.Quality Assurance |

| | |
|---|--|
| <ul style="list-style-type: none"> ● Include functionality to view detailed reports (location, description, photos). ● Add options for administrators to approve, reject, or flag reports for further investigation. ● Notify users about the status of their reports (approved/rejected flagged). ● Implement filters and search options for admins to quickly find reports based on criteria (e.g., date, urgency). ● Maintain a log for all actions taken on reports for auditing purposes. ● Ensure secure access control for admin functionality. <p>-UC10: Update Dengue Case Statistics</p> <ul style="list-style-type: none"> ● Create an interface for administrators to update dengue case statistics (e.g., new cases, recoveries, deaths). ● Integrate data visualization tools (e.g., charts, graphs) for real-time statistics. ● Enable batch uploads for large data sets (e.g., CSV or Excel imports). ● Provide options to edit or delete previous updates. ● Ensure the data is displayed dynamically on the Dengue Map (UC04). ● Track and log changes to ensure transparency and accountability. ● Implement role-based access control to restrict editing permissions. | |
| <p><u>Sprint #5</u></p> <p>Admin Module</p> <p>-UC11: Login to Admin Dashboard</p> <ul style="list-style-type: none"> ● Develop an admin login page with secure authentication mechanisms (e.g., multi-factor authentication, CAPTCHA). ● Implement role-based access control to ensure only authorized administrators can access the dashboard. ● Ensure secure session handling, including timeout and logout features. ● Log all login attempts (successful and failed) for auditing and security | <ol style="list-style-type: none"> 1.Project manager 2. Scrum Master 3.product Owner 4.Developer 5.UX designer 6.Quality Assurance |

- purposes.
- Provide an option for password recovery or reset.

-UC12: Monitor System Logs

- Create a system logs monitoring interface within the admin dashboard.
- Display logs for key activities (e.g., user registrations, login attempts, report submissions, data updates).
- Include filters for date, user, or activity type for easier navigation.
- Add real-time alerts for suspicious activities (e.g., multiple failed login attempts, unexpected data changes).
- Enable administrators to export logs for detailed analysis.
- Implement secure storage and access to logs, ensuring compliance with privacy and security standards.

-UC13: Manage Quiz

- Add new quizzes with a title, description, and questions (e.g., multiple-choice, true/false).
- Update quiz details, including questions and answers
- Remove outdated quizzes with confirmation prompts
- Monitor user participation and quiz score

2.4 User Story Details

User story details describe each user story that will use the Dengue Prevention and Control website in Malaysia where we will describe twelve user stories about how users will use this system. So each user story has their own ID and this user story is also based on the use case diagram we created. Below we will show each user story along with the ID.

2.4.1 US01: User Story <Guest>

Table 2.4.1.1: User Story Description for <Registration>

| User story: <Registration> |
|---|
| ID: US01 |
| User Story Description As a guest application DPCM I want to sign up as user for application So that i can use the application |
| Flow of events: 1. User sign up using google account or manual 2. User fill their name and password 3. User submit the detail |
| Alternative flow n: |
| Acceptance Criteria Precondition - The user must have access to the registration page Postcondition - The user receives a confirmation message or email. |
| Exception flow: If the user doesn't not fill in a blank field that was mandatory in the registration form, when the user clicks on the confirm button the system will display pop up message that shows "Please fill all information in the form" and will stay at the registration page. |

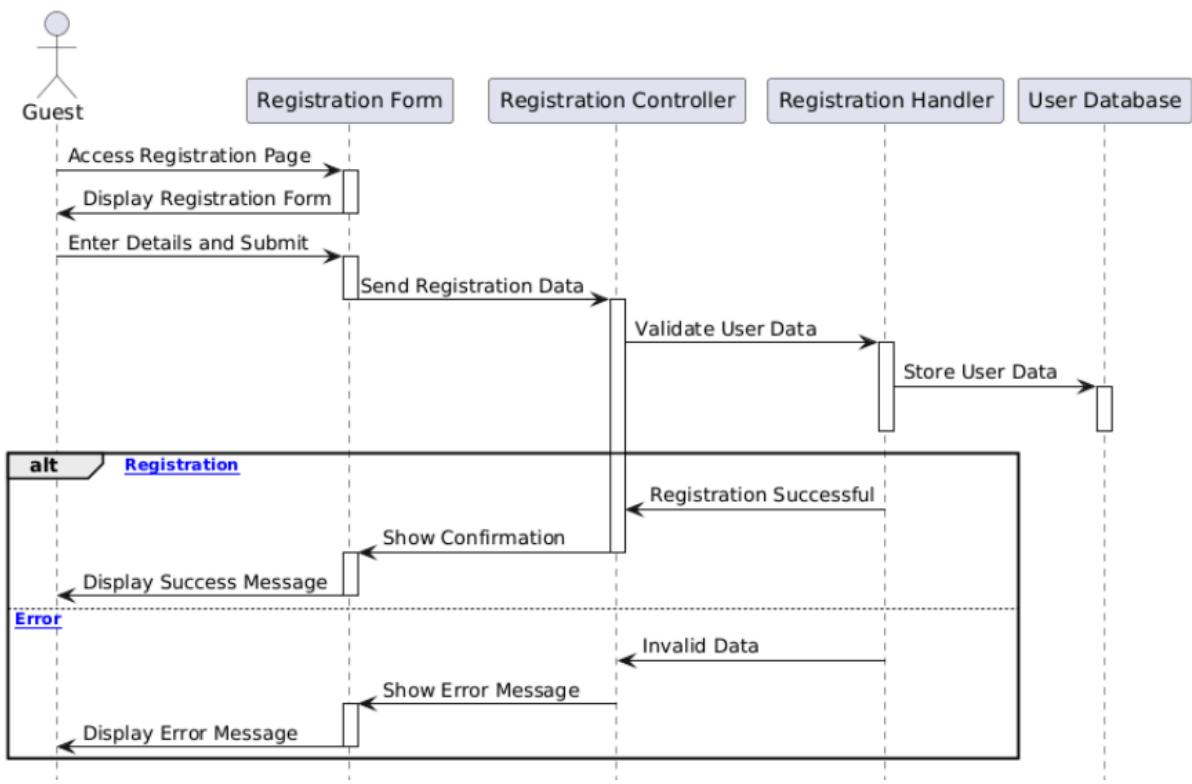


Figure 2.4.1.1 : Sequence Diagram for <Registration>

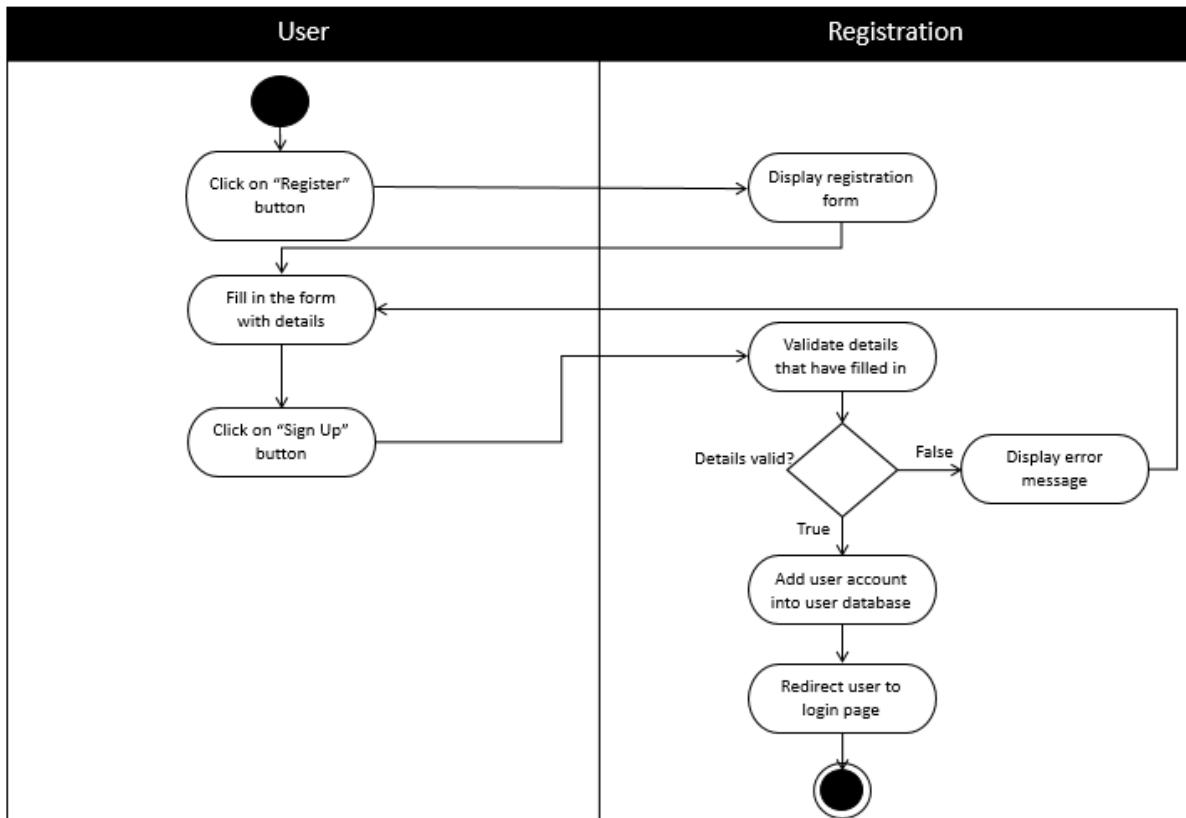


Figure 2.4.1.2 : Activity Diagram for <Registration>

2.4.2 US02: User Story <User>

Table 2.4.2.1: User Story Description for <Login>

| User story: <Login> |
|--|
| ID: US02 |
| User Story Description |
| As a user application DPCM I want to login as user for application So that i can use the application |
| Flow of events: <ol style="list-style-type: none"> 1. User login using google account or manual 2. User fill their name and password 3. User login the website |
| Alternative flow n: |
| Acceptance Criteria Precondition - A valid User is logged on to the system. Postcondition - Users can browse through all features in the system. |
| Exception flow: If the user enters the wrong username or password and clicks the login button, the system will display a pop message “Username or Password Invalid, Please Try Again.” and it will stay at the login page. |

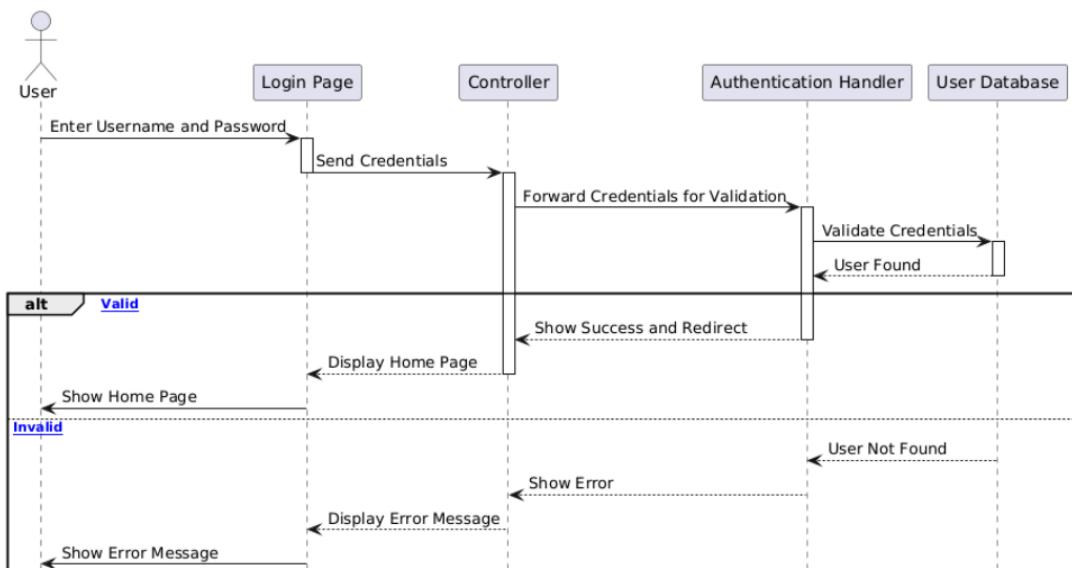


Figure 2.4.2.1.1 : Sequence Diagram for <Login>

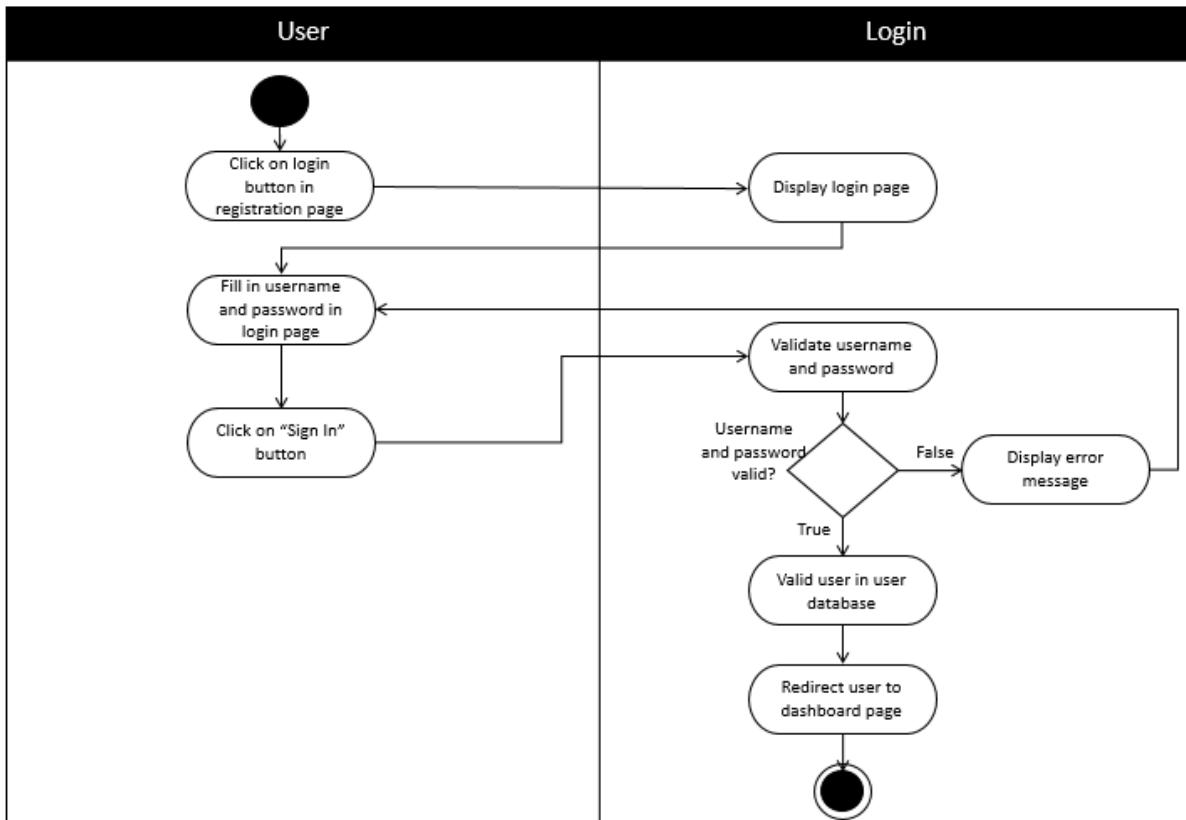


Figure 2.4.2.1.2 : Activity Diagram for <Login>

Table 2.4.2.2: User Story Description for <Manage Profile>

| User story: <Manage Profile> |
|--|
| ID: US03 |
| User Story Description |
| As a user application DPCM I want to manage profile as user in application So that i can view and edit user profile |
| Flow of events: <ol style="list-style-type: none"> 1. User login the application 2. User go to setting and click user profile 3. User view and edit their personal information 4. User save profile |
| Alternative flow n: User views their profile but does not make any changes. |
| Acceptance Criteria Postcondition - The updated user profile information is stored in the system database. precondition - The user has successfully logged into the application. |

Exception flow:

If the user leaves mandatory fields empty, the system displays a warning message: "Please fill in all required fields."

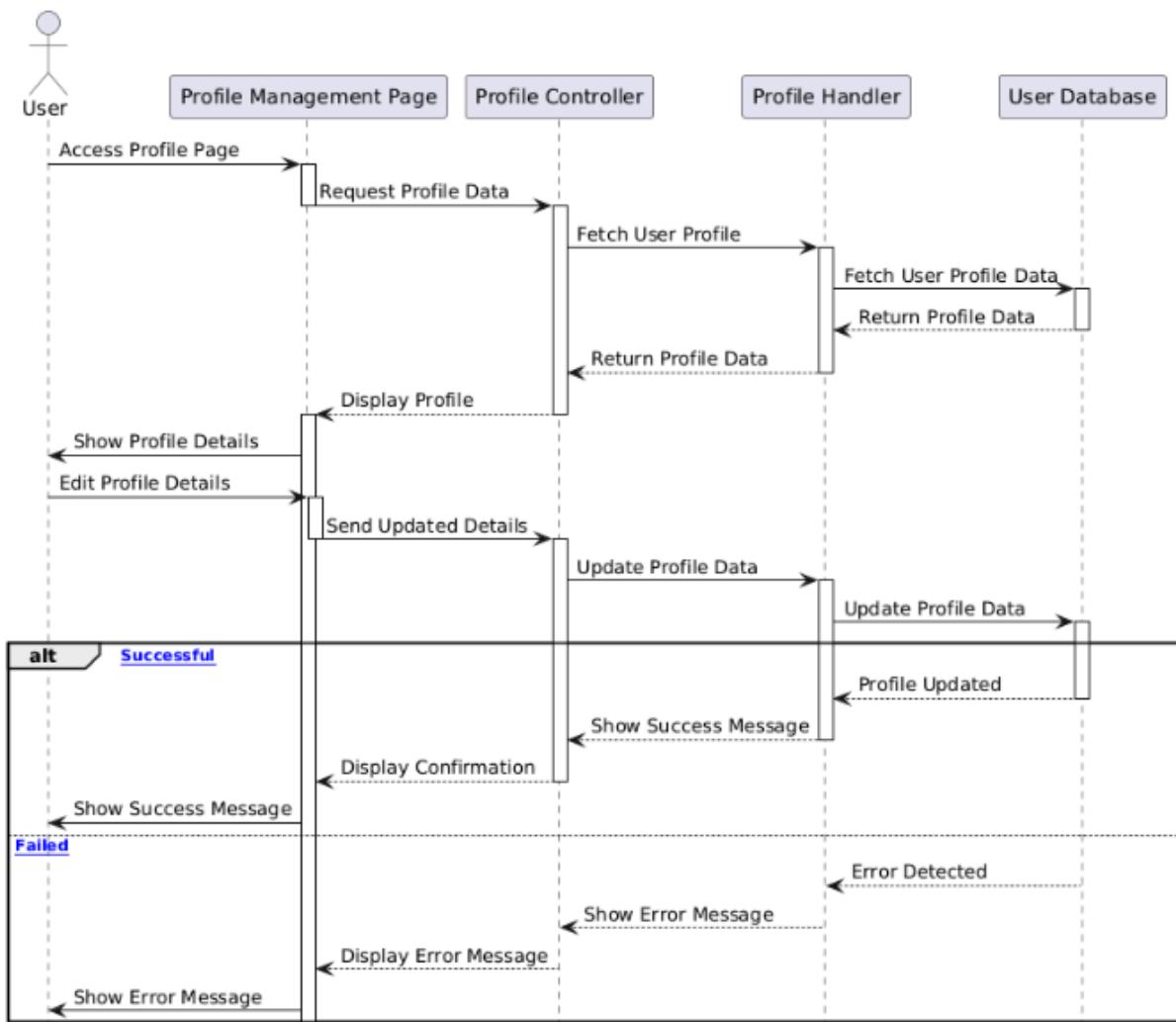


Figure 2.4.2.2.1 : Sequence Diagram for <Login>

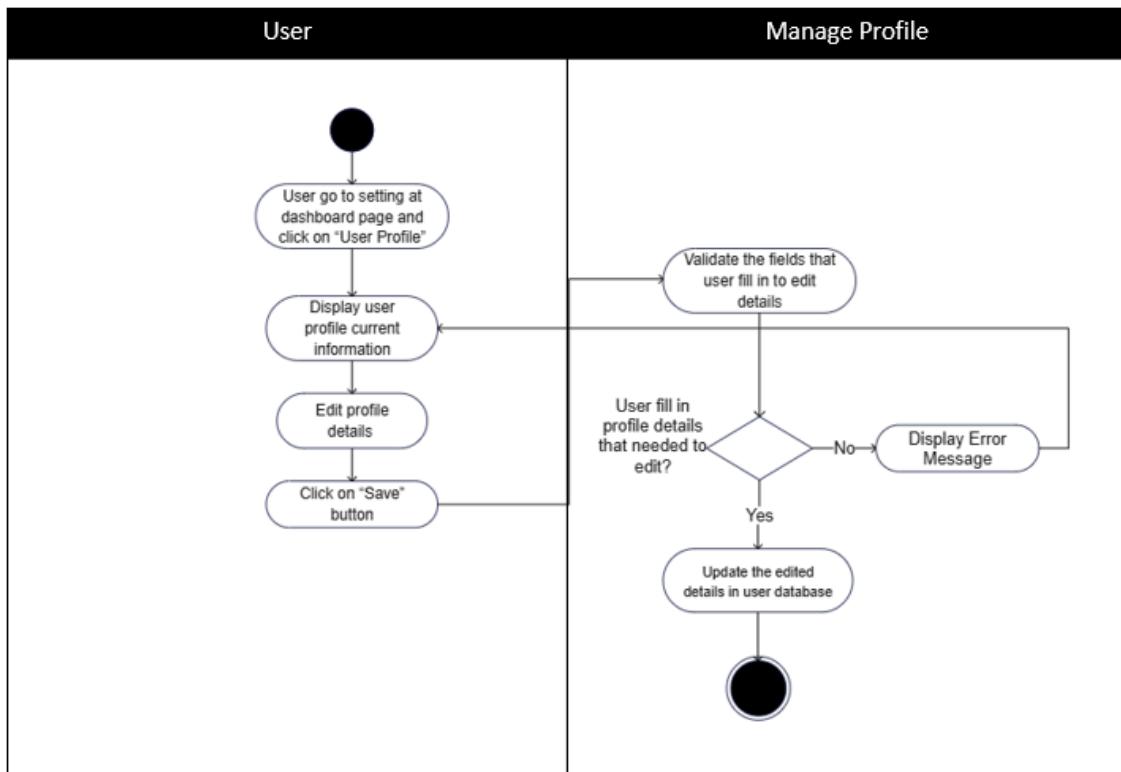


Figure 2.4.2.2.2 : Activity Diagram for <Login>

Table 2.4.2.3: User Story Description for <View Dengue Map Data>

| User story: <View Dengue Map Data> | |
|--|--|
| ID: US04 | |
| User Story Description | <p>As a user application DPCM I would like to see about dengue data So that i can know the latest dengue data</p> |
| Flow of events: | <ol style="list-style-type: none"> 1. User login the application 2. User see the main page 3. User click dengue map data 4. User can view latest data in map |
| Alternative flow n: | |
| Acceptance Criteria | |
| Postcondition - The user is able to view the latest dengue data on the interactive map. | |
| Precondition - The user has successfully logged into the application. | |

Exception flow:

If the map fails to render correctly due to system or browser issues, execution displays a message: "There was an error displaying the map. Please refresh the page or try again later."

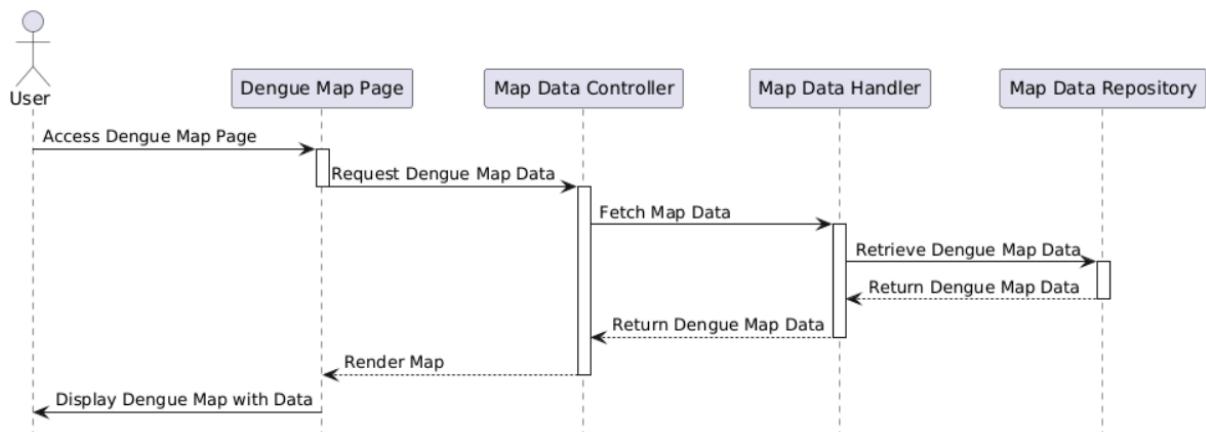


Figure 2.4.2.3.1 : Sequence Diagram for <View Dengue Map Data>

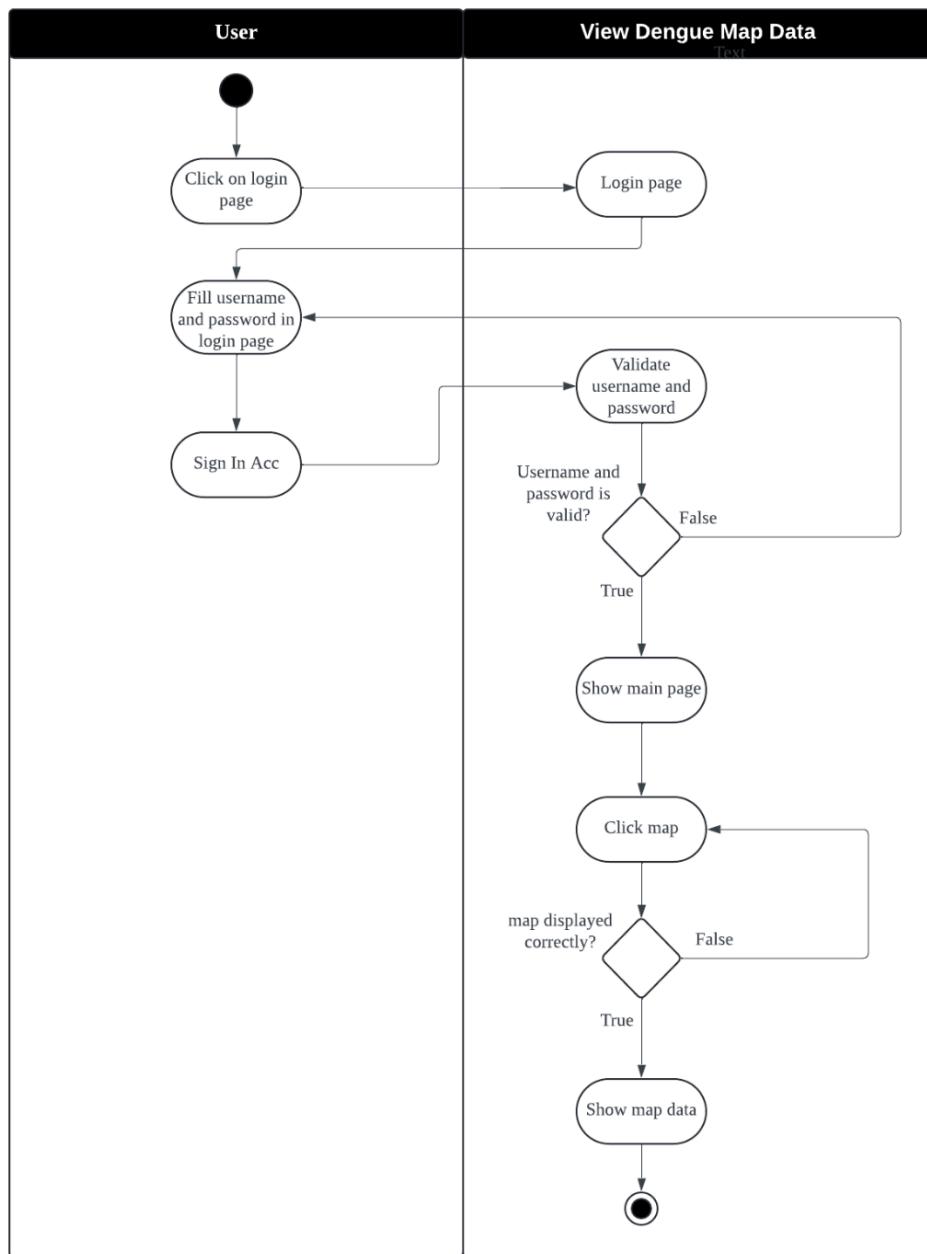


Figure 2.4.2.3.2 : Activity Diagram for <View Dengue Map Data>

Table 2.4.2.4: User Story Description for <Report Mosquitoes Breeding Sites>

| User story: <Report Mosquitoes Breeding Sites> | |
|--|--|
| ID: US05 | |
| User Story Description | |
| <p>As a user application DPCM I want to report about mosquitoes breeding sites So that application can update data from user</p> | |
| Flow of events: | |
| <ol style="list-style-type: none"> 1. User login the application 2. User go to page feedback customer 3. User fill their report in that page 4. User submit | |
| Alternative flow n: | |
| <p>Acceptance Criteria Postcondition - The user be able to report in the feedback page Precondition - The user has logged into application</p> | |
| Exception flow: | |
| If the user did not write anything or empty the report form and misclick or accidentally push the submit button, the system will show an error message “Please fill the report form”. And if there is an error in database and the user can’t submit the form, the system will show a message “Can’t proceed the request, try again later” | |

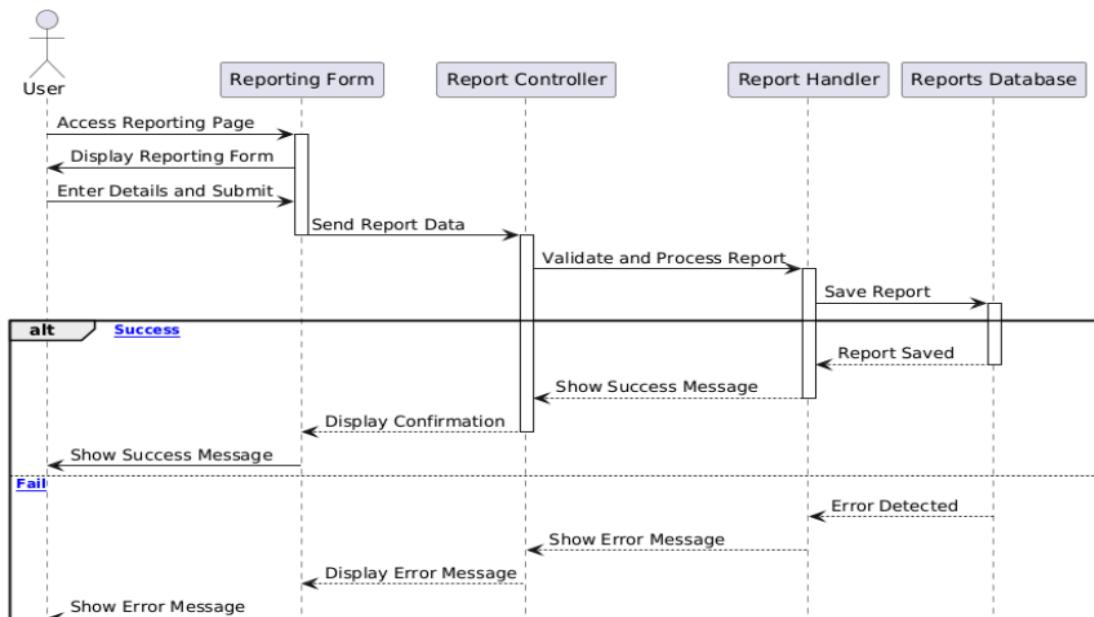


Figure 2.4.2.4.1 : Sequence Diagram for <Report Mosquitoes Breeding Sites>

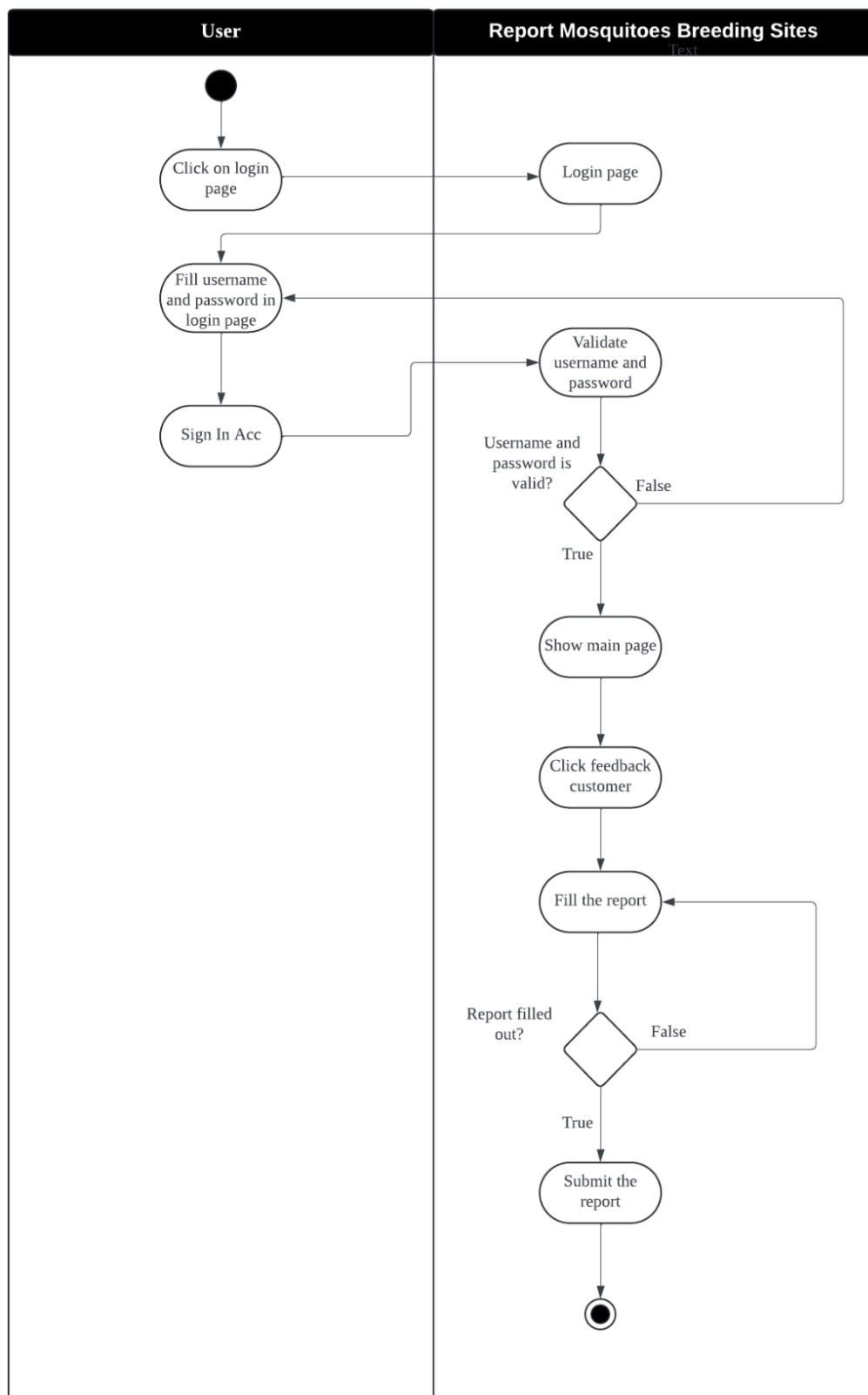


Figure 2.4.2.4.2 : Activity Diagram for <Report Mosquitoes Breeding Sites>

Table 2.4.2.5: User Story Description for <Quizzes>

| User story: <Quizzes> | |
|---|--|
| ID: US06 | |
| User Story Description | |
| As a user application DPCM I want to take quizzes So that i can test my knowledge about dengue | |
| Flow of events: | |
| <ol style="list-style-type: none"> 1. User login the application 2. User go to page mini game learning 3. User can play the quiz 4. User submit their answer | |
| Alternative flow n: | |
| Acceptance Criteria | |
| Postcondition - The user allowed to play mini game in application Precondition - The user has logged into application | |
| Exception flow: | |
| If there are many requests from users, the system may go down due to high traffic, and the quiz page may not appear or will take a long time to load. And if there is an error when the user wants to submit, the system will display a message “Cannot proceed the request, please try again”. | |

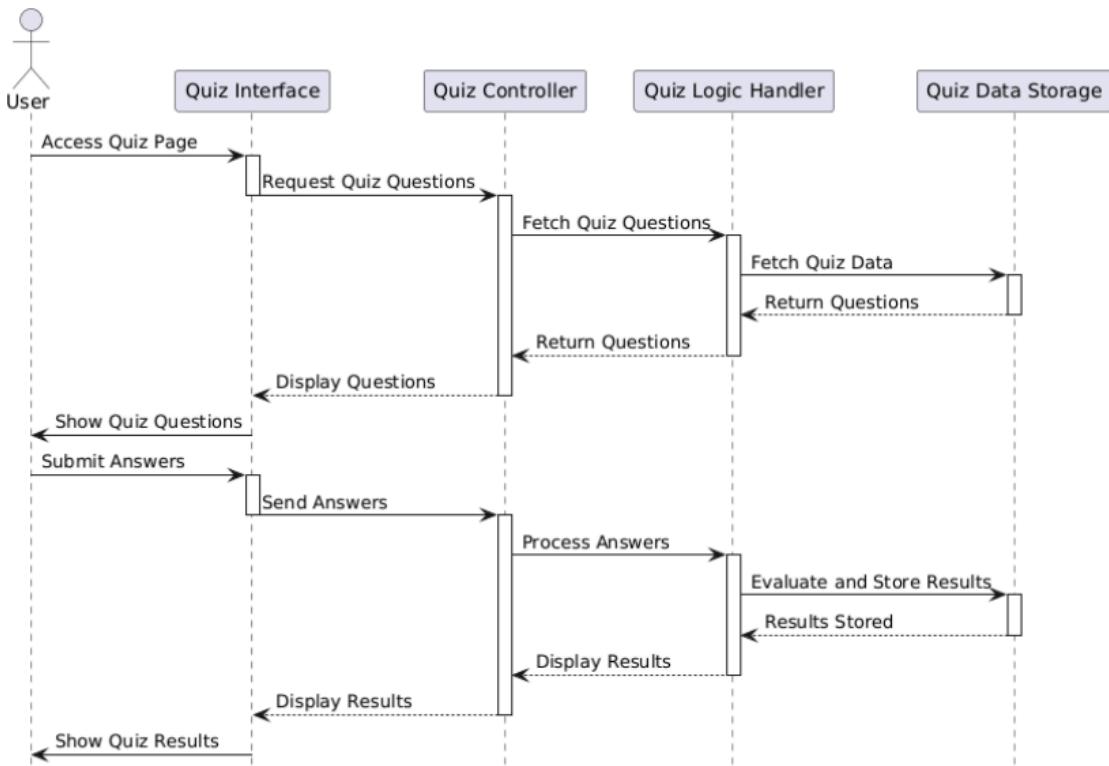


Figure 2.4.2.5.1 : Sequence Diagram for <Quizzes>

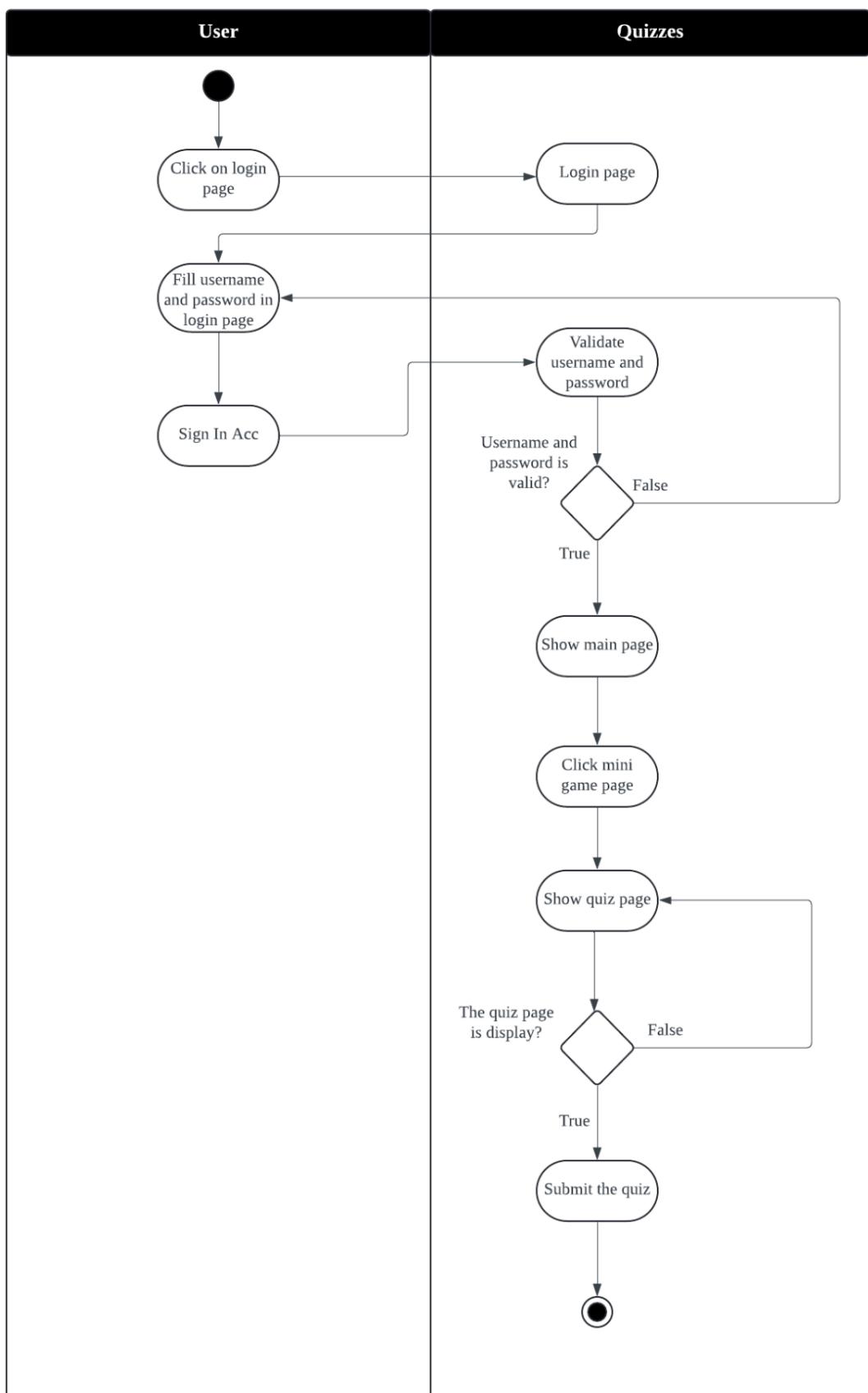


Figure 2.4.2.5.2 : Activity Diagram for <Quizzes>

Table 2.4.2.6: User Story Description for <View Quizzes Result>

| User story: <View Quizzes Result> |
|--|
| ID: US07 |
| User Story Description |
| <p>As a user application DPCM I want to see the result for quizzes So that i can know the score for quizzes</p> |
| <p>Flow of events:</p> <ol style="list-style-type: none"> 1. User login the application 2. User answer the question in quizzes 3. User submit the answer 4. User see the result for quizzes |
| Alternative flow n: |
| <p>Acceptance Criteria Postcondition - The user can see the result after submit the quiz Precondition - The user has logged into application</p> |
| <p>Exception flow: If there is an error when the user wants to see the result (not appear), the system will display a message “Cannot proceed the request, please try again later”.</p> |

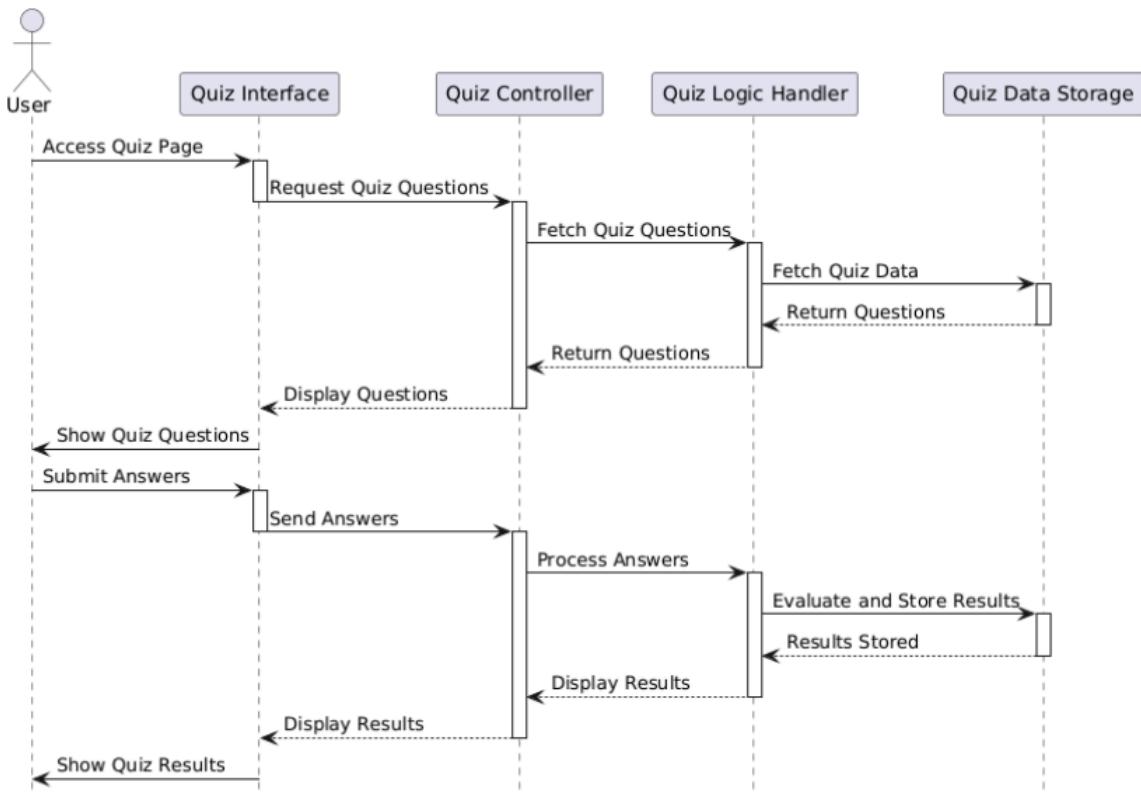


Figure 2.4.2.6.1 : Sequence Diagram for <View Quizzes Result>

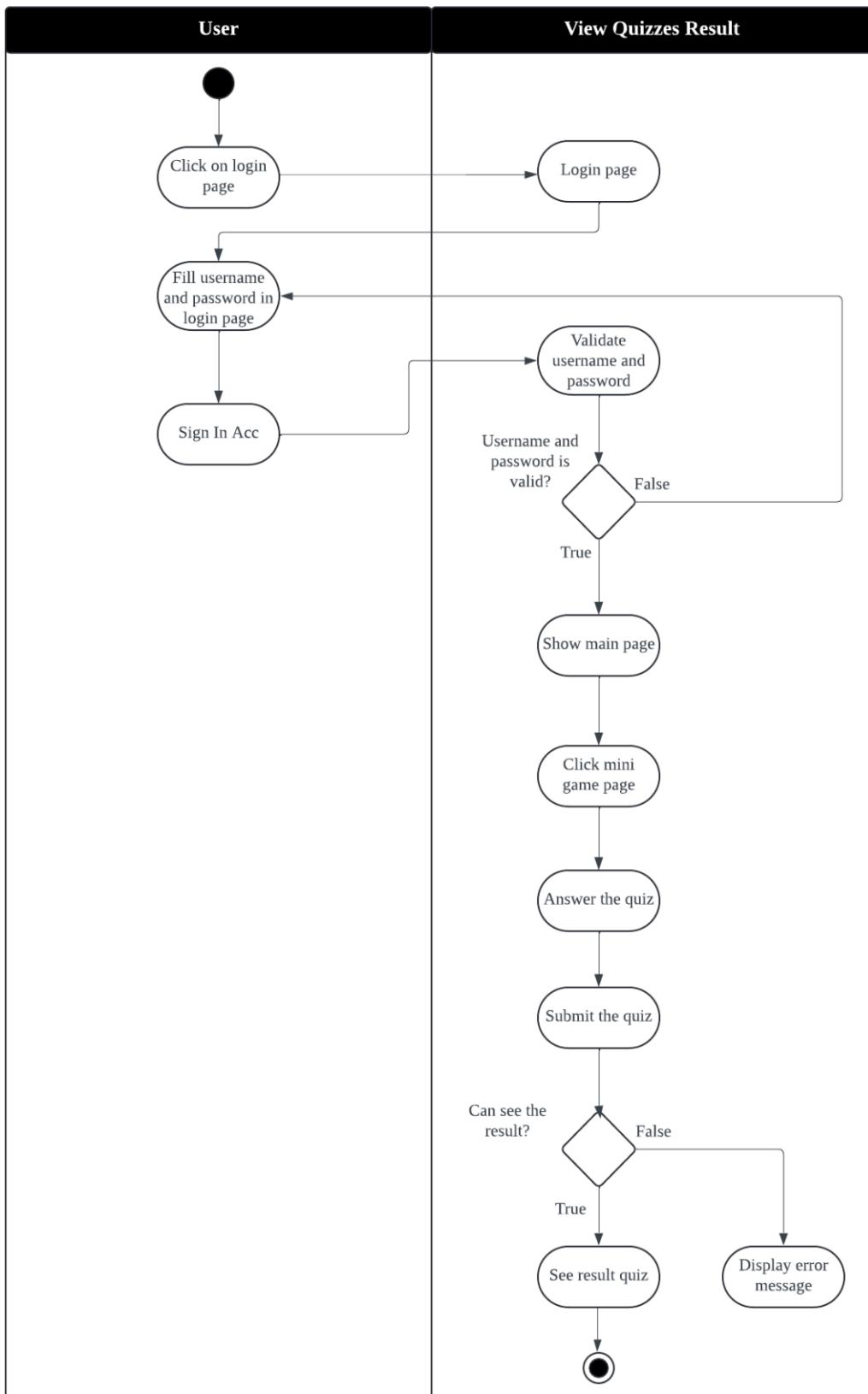


Figure 2.4.2.6.2 : Activity Diagram for <View Quizzes Result>

Table 2.4.2.7: User Story Description for <View Educational Videos>

| User story: <View Educational Videos> | |
|--|--|
| ID: US08 | |
| User Story Description | |
| As a user application DPCM I want to see videos about dengue so that I understand more about dengue fever and dengue prevention | |
| Flow of events: | |
| <ol style="list-style-type: none"> 1. User login the application 2. User go to page mini game learning 3. User click the video about dengue fever and dengue prevention | |
| Alternative flow n: | |
| Acceptance Criteria | |
| Postcondition - User be able to watch the video Precondition - The user has logged into application | |
| Exception flow: If there are high traffic in the system, may the video will take a long time to load, or it will send an error message “Try again later” | |

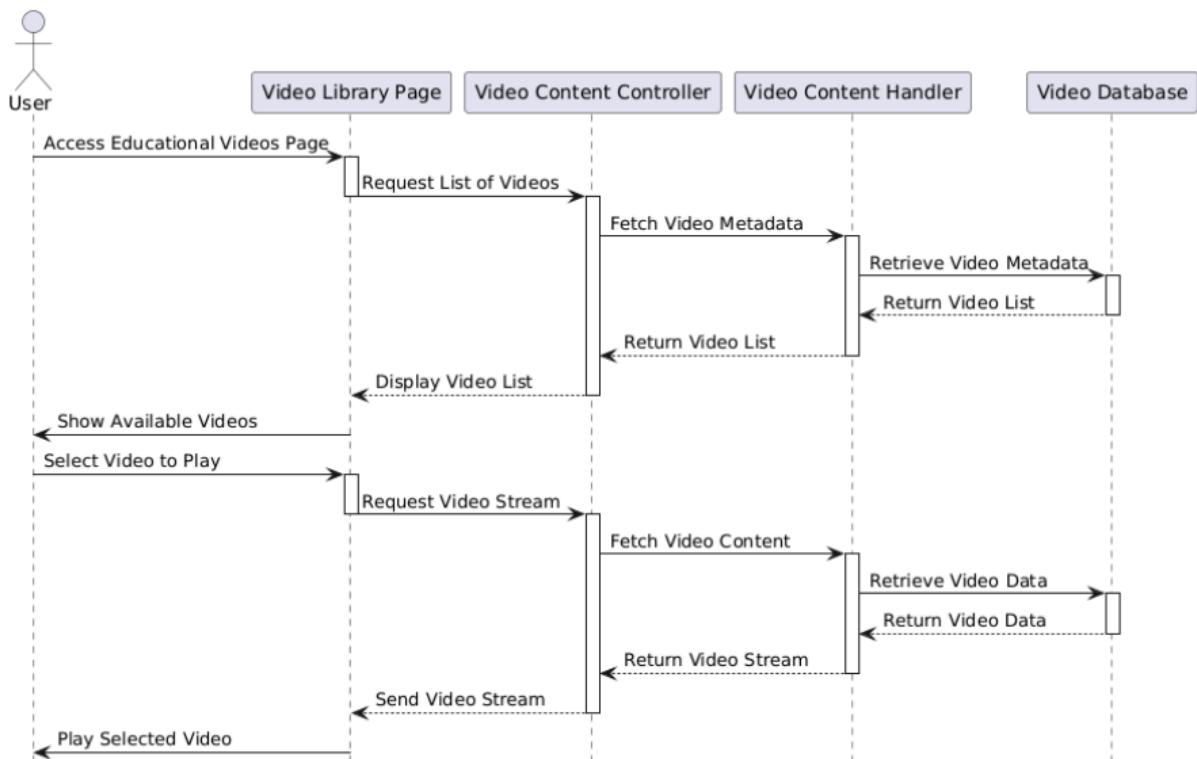


Figure 2.4.2.7.1 : Sequence Diagram for <View Education Videos>

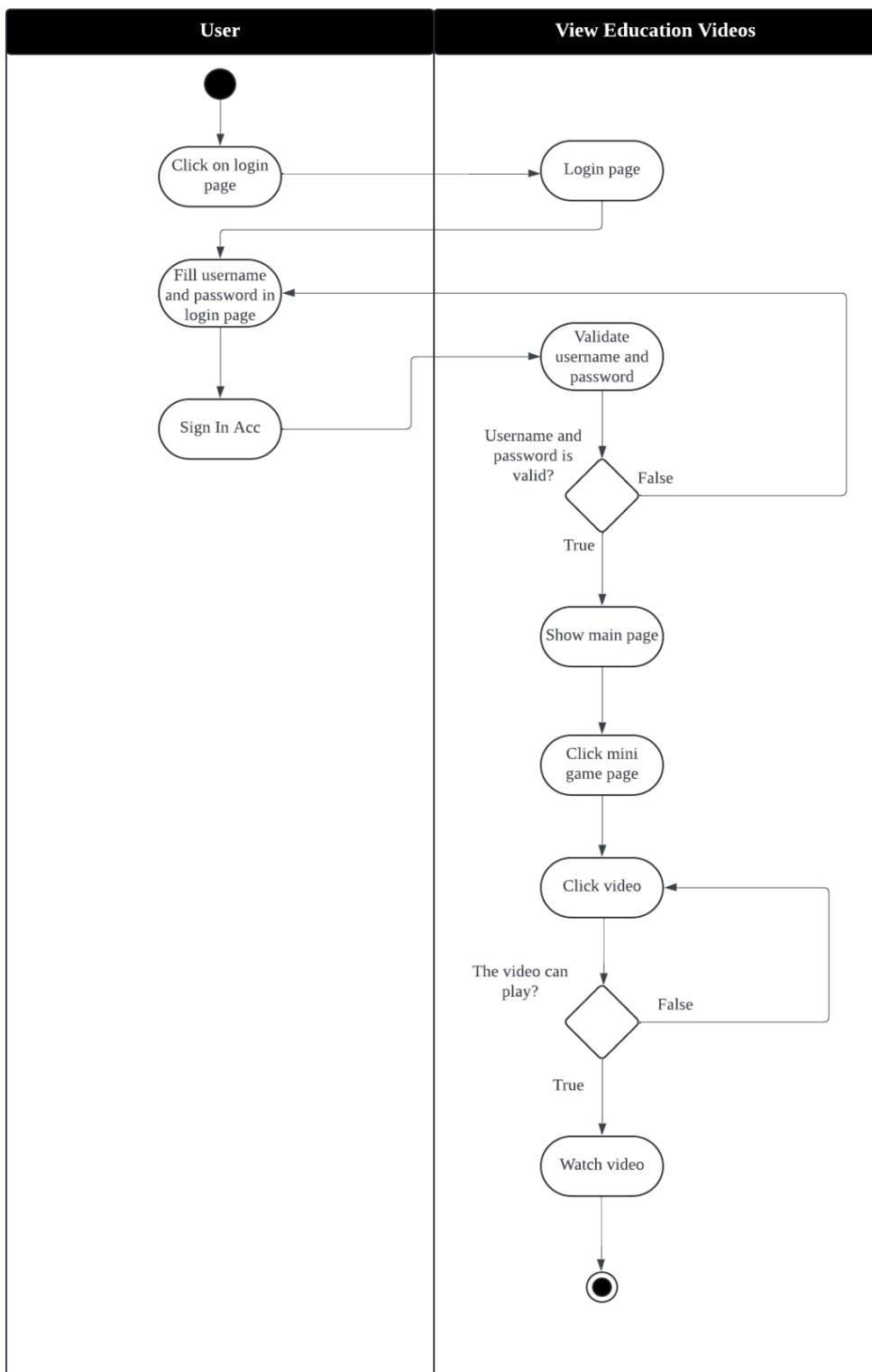


Figure 2.4.2.7.2 : Activity Diagram for <View Education Videos>

2.4.3 US09: User Story <Administrator>

Table 2.4.3.1: User Story Description for <Login to Admin Dashboard>

| User story: <Login to Admin Dashboard> | |
|---|--|
| ID: US09 | |
| User Story Description | |
| <p>As a admin application DPCM I want to login as admin for application So that i can manage the admin dashboard</p> | |
| Flow of events: | |
| <ol style="list-style-type: none"> 1. Admin login the application 2. Admin go to admin dashboard 3. Admin can manage the dashboard | |
| Alternative flow n: | |
| Acceptance Criteria | |
| <p>Postcondition - The admin can manage dashboard using admin dashboard Precondition - The admin has successfully logged into the application.</p> | |
| Exception flow: | |
| The system fails to display the admin dashboard, it will display a message “Try again” and the admin will stay in the home page. | |

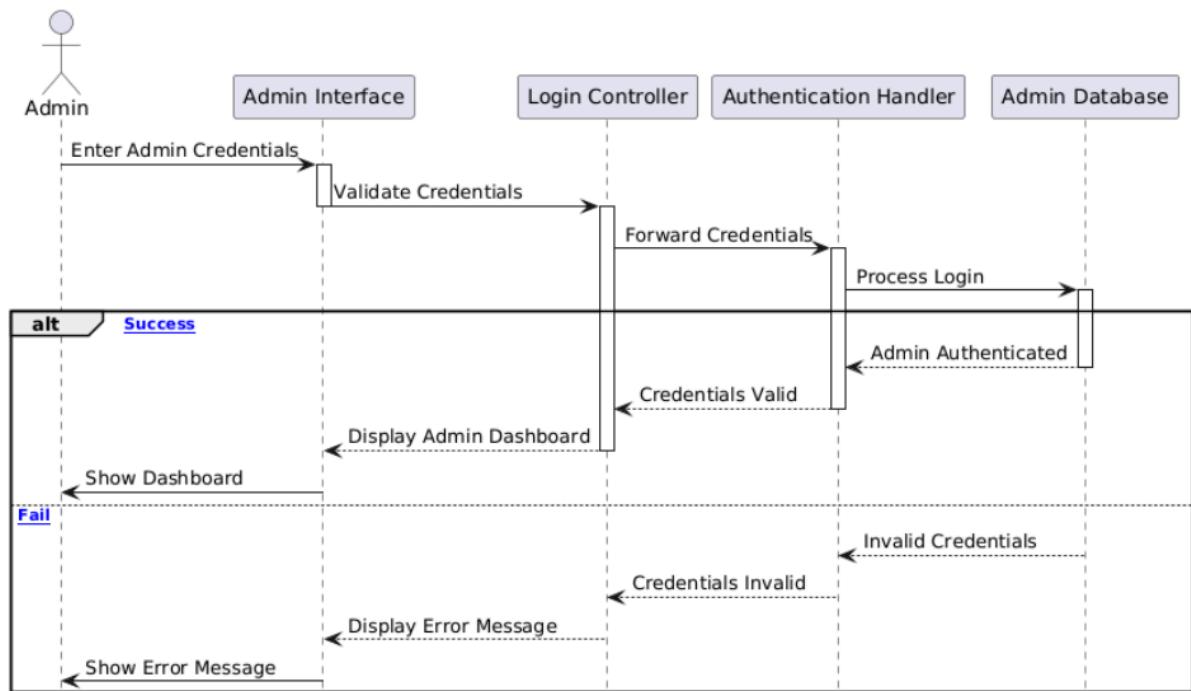


Figure 2.4.3.1.1 : Sequence Diagram for <Login to Admin Dashboard>

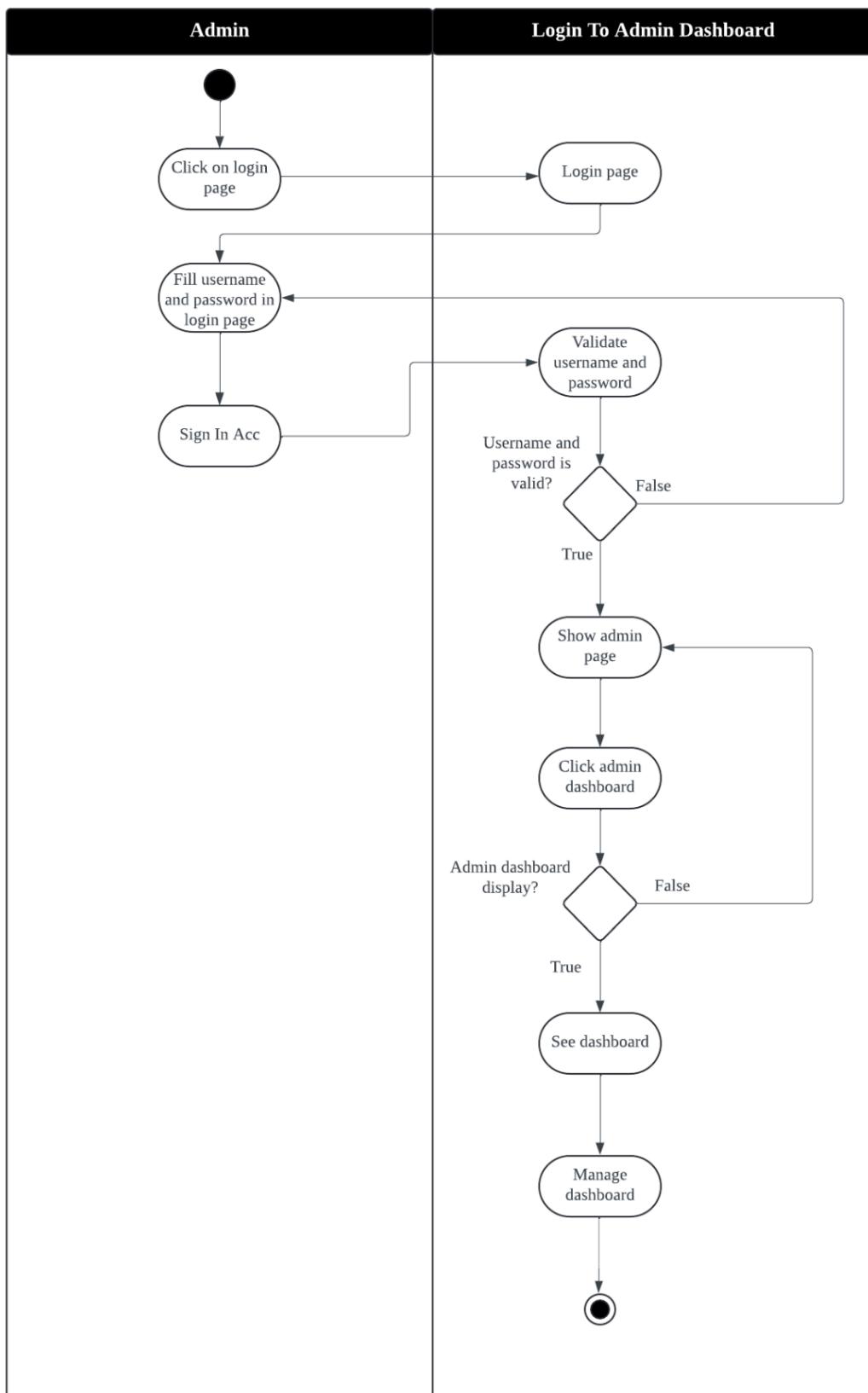


Figure 2.4.3.1.2 : Acitivity Diagram for <Login to Admin Dashboard>

Table 2.4.3.2: User Story Description for <Monitor System Logs>

| User story: <Monitor System Logs> | |
|--|--|
| ID: US10 | |
| User Story Description | |
| As a admin application DPCM I want to login as admin So that i can manage the application | |
| Flow of events: | |
| <ol style="list-style-type: none"> 1. Admin login the application 2. Admin can manage the application 3. Admin can monitor system logs | |
| Alternative flow n: | |
| Acceptance Criteria | |
| Postcondition - The admin can manage application and monitor system logs Precondition - The admin has successfully logged into the application. | |
| Exception flow: | |
| The system fails to display the log list or details, then it will ask to re-request the log list or details by displaying an error message “Can’t proceed the request, try again”. | |

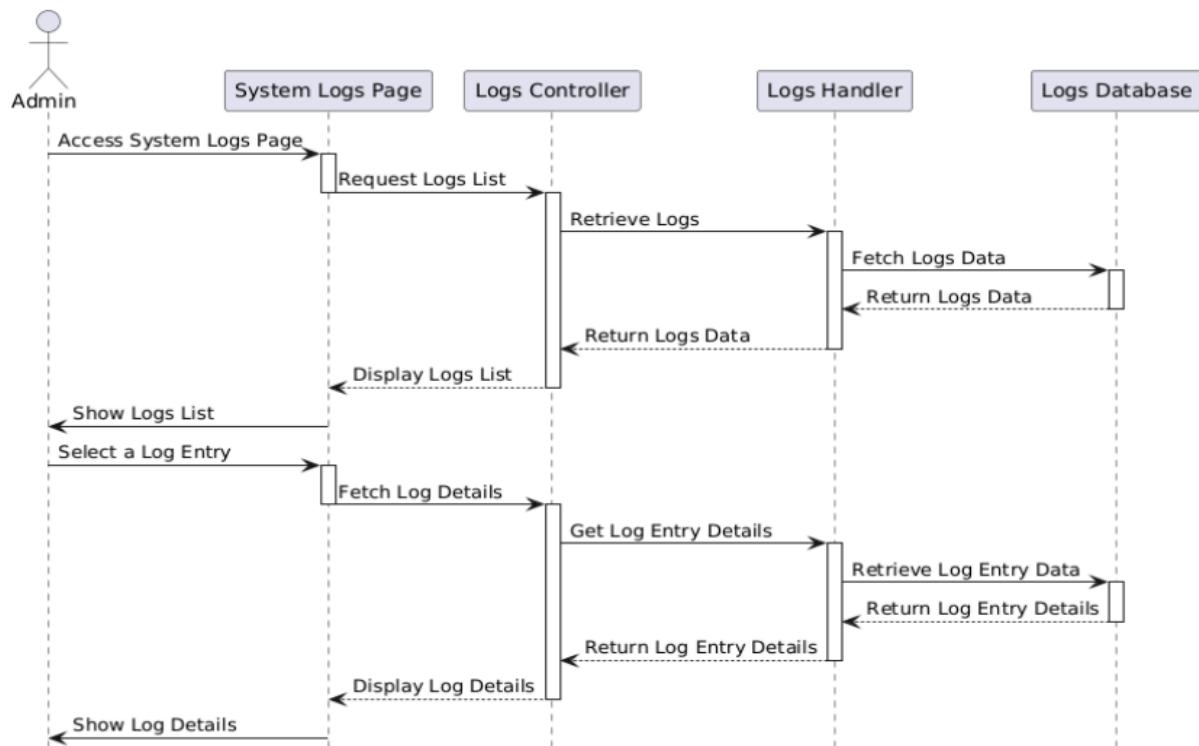


Figure 2.4.3.2.1 : Sequence Diagram for <Monitor System Logs>

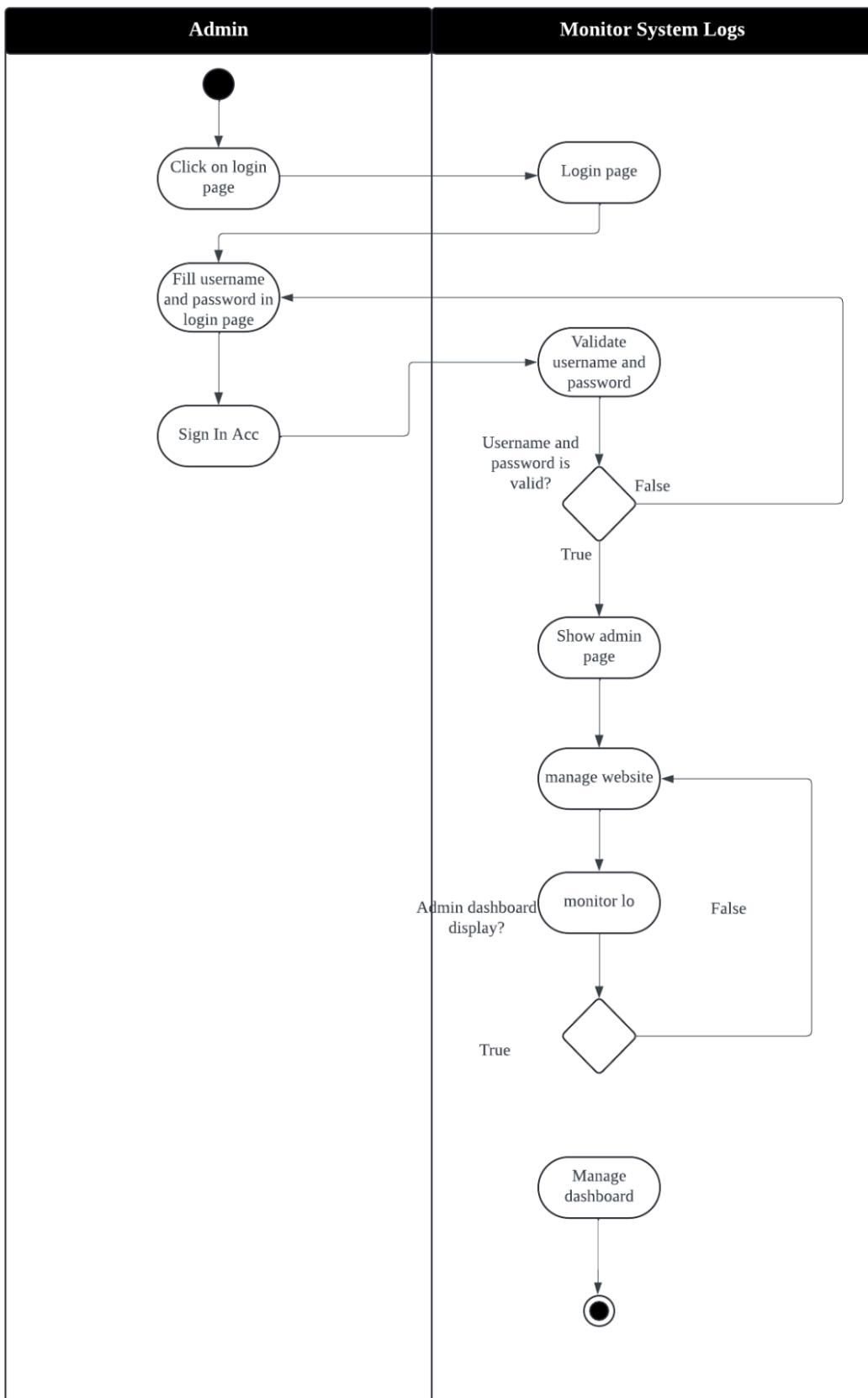


Figure 2.4.3.2.2 : Sequence Diagram for <Monitor System Logs>

Table 2.4.3.3: User Story Description for <Verify User Reports>

| |
|---|
| User story: <Verify User Reports> |
| ID: US11 |
| <p>User Story Description</p> <p>As an admin application DPCM I want to see and verify the report So that I can update the dengue map data.</p> |
| <p>Flow of events:</p> <p>+ </p> <ol style="list-style-type: none"> 1. The admin accesses the Reports Dashboard. 2. The system displays a list of Pending Reports with basic details 3. The admin selects a specific report to review. 4. The system displays the report's detailed information, including location, description, and any attached images. 5. The admin verifies the report and chooses to either: <ul style="list-style-type: none"> - Approve the report: Updates the status to "Approved" and triggers the update of the dengue map. - Reject the report: Updates the status to "Rejected" and sends a rejection message to the user. |
| <p>Alternative flow n:</p> <p>The admin selects a report but finds the provided information insufficient. The admin marks the report as "Needs More Information." A notification is sent to the user requesting additional details or clarification. The report remains in the "Pending" state until updated by the user.</p> |
| <p>Acceptance Criteria</p> <p>Postcondition - The report status is updated to either "Approved" or "Rejected." If approved, the dengue map data is updated to reflect the new information.</p> <p>Precondition - The admin has successfully logged into the application.</p> |
| <p>Exception flow:</p> <p>When the system fails to display the pending report's details, it will display a message "Try again". The same when the admin approves or rejects the report.</p> |

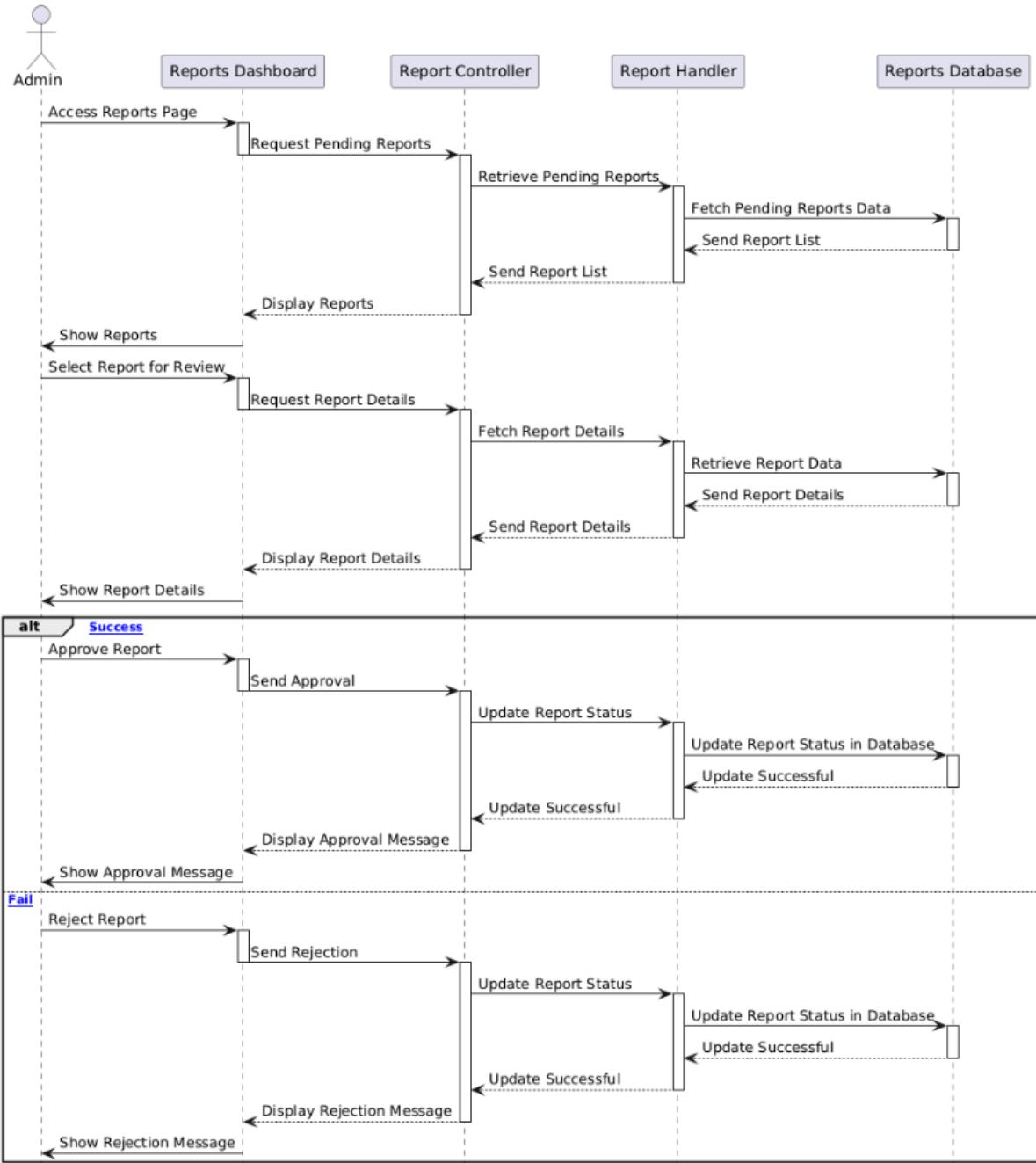


Figure 2.4.3.3.1 : Sequence Diagram for <Verify User Reports>

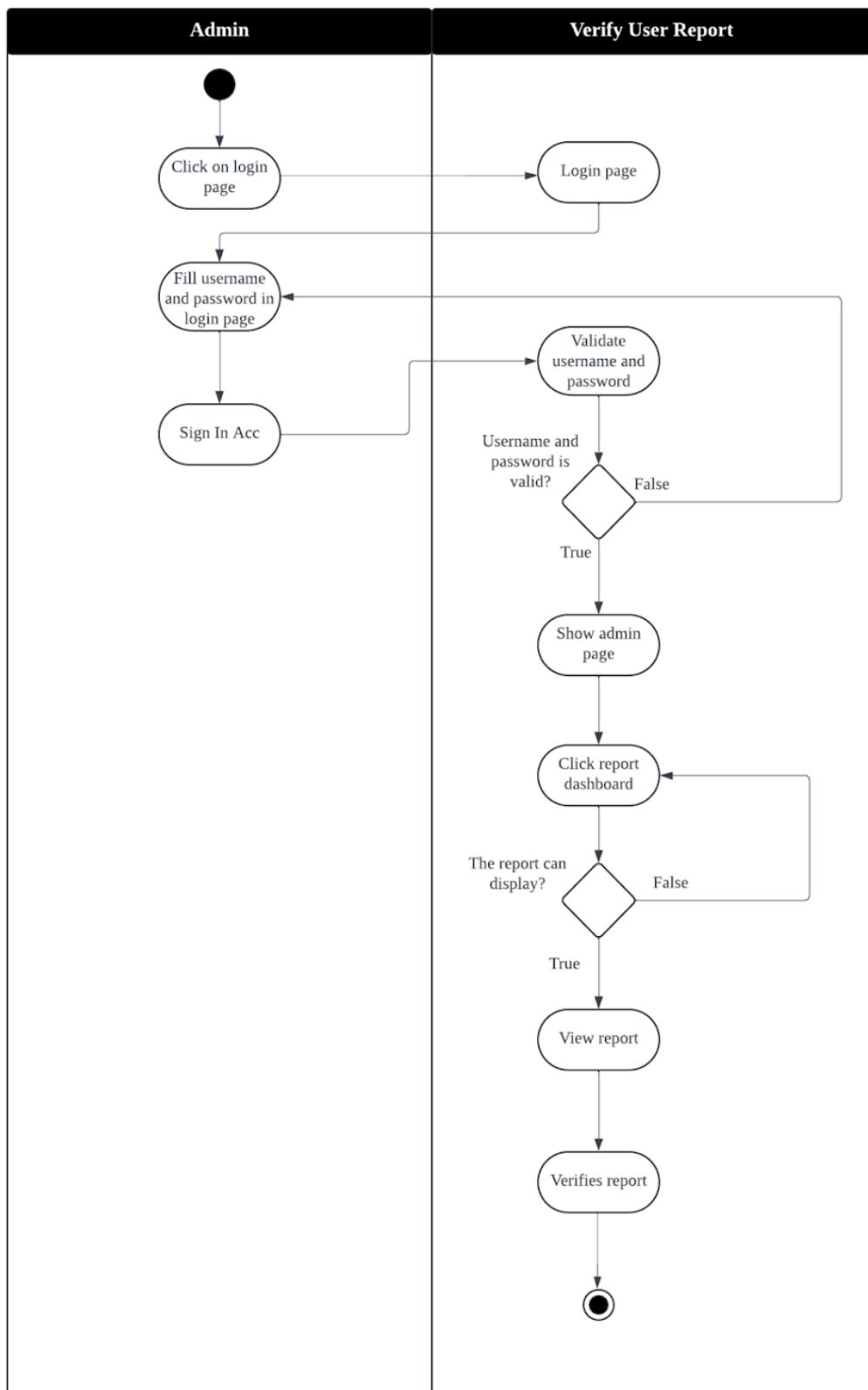


Figure 2.4.3.3.2 : Activity Diagram for <Verify User Reports>

Table 2.4.3.4: User Story Description for <Update Dengues Case Statistics>

| User story: <Update Dengues Case Statistics> |
|---|
| ID: US12 |
| User Story Description As a admin for application DPCM I want to update the data about dengue case So that user can see the up to date data |
| Flow of events: 1. Admin login the application 2. Admin update the data website 3. Admin save the data |
| Alternative flow n: |
| Acceptance Criteria Postcondition - The data in application can update and save by admin for the latest information Precondition - The admin has successfully logged into the application. |
| Exception flow: If the admin is not able to save the updated data, the system will display an error message “Please try again” and the admin will stay in the dengue statistics page. |

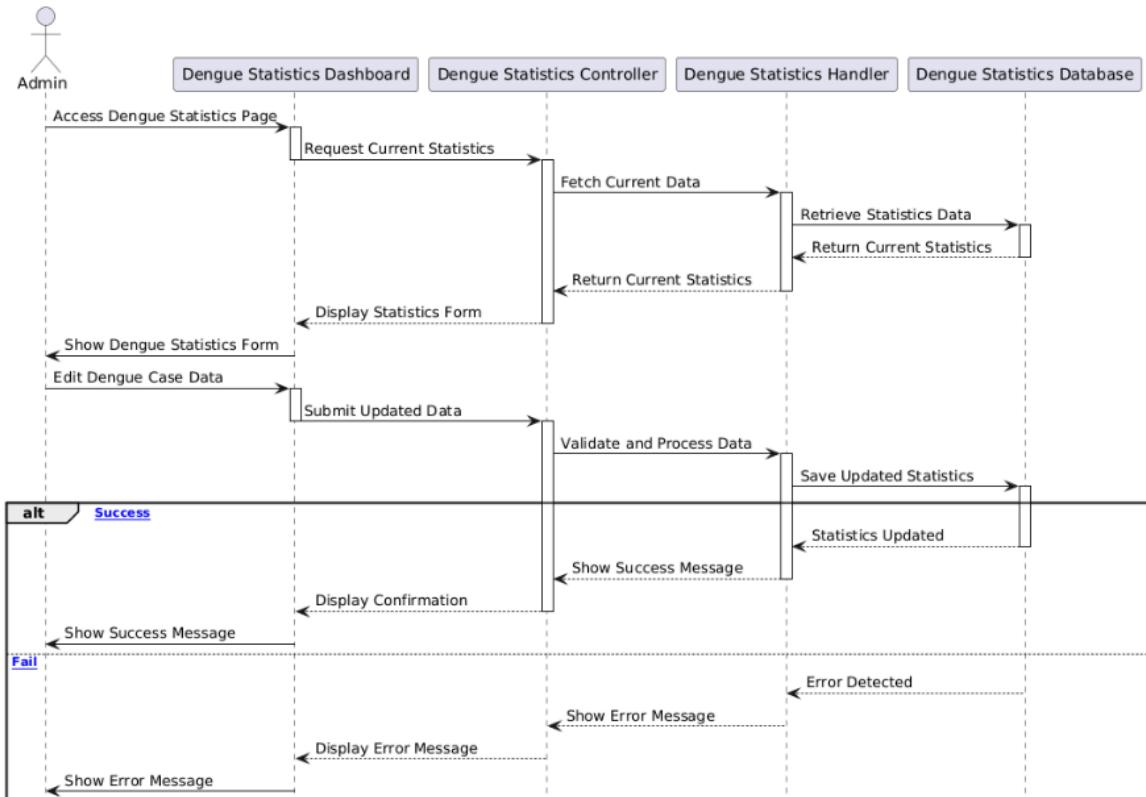


Figure 2.4.3.4.1 : Sequence Diagram for <Update Dengue Case Statistics>

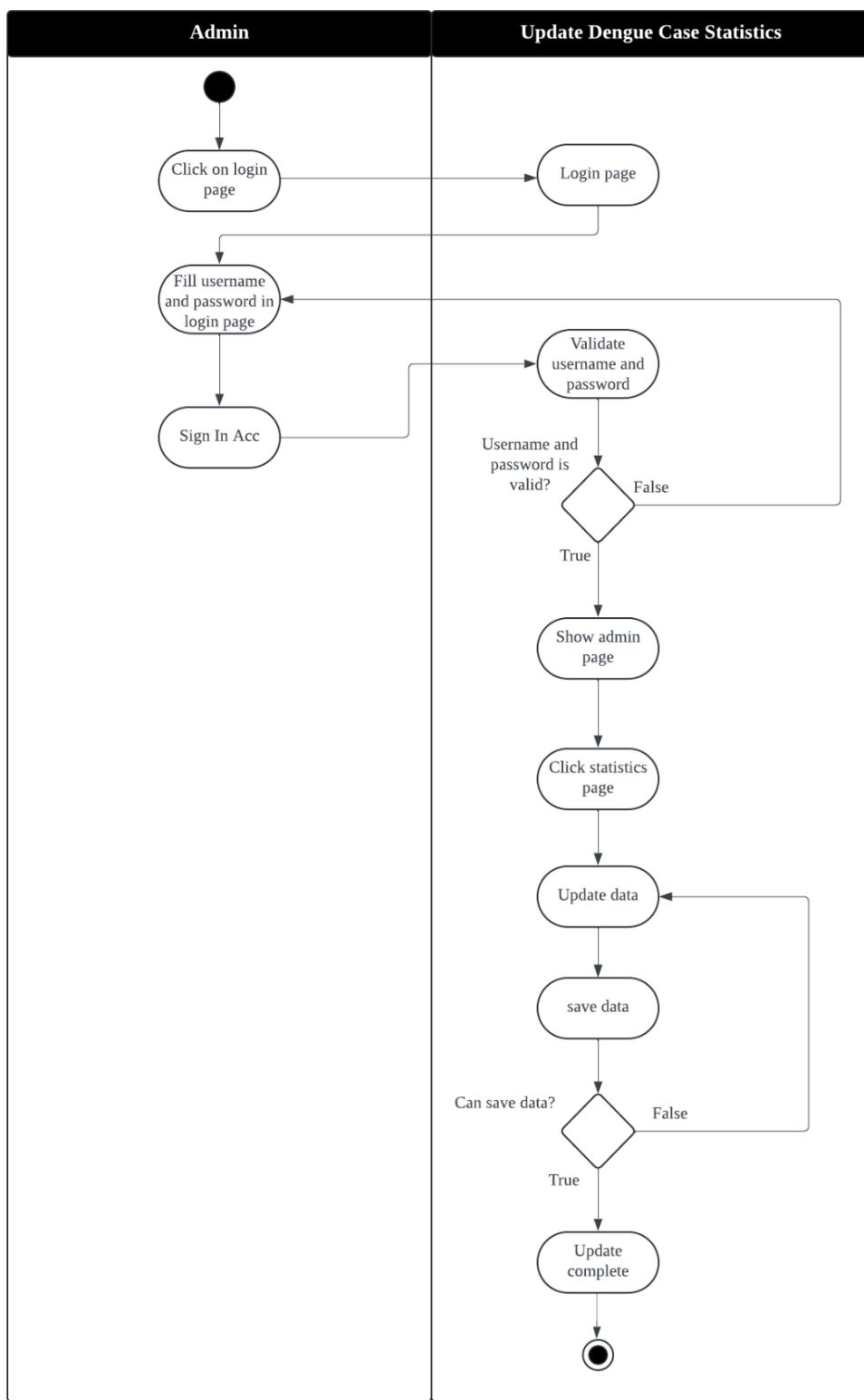


Table 2.4.3.5: User Story Description for <Manage Quizzes>

| User story: <Manage Quizzes> | |
|--|--|
| ID: US13 | |
| User Story Description | |
| <p>As a admin for application DPCM I want to manage quizzes question So that user can see different question in quiz</p> | |
| Flow of events: | |
| <ol style="list-style-type: none"> 1. Admin login the application 2. Admin go to page mini game 3. Manage question 4. Save editing | |
| Alternative flow n: | |
| Acceptance Criteria | |
| <p>Postcondition - The data in application can update and save by admin for the latest information</p> <p>Precondition - The admin has successfully logged into the application.</p> | |
| Exception flow: | |
| If the admin is not able to save the updated data, the system will display an error message “Please try again” and the admin will stay in the mini game page. | |

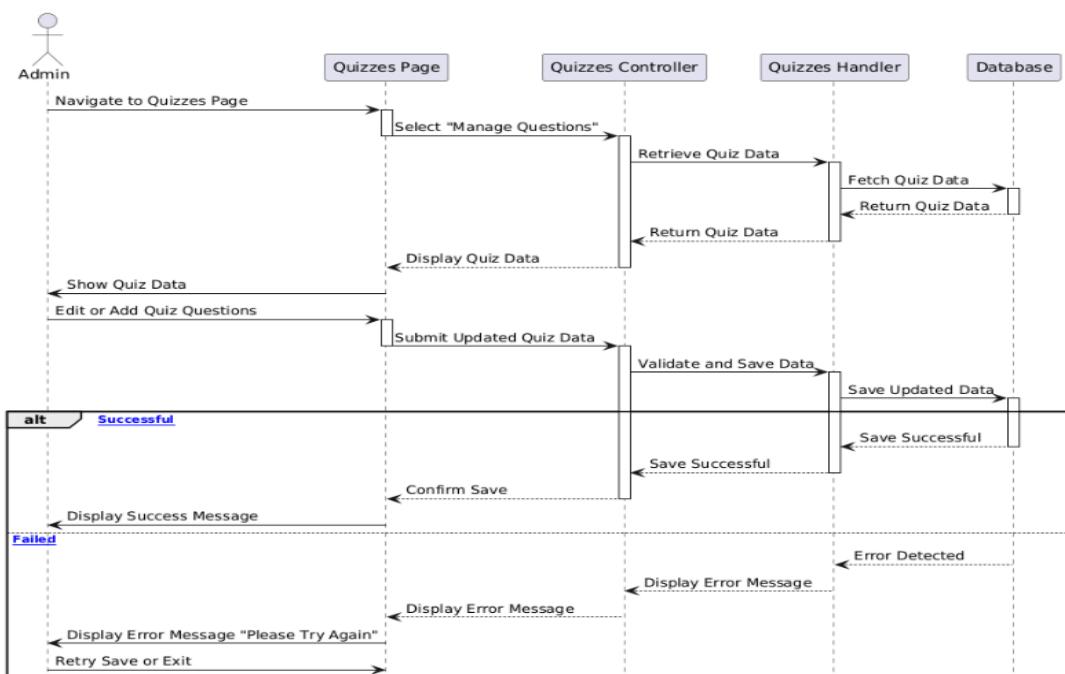


Figure 2.4.3.5.1 : Sequence Diagram for <Manage Quizzes>

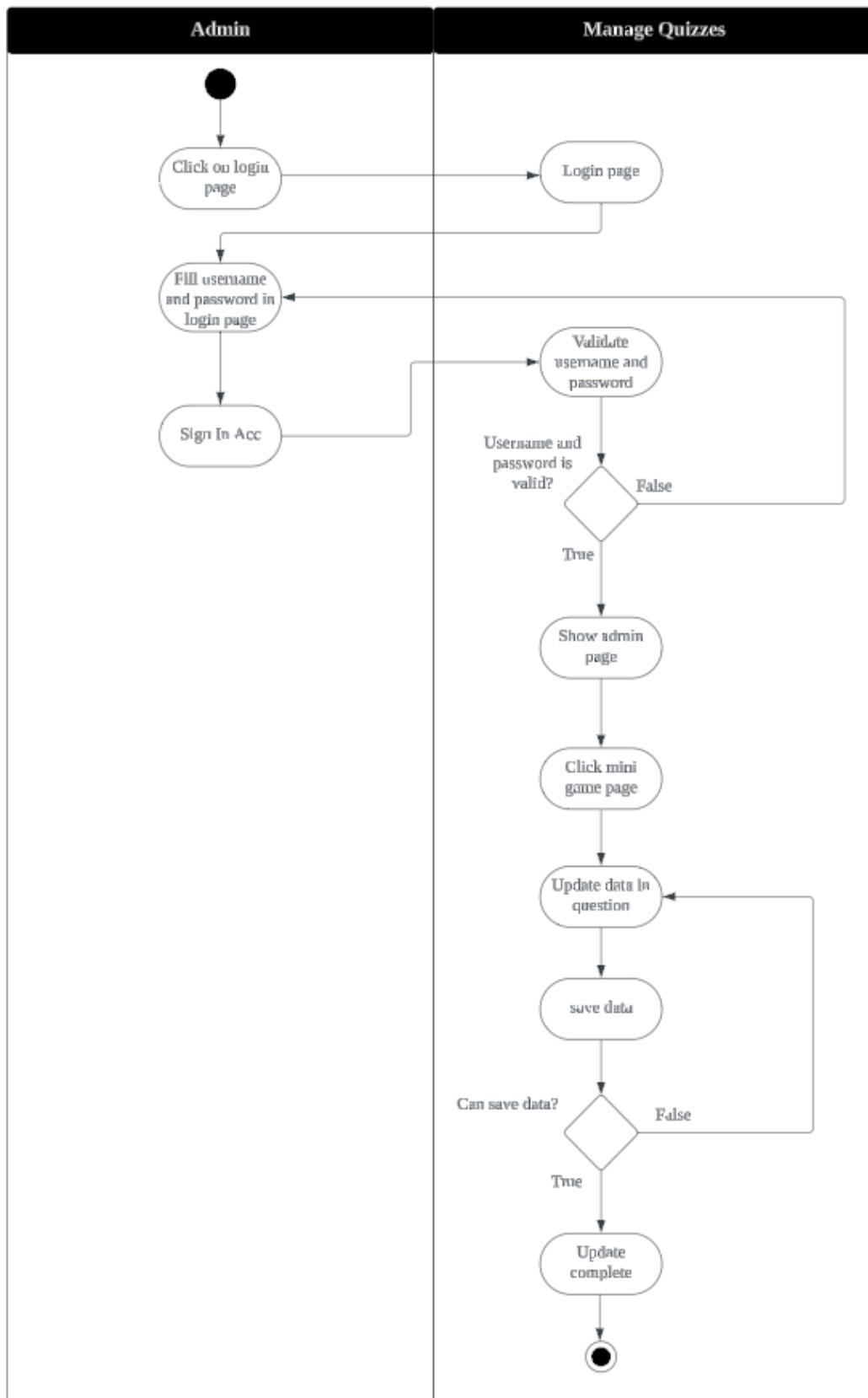


Figure 2.4.3.5.2 : Activity Diagram for <Manage Quizzes>

2.5 Performance and Other Requirements

Non-functional requirements, which emphasize how a system functions rather than what it does, specify the attributes and features of a system. These specifications are essential because they guarantee that the system operates effectively, stays safe, and offers a flawless user experience in a variety of scenarios. While functional requirements deal with certain features or actions (such reporting breeding locations or seeing a dengue map), non-functional requirements make sure that these duties are carried out safely, effectively, and dependably.

Performance Requirements

1. Response Time:

- In typical circumstances, the system should react to user input (such as page loads and form submissions) in less than two seconds.
- It should take no more than **five seconds** to render a map for real-time dengue data.

2. Throughput:

- During peak hours, the system should support up to **100 users** at once without experiencing appreciable performance deterioration.

3. Capacity:

- The system should be able to accommodate **200 registered users** while still operating at peak efficiency.
- Up to **500 reports of mosquito breeding places** and **500 quiz attempts** should be handled by database storage without experiencing any performance problems.

4. Availability:

- With the exception of planned maintenance, the system should always be accessible with a 99.9% uptime rate.

Other Software System Attributes

1. Usability:

- The interface should be simple and easy to use so that users may accomplish activities (such reporting breeding locations or seeing dengue data) with just **three clicks**.

2. Reliability:

- For essential features like real-time map updates and report submissions, the system must have an error rate of less than 0.1%.
- All user data should be backed up daily to prevent data loss.

3. Maintainability:

- The system codebase should be well-documented and modular such that bug patches or changes can be made within **48 hours** of the problem being discovered.

4. Portability:

- The system should run seamlessly on major browsers (Chrome, Firefox) and platforms (Windows 10+, macOS).

5. Security:

- To ensure security, user data must be encrypted both during transmission (HTTPS/TLS 1.2 or above) and at rest.
- Set up admin accounts using two-factor authentication.

6. Safety:

- Unless anonymized for privacy, the system shouldn't retain or show sensitive personal information, such as complete addresses.

7. Environmental:

- Reduce needless work and optimize searches to minimize server energy usage.

2.6 Design Constraints

Based on the design of the DPCM system that we produce, this system has several constraints that we will use for this DPCM system, among which are

1. Environmental constraints:

The environment that is constrained in this system is related to the physical environment where this system displays an attractive design and provides a comfortable view when reading information. This system also uses bright colors that can satisfy the user's eyes when looking like reducing the user's colors that are too extreme.

2. Hardware constraints:

Hardware constraints related to the systems we build where the hardware requirements for these systems such as memory processors and storage. The minimum type of processor required is Intel i3 or AMD R3, for memory the minimum is 4GB and for storage the minimum is 2TB to store all the data used.

3. Security constraints:

The security requirements for this system are very necessary to protect all data in this system as well as data from users who use this system. Among the security we use is authentication where the system will ask users to register their information and self-authentication.

4. Compatibility constraints:

This constraint is related to the compatibility of the system produced with the equipment used as this system is specially designed to make this system compatible with computer software where it is suitable for computers that use Windows 10 and above.

5. Performance constraints:

The constraints for this system's performance are being able to control 500 users and have a fast response time. This system also uses real time data where users can find out the latest information and notifications about dengue.

3. Architectural Rationale

This part is one of the most important parts to build an application because in this part it will explain the architectural style and the rationale or justification of the selection we use to build the Dengue Prevention and Control system in Malaysia (DPCM) application. In this section we will also include some diagrams of components and subsystems that we will use for this system.

3.1 Architecture Style and Rationale

3.1.1 Architectural Overview

The Dengue Prevention and Control system in Malaysia (DPCM) is designed with three layers; presentation layer (frontend), application layer (backend), and data layer (database). This kind of architecture provides separation of concerns, scalability, and maintainability. The system is implemented as a web-based application for accessibility on all platforms, including desktops, tablets, and smartphones.

Key Architectural Design Choice:

- Dynamic website: Dynamic websites are interactive and dynamic, allowing updates in real-time and ensuring user engagement.
- RESTful API integration: Allows backend communication with third-party real-time data sources.
- Scalable database: Support for extending the system in the future.

3.1.2 Architectural Components

1. View Layer(Frontend)

Responsibilities:

- Show live dengue maps and notification to users
- Utilize dynamic tools such as quiz and gamified learning
- Bring social elements to the system through forums, user interactions, etc.

2. Domain Layer (Backend)

Responsibilities:

- Handle user data and requests
- Real-time dengue outbreak tracker that integrated with external APIs
- Handling gamified learning and report systems

3. Data Layer (Database)

Responsibilities:

- Store the informations from users, such as user data, forum posts, or dengue cases report
- Ensure data integrity and security

3.1.3 Architecture Rationale

1. Scalability

Each layer can be scaled independently because the system will be using three-tier architecture. The database can be tuned for high traffic, and then other frontend instances can be balanced and deployed.

2. Flexibility

Also, one of the benefits of the separation of the UI layer is that the system can upgrade to the latest technology or a new framework of UI without affecting the backend logic

3. Security

Backend validation prevents the user input from being handled in dangerous ways. Data exchange will be performed over secure communication protocols.

4. Real-Time Features

Additionally, the backend connects with third-party APIs and leverages WebSocket protocols for real-time data refreshes, keeping the application current with dengue trend fluctuations

5. User Engagement

Modular components allow the necessary gamified elements and social interaction features to be continuously updated or expanded as needed.

4. Architectural Views

This SD section includes one of the important sections to build an application where this section will focus on the system architecture view based on the framework developed by Philippe Kruchten. In this architectural view, there are four types of views, among which are the scenario view, logical view, development view, process view and physical view as shown in the diagram below and the explanation for each view.

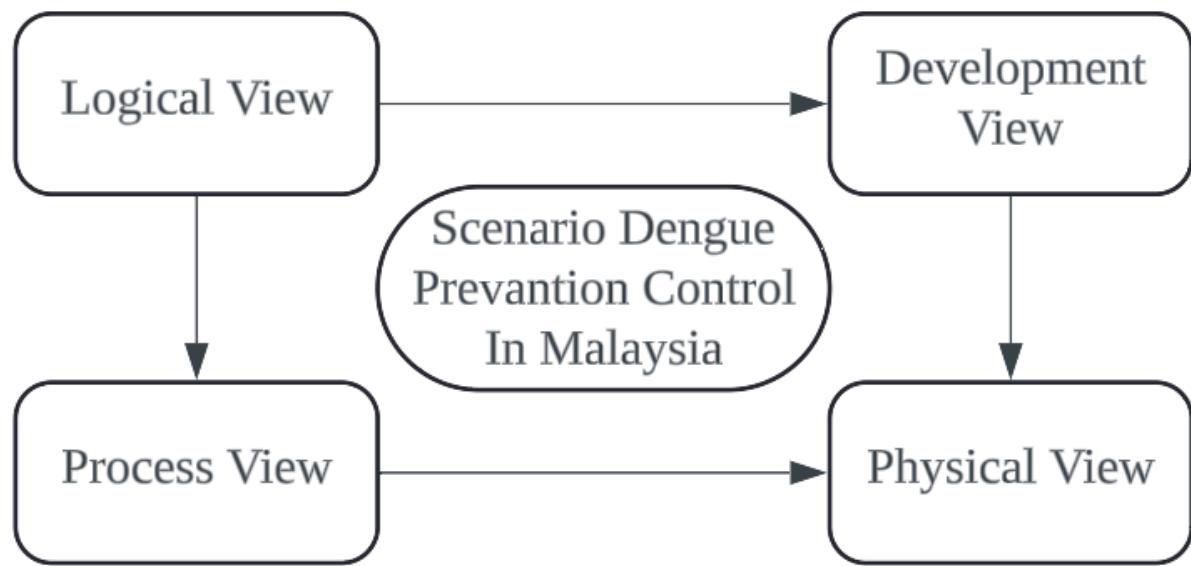


Figure 4.1: Architectural Views

Table 4.1: Architectural Views

| Architectural Views | Description | Diagrams used |
|----------------------------|--|--------------------------------------|
| Use Case View | This use case will show an overview of how this system works for users | Use Case Diagram |
| Logical View | This logical view shows the main abstractions in the system as objects or classes of objects and displays each subsystem that exists | Class Diagram |
| Process | The process view is related to the dynamic aspect of the system, specifically this view will show how each component in the system will interact with each other | Sequence Diagram |
| Development View | The development view shows how software is decomposed into components for development | Component Diagram Package Diagram |
| Physical View | The physical view shows how the system is physically deployed on the hardware infrastructure | Deployment Diagram |

4.1 Use Case View

A Use Case View for a system Dengue Prevention and Control in Malaysia provides a detailed perspective on how the system interacts with its users and other systems. Below is a detailed description of what a Use Case View might look like based on the name of the system

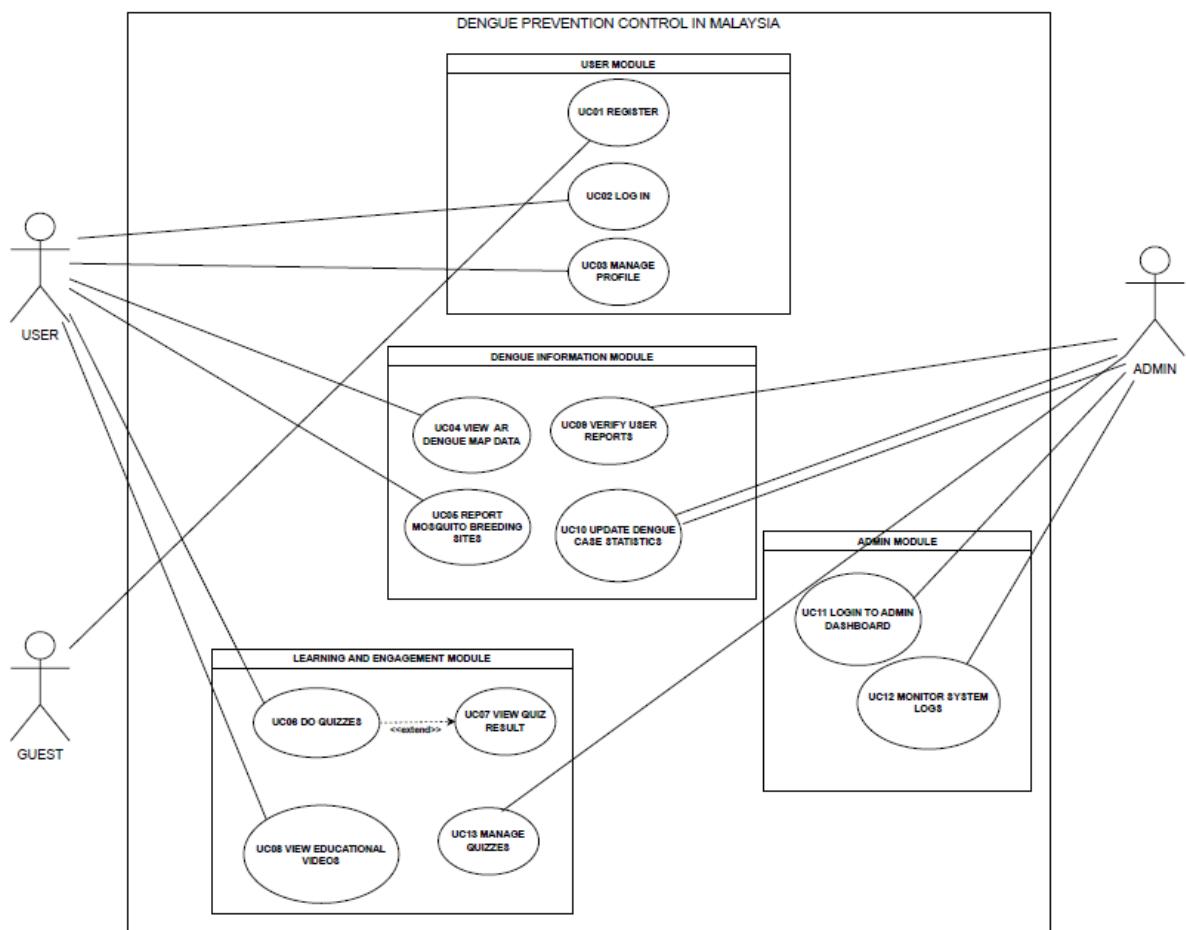


Figure 4.2: Use Case Diagram for <DPCM system>

4.2 Development View

This development view shows the organization of system software modules in its development environment. It focuses on how the system is decomposed into components for implementation. In this section we will show how this system divides and allocates to other subsystems. As in the diagram, we can see how this system works starting from the user by confirming to access the application, after the user enters the interface which is the happy Dengue Application, the user will see several sections on the interface including Dengue Information, Feedback and Game. Based on these 3 types of sections, users can choose to see the section they want to see, for example if the user chooses to press Dengue Information then this section will take the user to another subsystem where the subsystem is Dengue Data. In the Dengue subsystem of this data there is some information that users can see such as Dengue Status and Dengue Map so users can find out the latest information about dengue. In addition, if the user wants to give any report or feedback regarding dengue in their area, the user can look at the Feedback section where it is also connected to the Dengue Data subsystem, if the user clicks on the Feedback section the user can give an opinion or report about the dengue case in the incident area so based on the report that the user gives the system will update according to the new data. Next, on the interface, the user can play a simple game about dengue, so if the user clicks on the Game section, the system will take the user to another subsystem, the Mini Game. In this mini game there are two types of sections, among them quizzes where users can answer quizzes based on dengue and users can find out their overall score. Next is the video section where users can learn and understand more easily using the simple videos that have been provided. Below we include a Component Diagram for the Dengue Prevention and Control In Malaysia system.

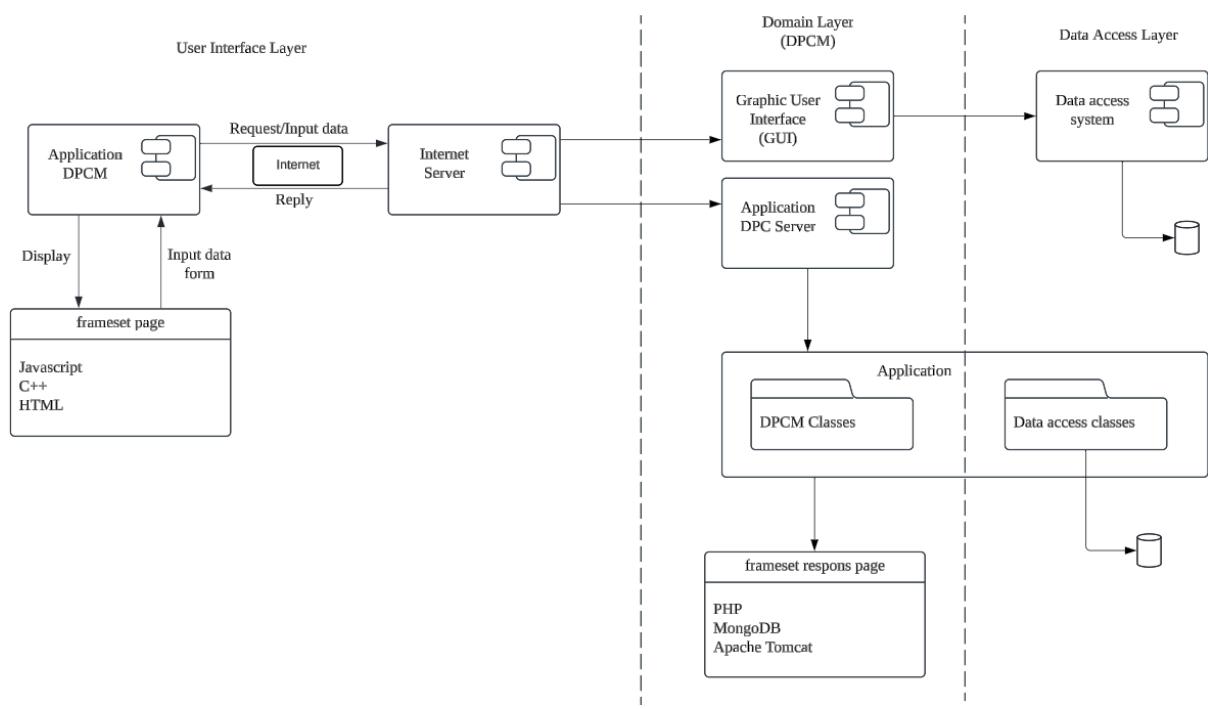


Figure 4.3: Component Diagram of <DPCM System>

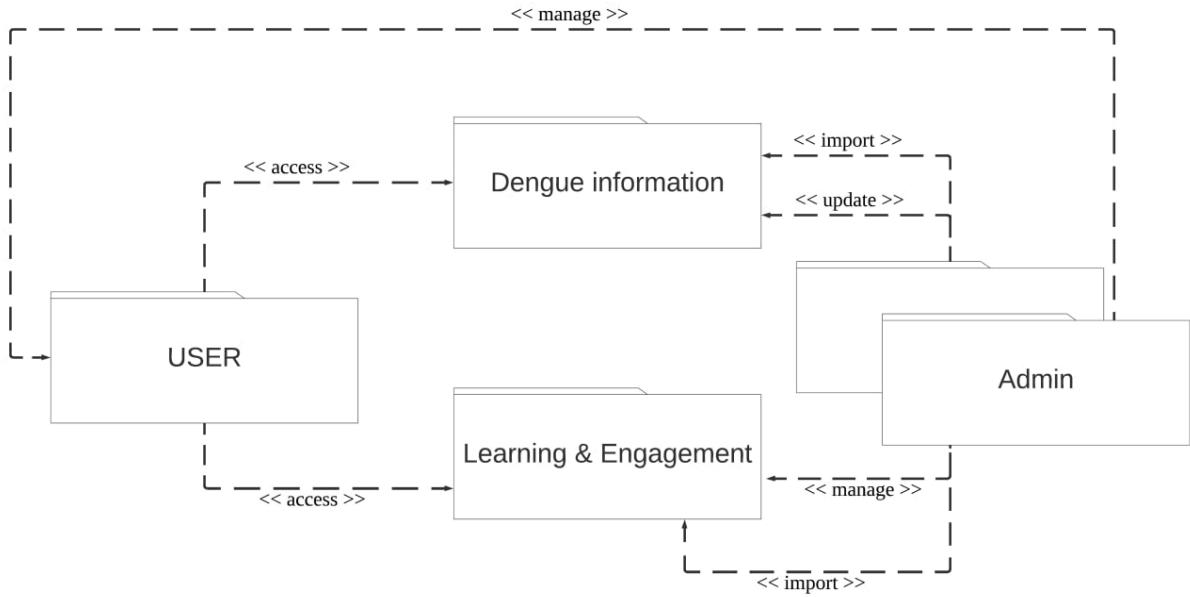


Figure 4.4: Package Diagram of <DPCM System>

The Unified Modeling Language(UML) package diagram above shows the structured design of a system complying with the Model View Controller architectural type. MVC is an architectural design pattern that partitions a system into three core cooperating components namely Models, Views and Controllers in order to abstract the internal structure of data and information from the way such information is represented and used. The diagram shows the decomposition of system capabilities into three core functional packages: administration module, user management module, and quiz module. The package contains model, view and controller roles thus promoting the concepts of modularity, scalability and maintainability.

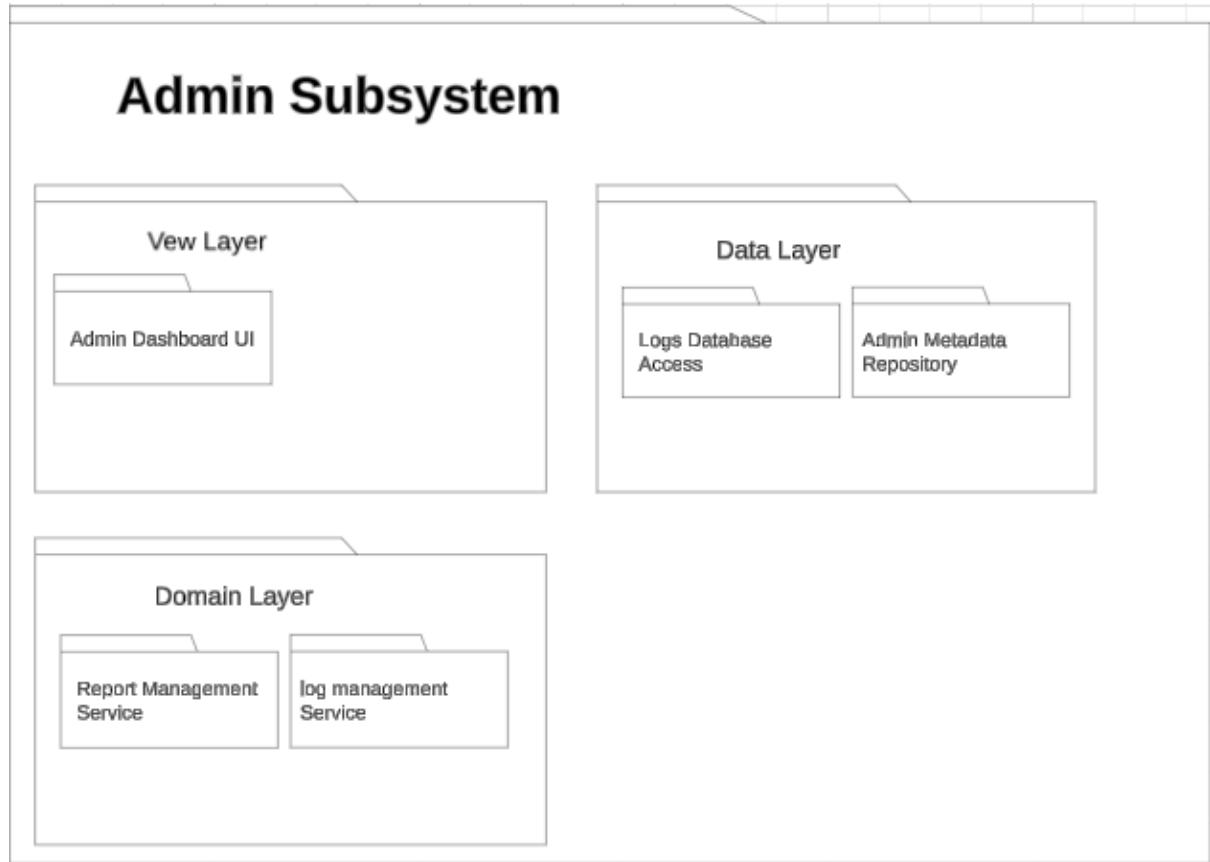


Figure 4.5: Package Diagram For<Admin Package>Subsystem

This package mainly contains parts which are both for Controllers (ADMIN) and for Models (ReportCases and Dengue Map). The component ADMIN is the main controller that is focused on administrative functionalities and maintaining data consistency across the system when updates are made.

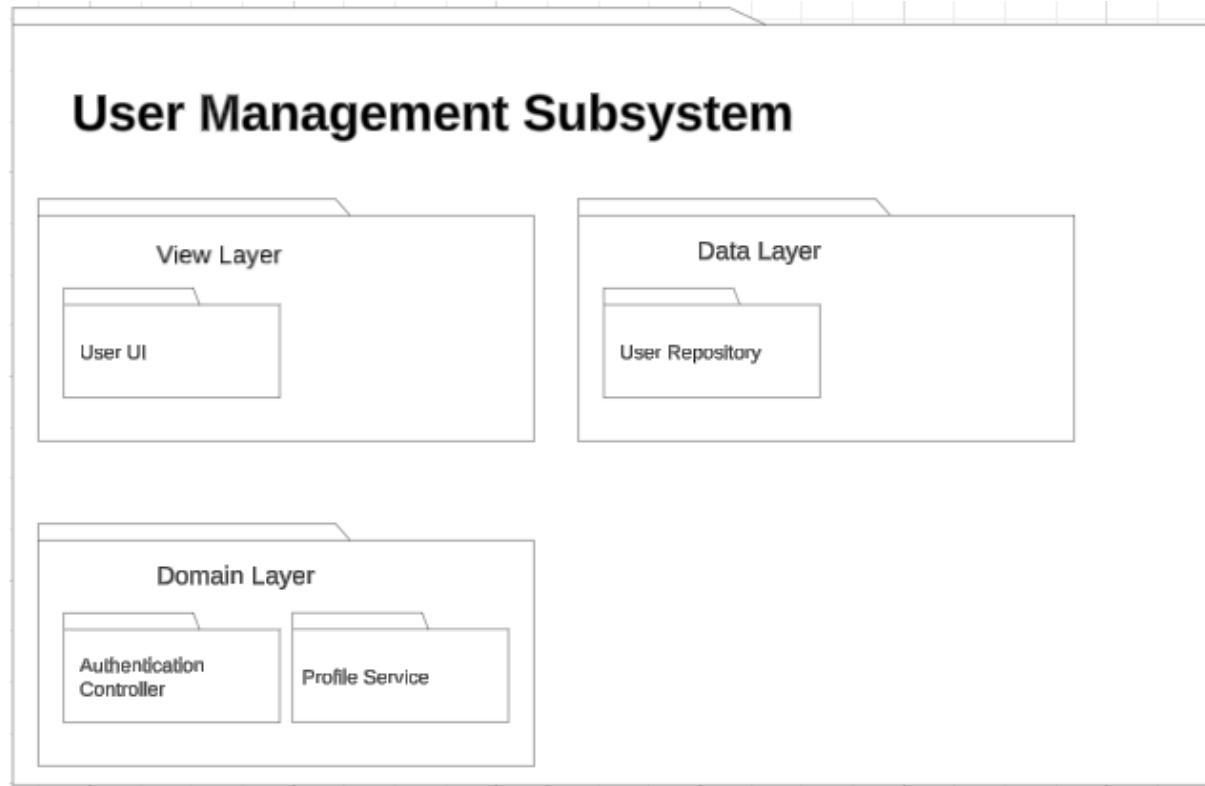


Figure 4.6: Package Diagram For<User Management Package>Subsystem

This package is consistent with the Controller layer of the MVC. The USER and GUEST constituents act as intermediaries between the end users and the system such as the ReportCases, QUIZ and Dengue Map by controlling the permissions on Models and Views.

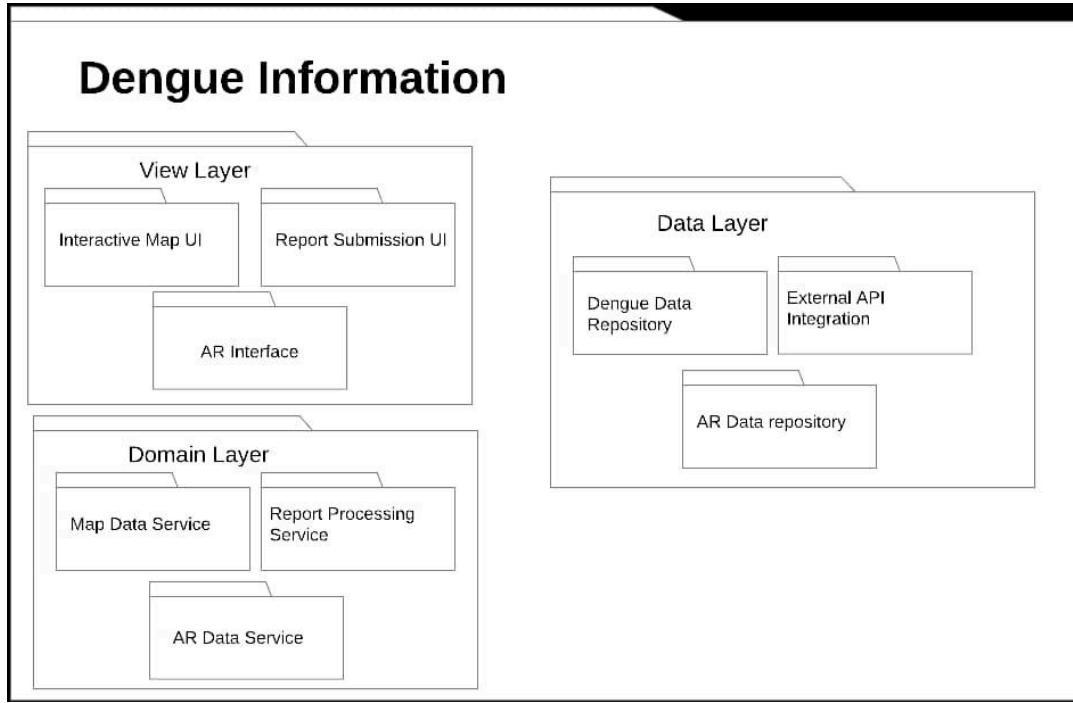


Figure 4.7: Package Diagram For<Dengue Information>Subsystem

The Dengue Map is a component aimed at depicting geographical information in context to cases of dengue. This component could participate in MVC architecture as a model which holds and visualizes important statistics that are related to the dengue epidemic. Nevertheless, it might also perform View functions by enabling users to interact with the application in a more visually appealing manner.

Learning & Engagement

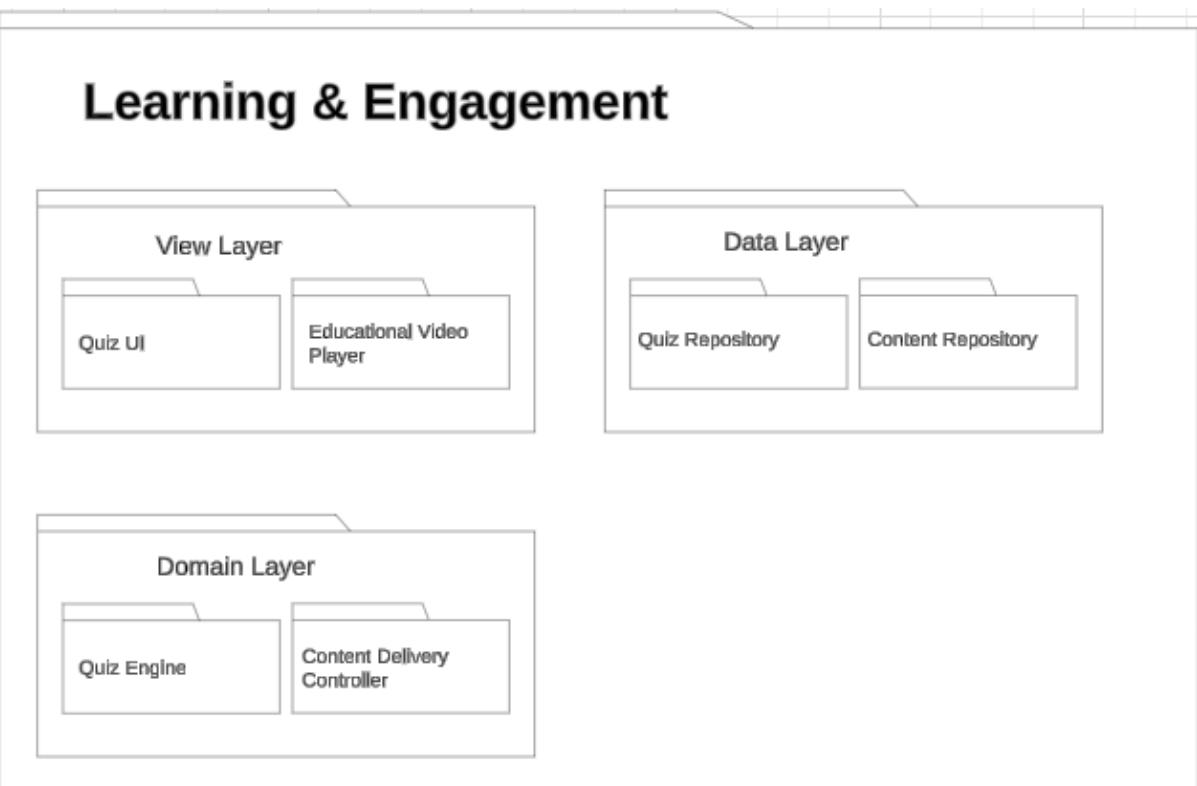


Figure 4.8: Package Diagram For<Learning & Engagement>Subsystem

The Quiz Package merges all three components of the MVC architecture. The View in this case is represented by the QUIZ component, which allows the user to engage and participate. The QuizResult component functions as the Model with the perspective of data storage and control. All of these components work together in order to provide a thorough quiz feature.

4.3 Logical View

The logical view for this system will show each object or object class that we use in the Dengue Prevention and Control In Malaysia system. This view will focus on each class used in the system and will show each existing subsystem.

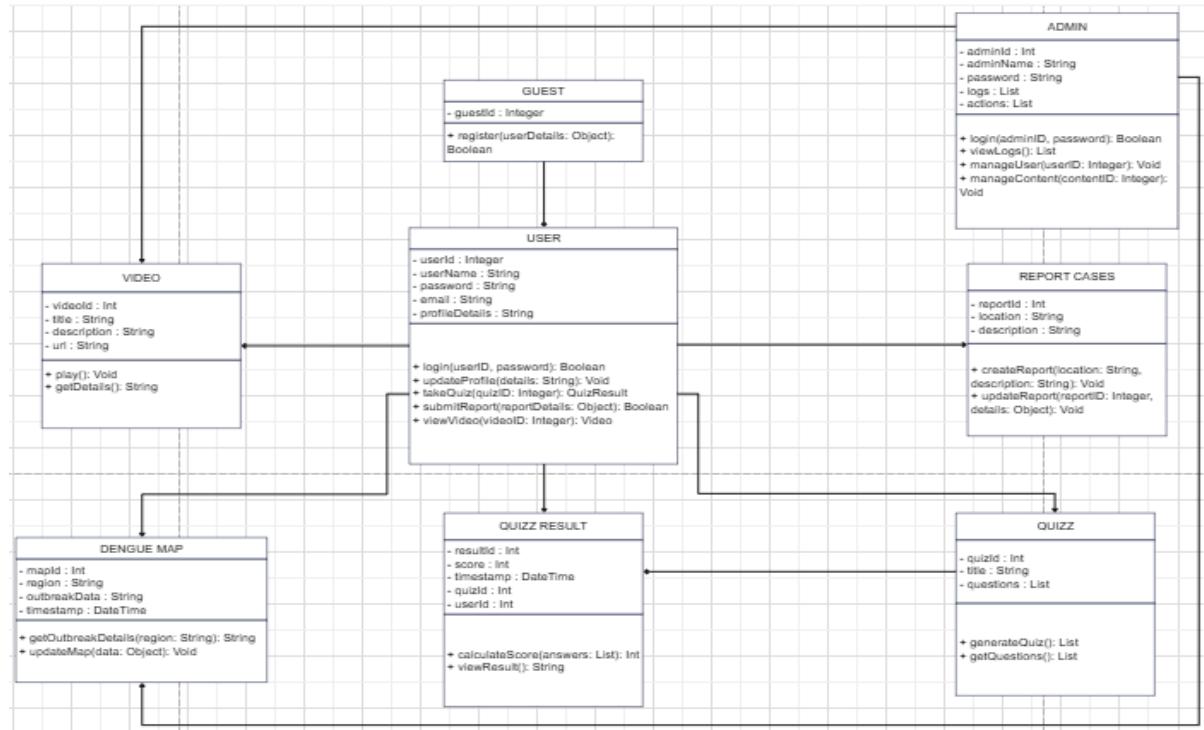


Figure 4.9: Class diagram for <DPCM System>

The class diagram illustrates the logical structure of the DPCM system, highlighting key entities, their attributes, methods, and relationships. The Admin class oversees system operations, managing users and content with methods like login(), viewLogs(), and manageContent(). The Guest class represents unauthenticated users who can register to become a User, which includes attributes like userId and methods such as login(), updateProfile(), takeQuiz(), submitReport(), and viewVideo(). Content is managed through the Video and Dengue Map classes, which provide functionalities to view and update outbreak data or educational resources. The Quiz and Quiz Result classes facilitate user engagement, allowing quizzes to be generated, answered, and scored, while the Report Cases class enables users to create and update reports on dengue outbreaks. Relationships among these classes demonstrate seamless interaction, with Admin managing reports, User interacting with quizzes, videos, and maps, and Quiz Result connecting user performance with specific quizzes.

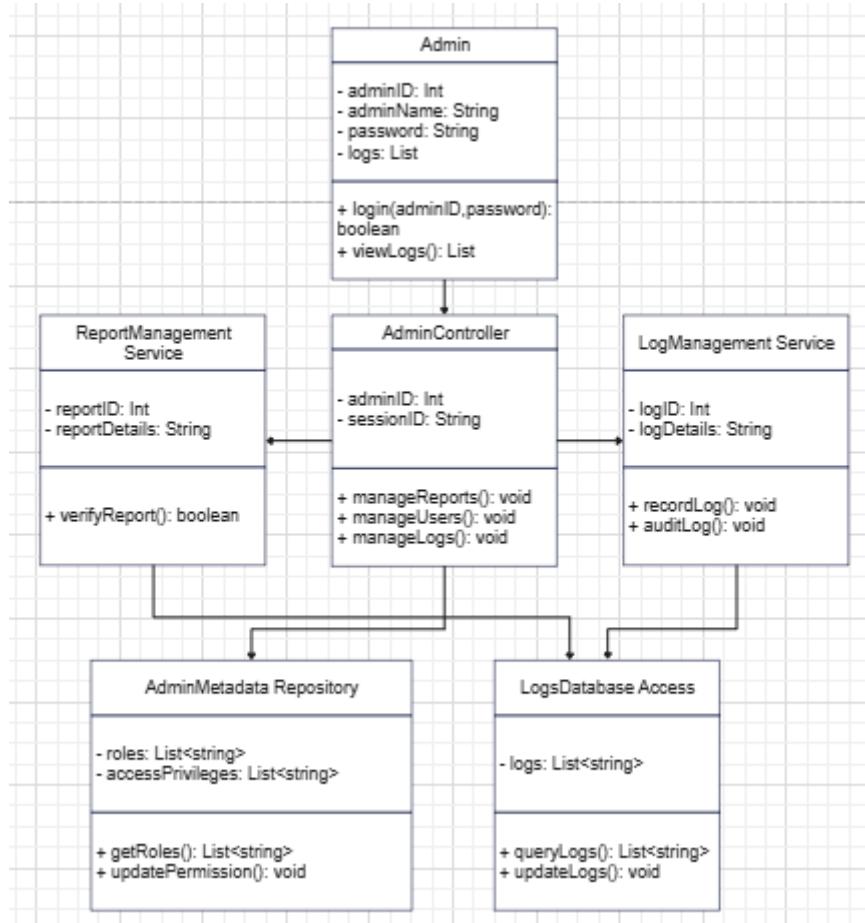


Figure 4.10: Class diagram for <Admin Subsystem>

The Admin Subsystem provides administrative tools for managing reports, logs, and users. The AdminController acts as the intermediary between the dashboard and backend services, handling requests such as report processing and log management. The Admin class represents individual admin users with attributes like adminID, name, and password, and methods for logging in, logging out. The ReportManagementService processes user-submitted reports, while the LogManagementService audits and records system logs. Data is stored and accessed through the LogsDatabaseAccess, which manages log queries and updates, and the AdminMetadataRepository, which handles admin roles and permissions to ensure secure access control.

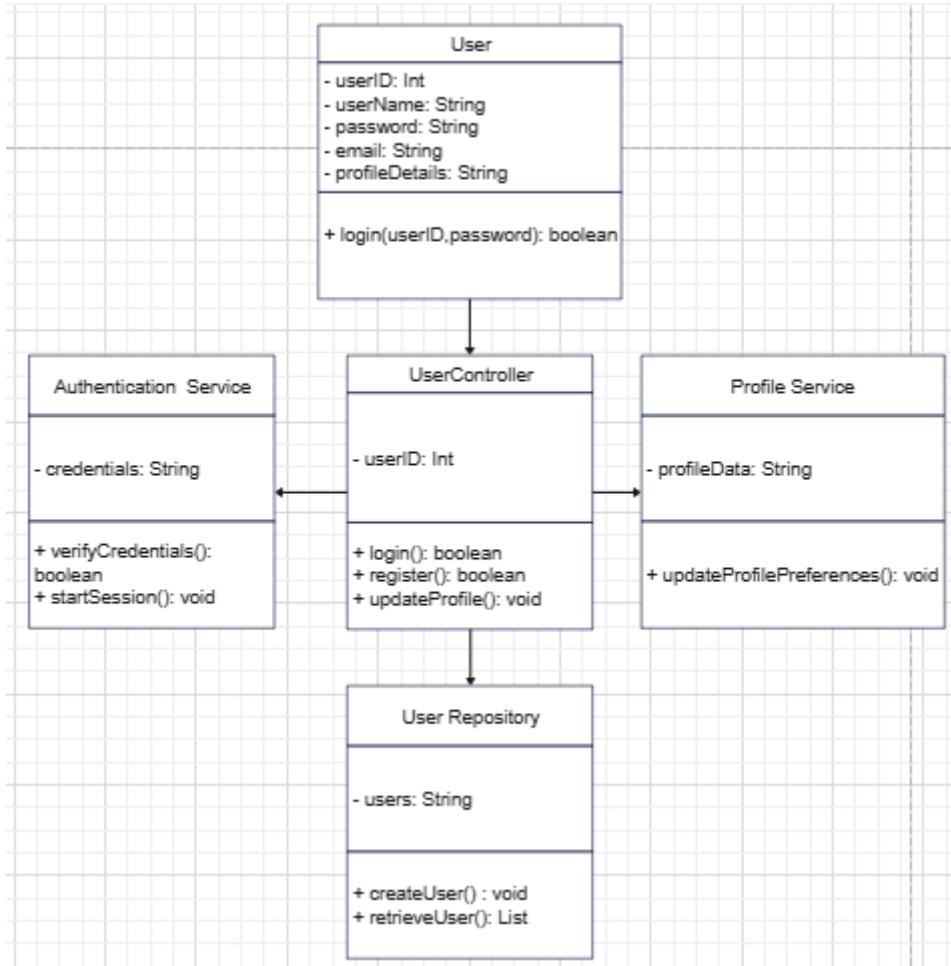


Figure 4.11 Class diagram for <User Management Subsystem>

The User Management Subsystem handles user account operations like login, registration, and profile management. The UserController bridges the interface with backend services. The User class represents individual system users, storing attributes like userID, name, and email, with methods for managing their account and profile data. The AuthenticationService validates user credentials and manages session initiation, while the ProfileService facilitates profile updates and preference management. The UserRepository serves as the data layer, storing and retrieving user information to support these operations.

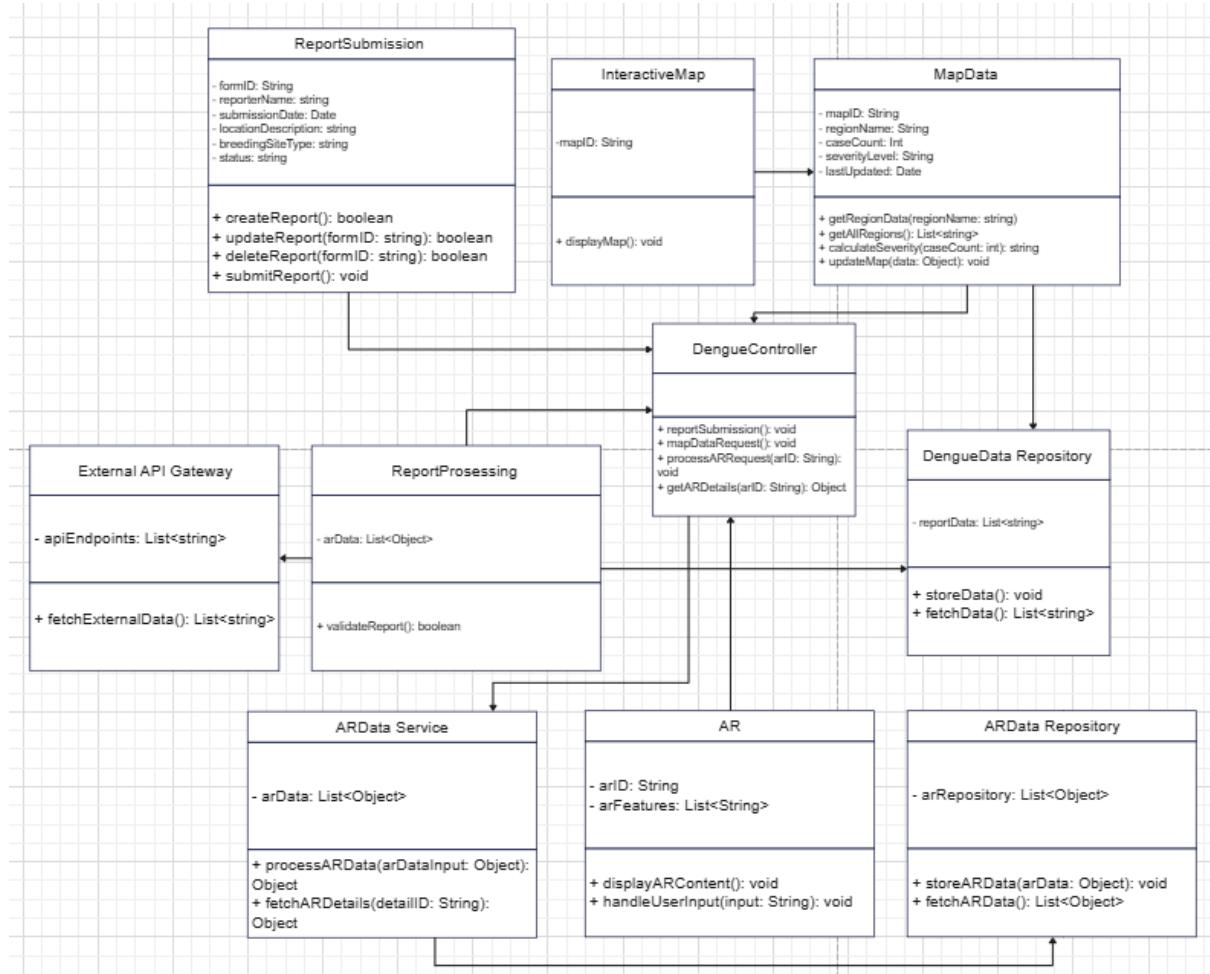


Figure 4.12 Class diagram for <Dengue Information Subsystem>

The Dengue Information Subsystem focuses on providing real-time dengue statistics, processing reports submitted by users, and managing augmented reality (AR) features for enhanced user interaction. The InteractiveMap visualizes dengue trends and statistics, while the ReportSubmission allows users to report mosquito breeding sites. Additionally, the AR offers an immersive way to visualize dengue-related data using augmented reality. The DengueController processes requests related to report submissions, map data, and AR content. Reports are validated and categorized by the ReportProcessingService, while the MapDataService and ARDataService prepare data for visualization and AR content delivery, respectively. The DengueDataRepository serves as the main data store for dengue-related statistics and reports, while the ARDataRepository manages AR-specific data. The ExternalAPIGateway integrates third-party data sources to enhance the system's coverage and accuracy.

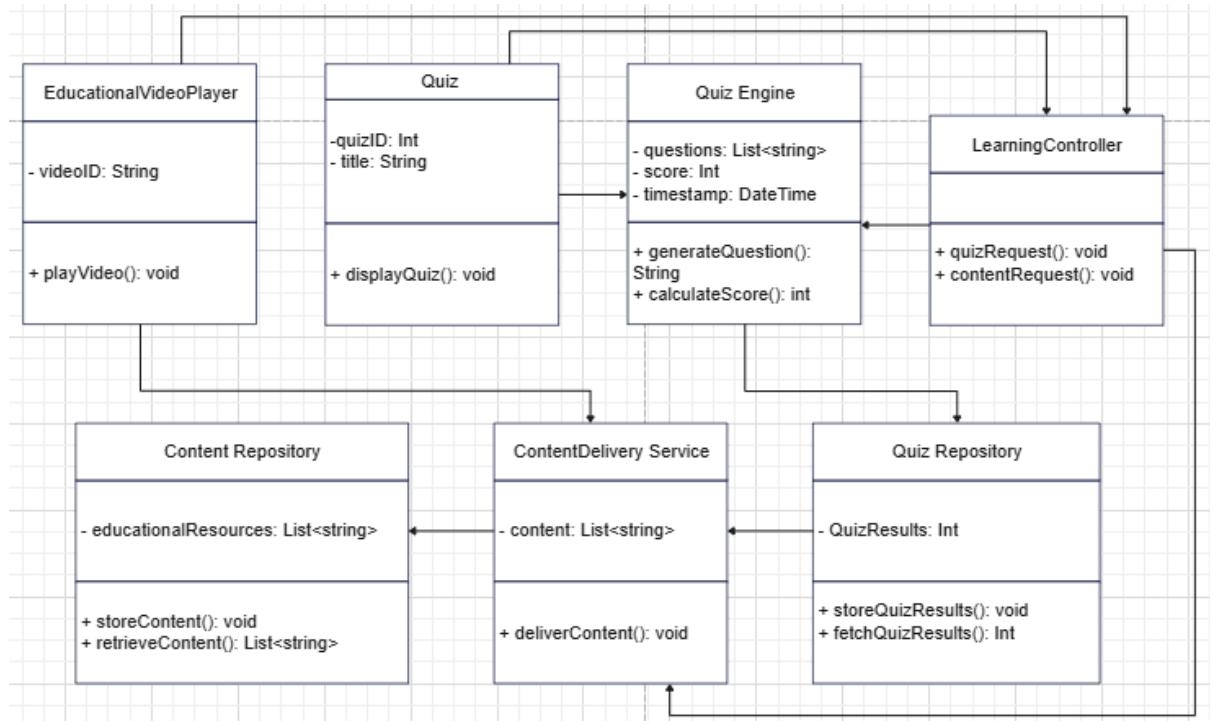


Figure 4.13 Class diagram for <Learning & Engagement Subsystem>

The Learning & Engagement Subsystem provides educational resources and interactive quizzes to engage users. The Quiz offers an interface for users to take quizzes, while the EducationalVideoPlayer presents dengue-related videos. The LearningController handles requests related to quizzes and content delivery. The QuizEngine generates quiz questions and calculates scores, and the ContentDeliveryService manages the distribution of educational resources. Data is stored in the QuizRepository, which tracks quiz questions and results, and the ContentRepository, which maintains educational videos and resources for user access.

4.4 Process View

By focusing on runtime behaviour, component interactions, and the sequence in which certain activities are completed, the Process View in software architecture illustrates the dynamic aspects of a system. It draws attention to the interconnections between system components in order to satisfy functional requirements and ensure efficient runtime execution. The sequence diagram, a part of the Unified Modelling Language (UML), is an essential tool for presenting the Process View. They keep track of information and regulate the movement of different components or items inside a system throughout time. For this reason, we employ it to provide our DPCM system a better understanding.

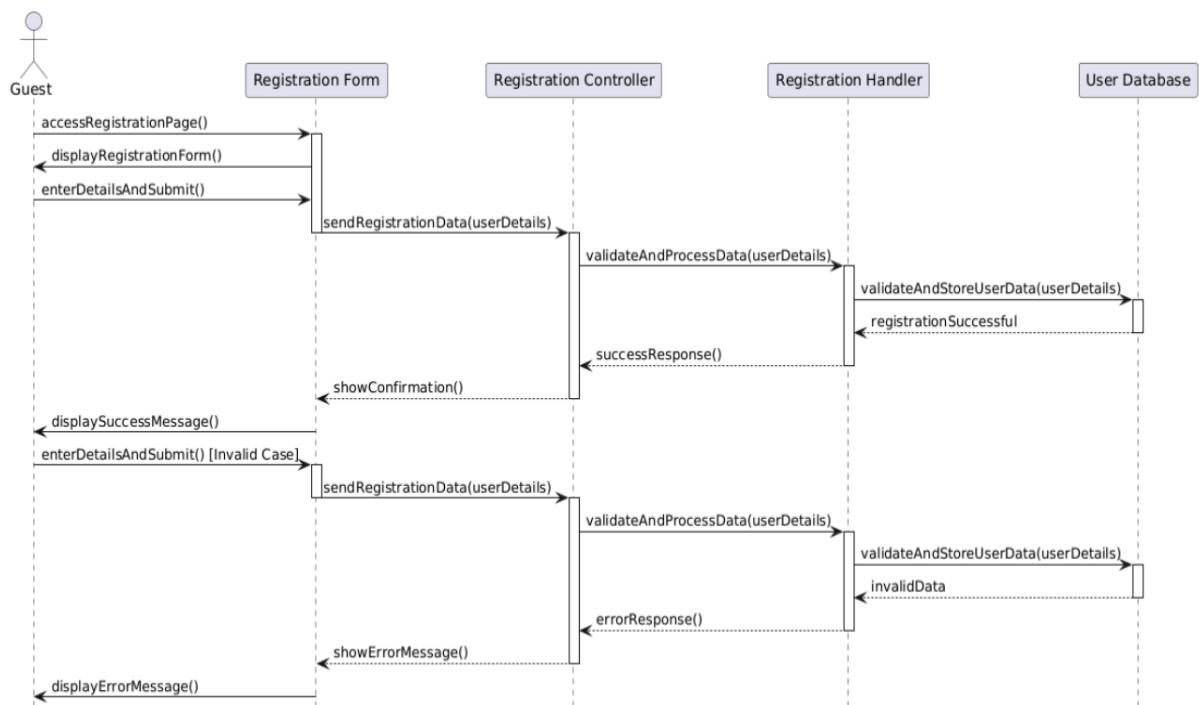


Figure 4.14: Shows SD001 Guest Registration

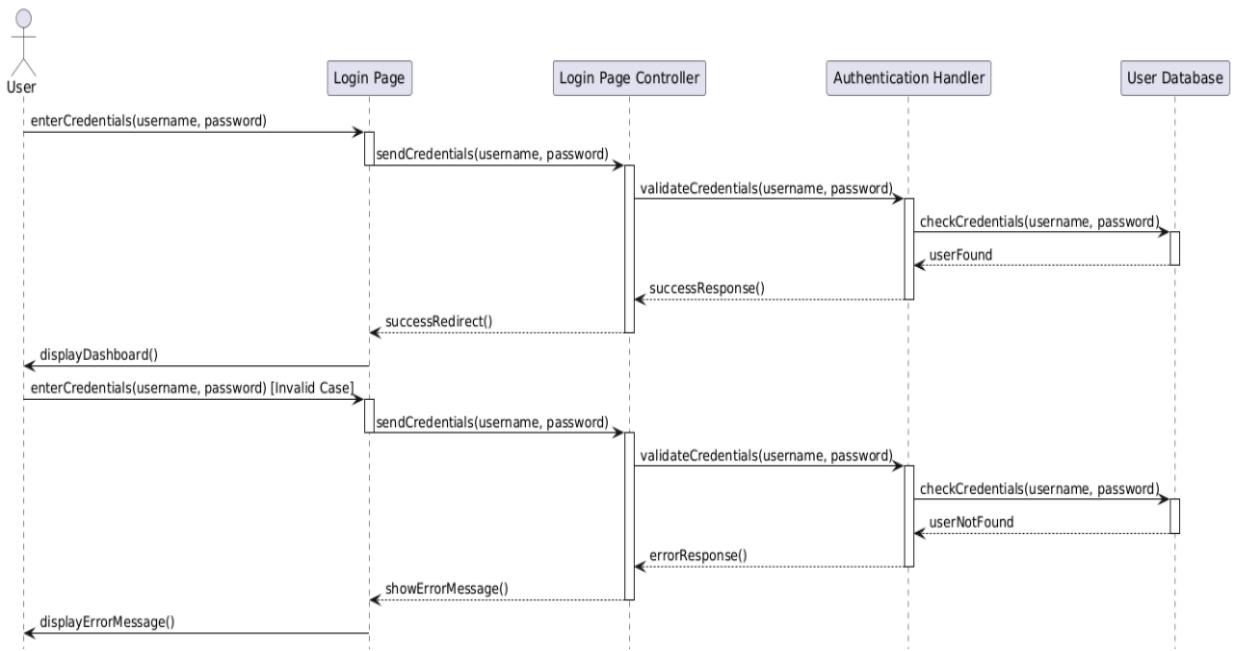


Figure 4.15: Shows SD002 User Login

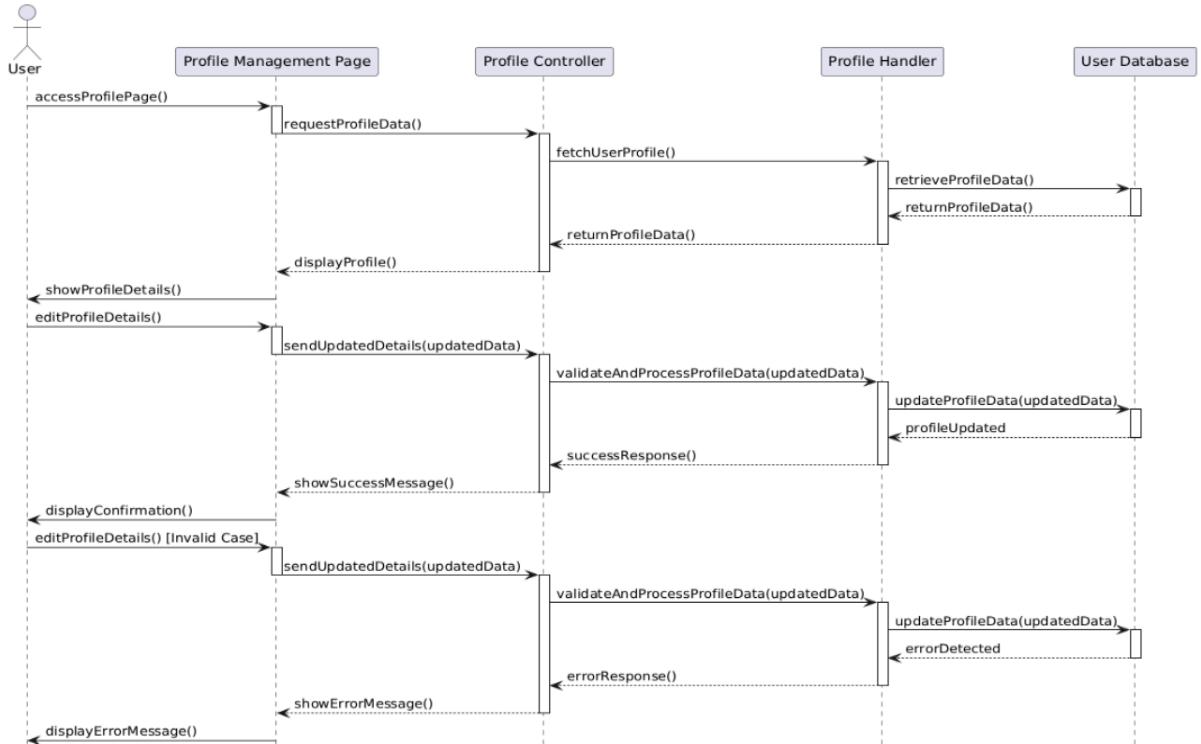


Figure 4.16: Shows SD003 User Manage Profile

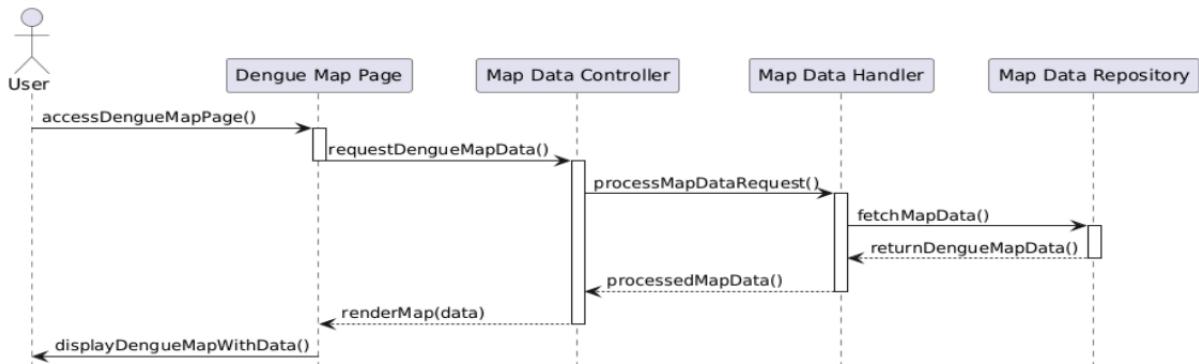


Figure 4.17: Shows SD004 User View Dengue Map Data

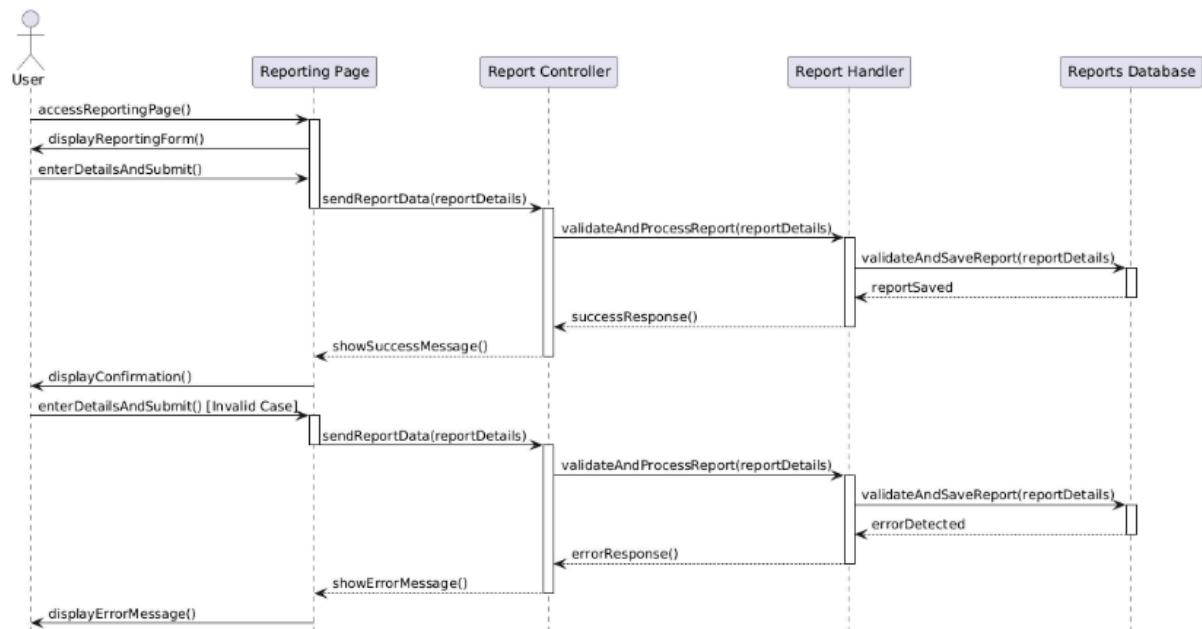


Figure 4.18: Shows SD005 User Report Breeding Sites

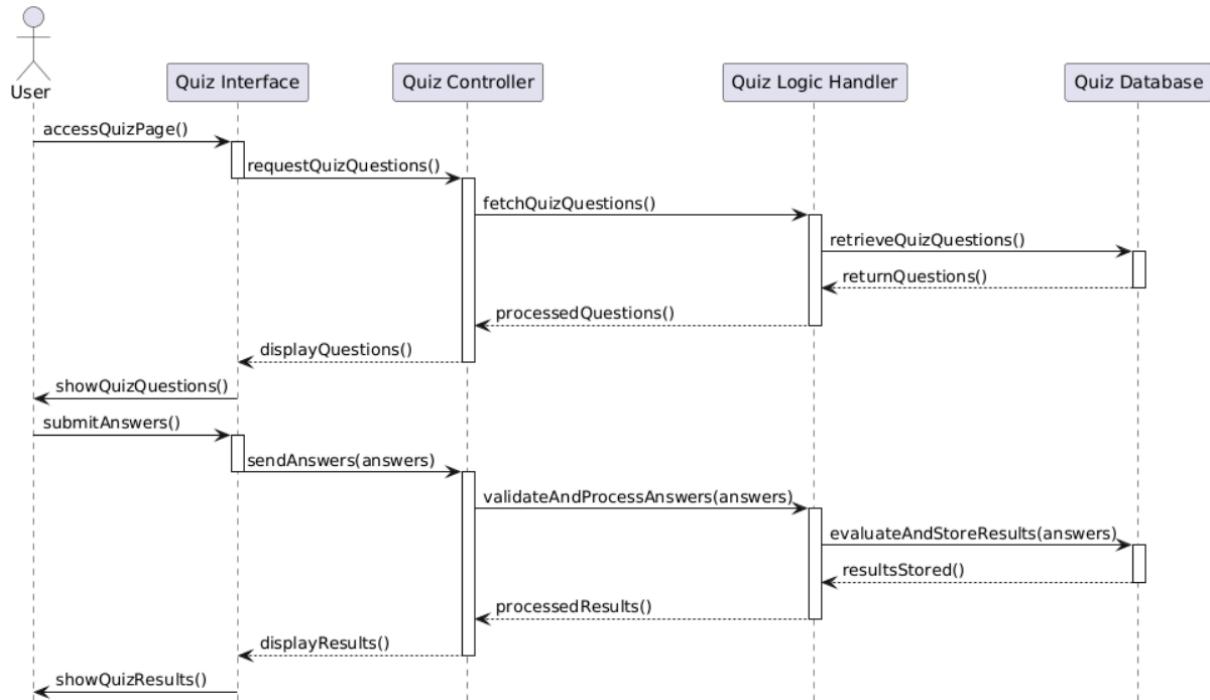


Figure 4.19: Shows SD006 User Take Quizzes And View Quizzes Result

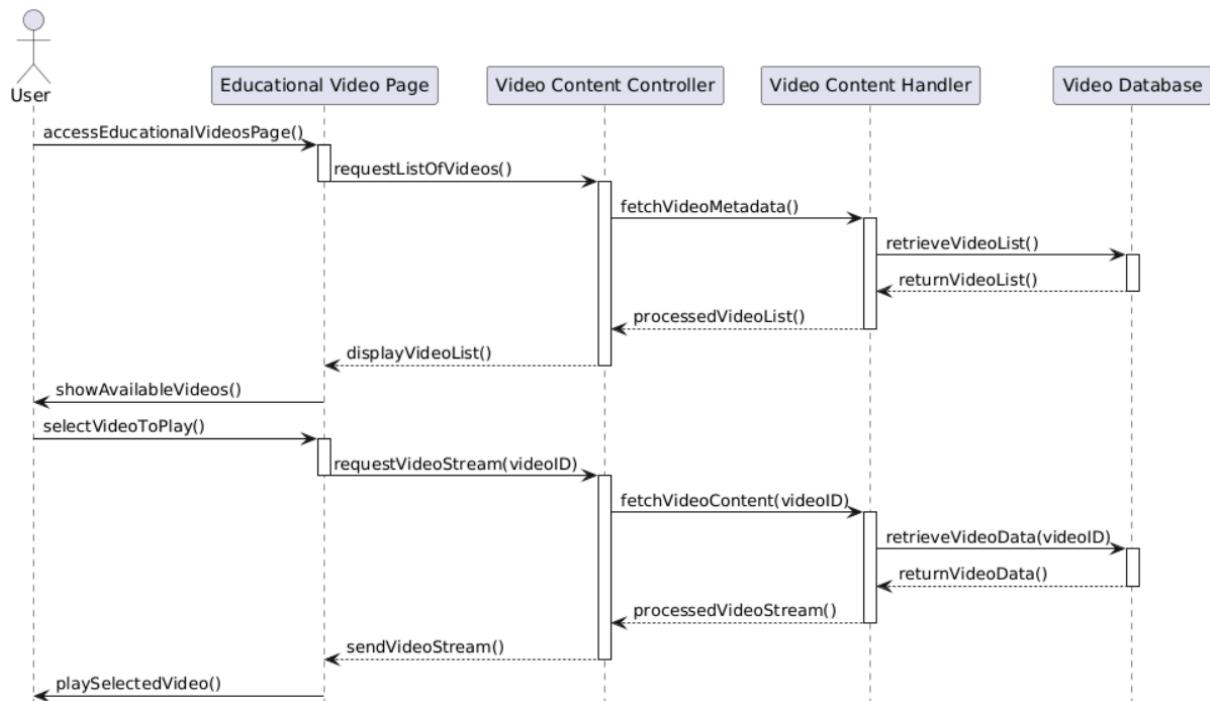


Figure 4.20: Shows SD007 User View Educational Video

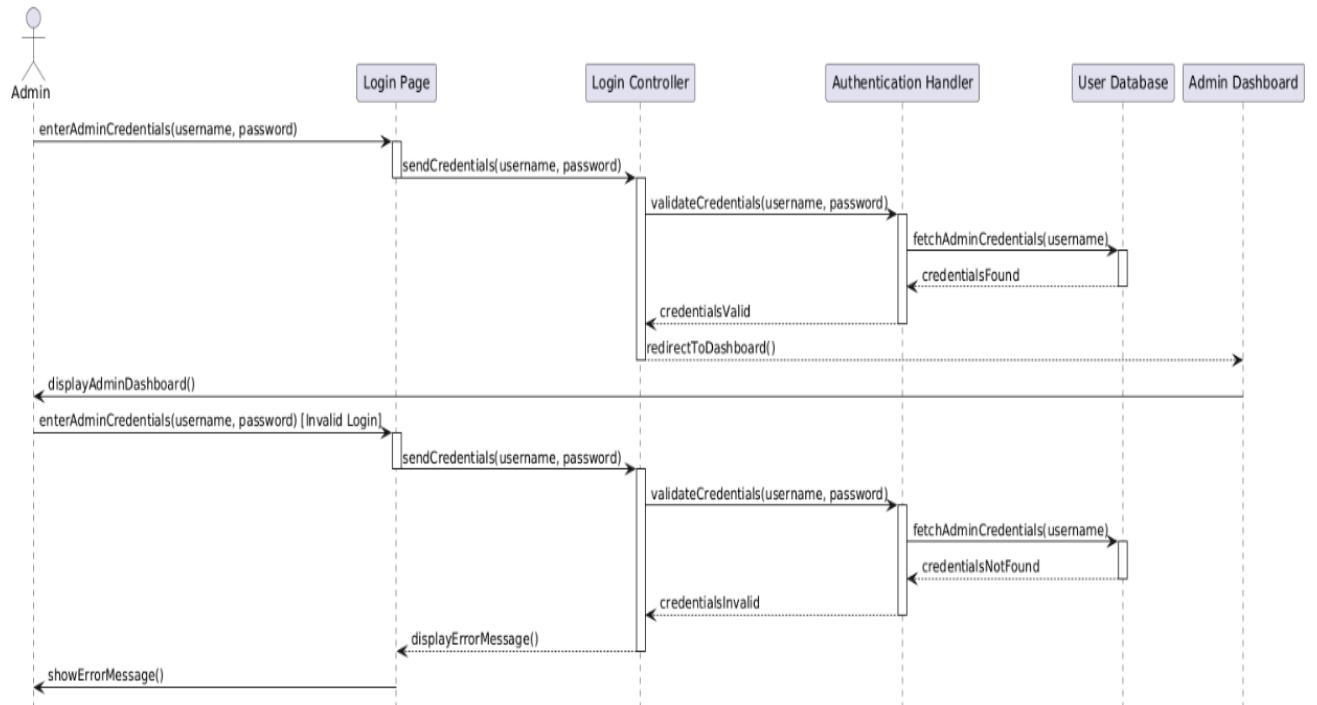


Figure 4.21: Shows SD008 Admin Login To Admin Dashboard

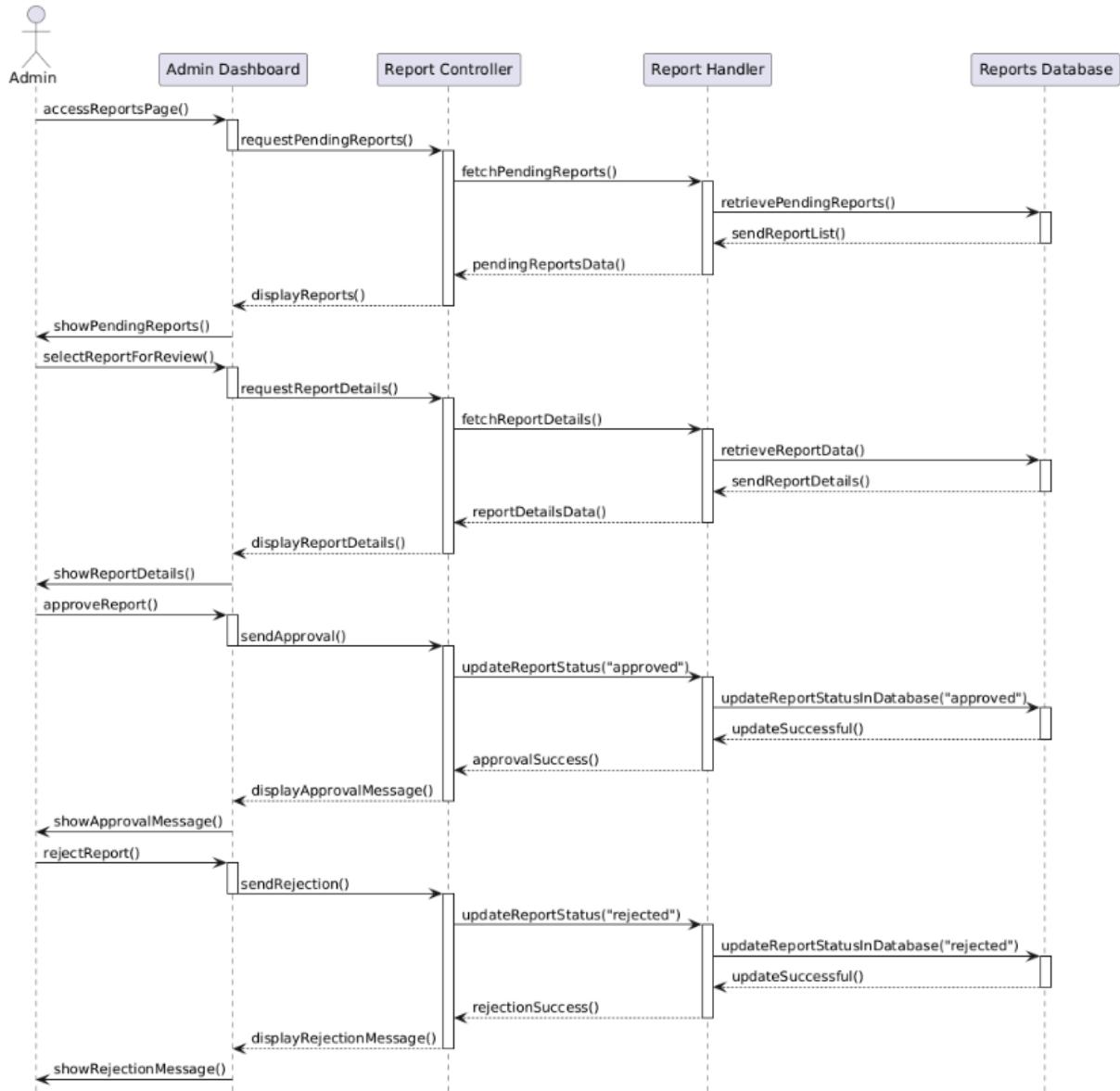


Figure 4.22: Shows SD009 Admin Verify User Reports



Figure 4.23: Shows SD010 Admin Update Dengue Map Data

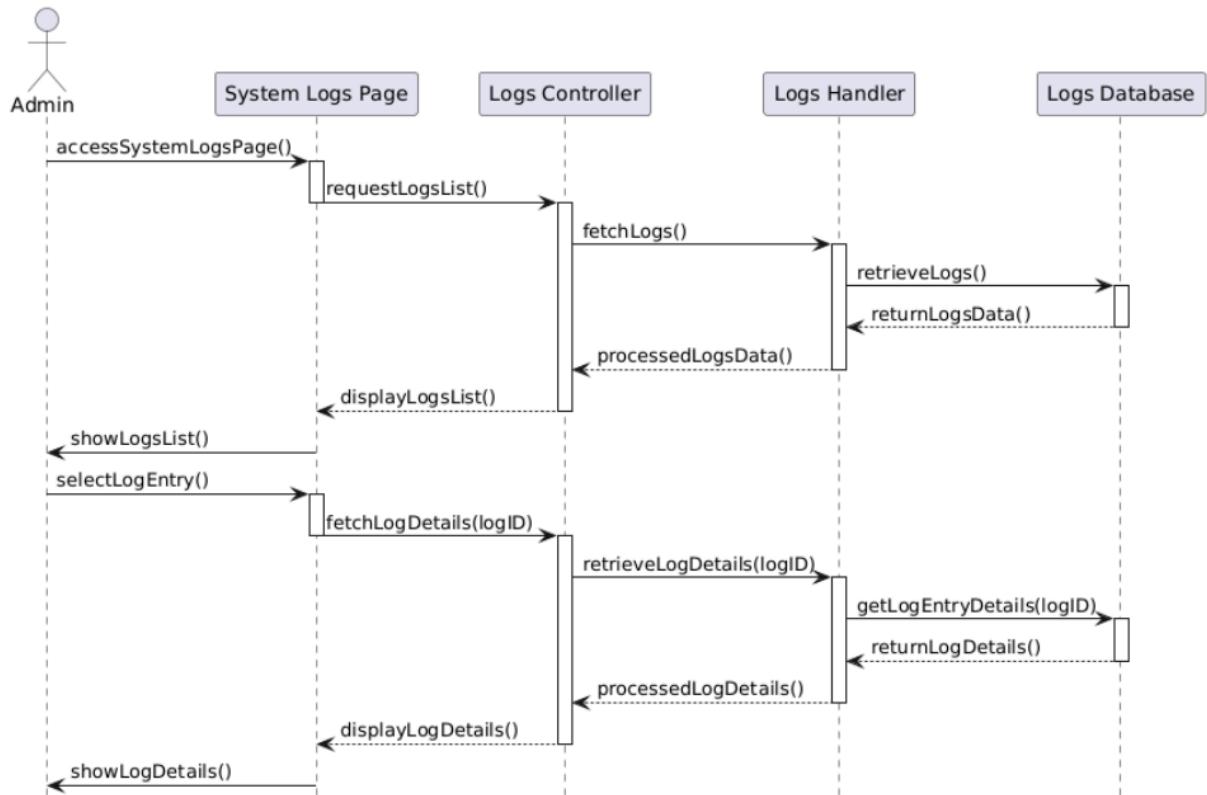


Figure 4.24: Shows SD011 Admin Monitor System Logs

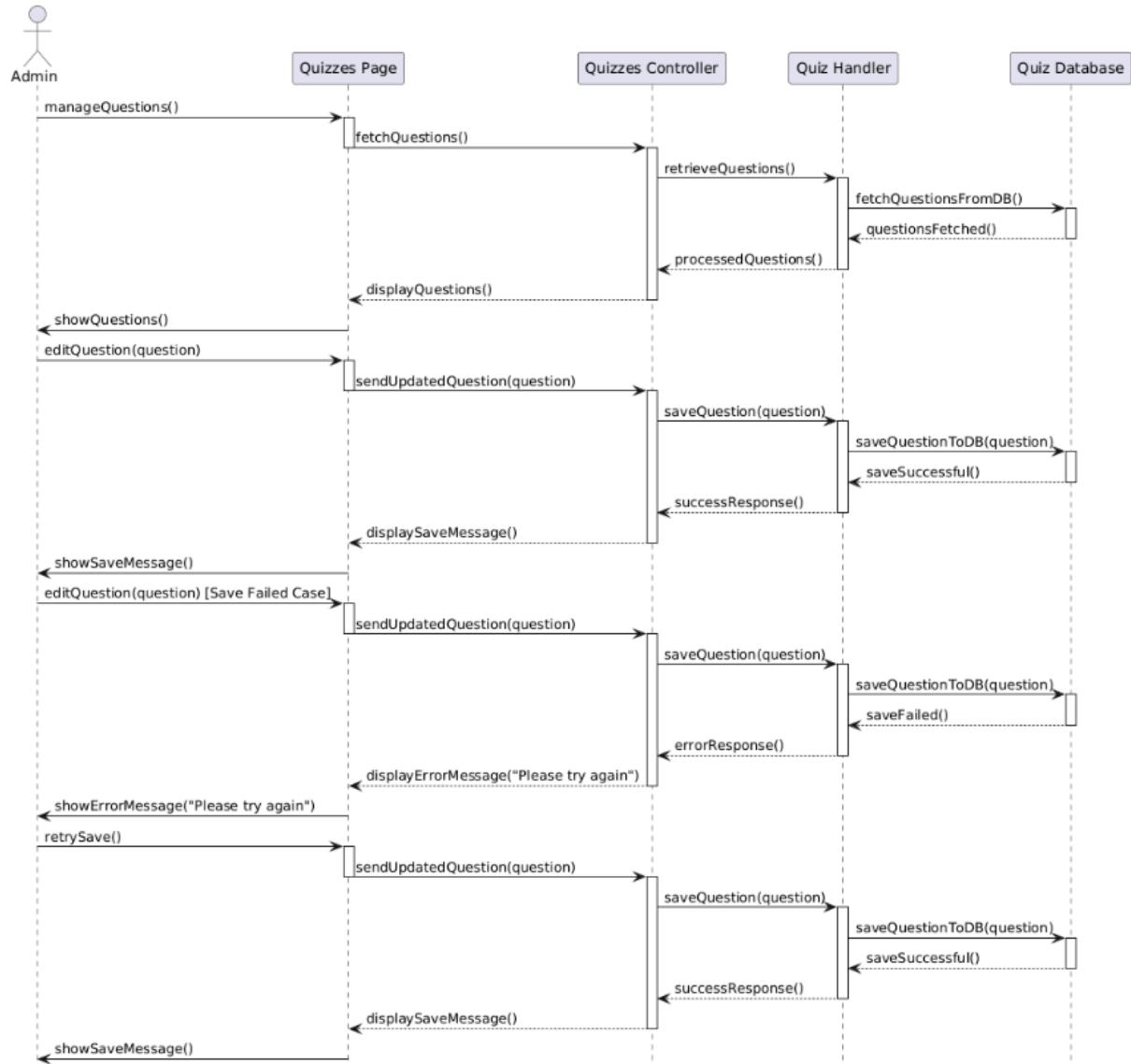


Figure 4.25: Shows SD012 Admin Manage Quizzes

4.5 Physical View

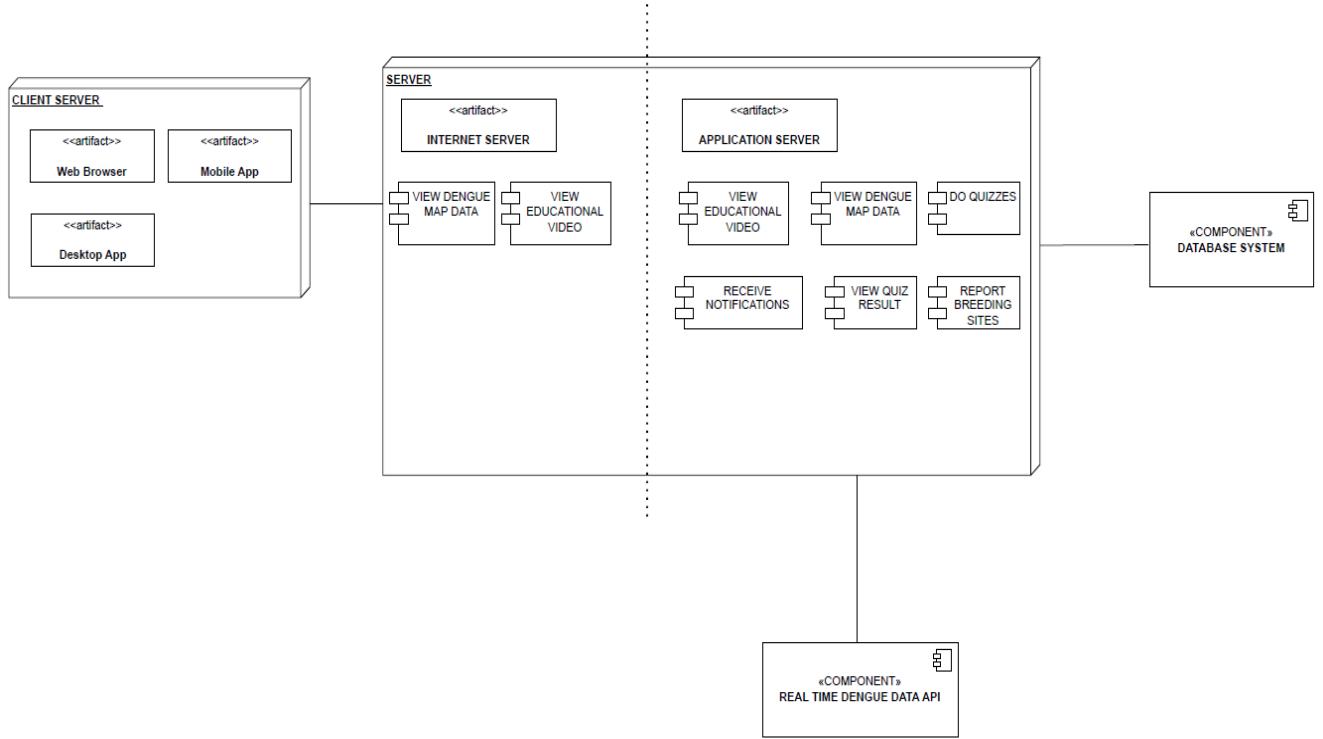


Figure 4.26: Deployment diagram for DPCM SYSTEM

Client-side components, server-side components, and third-party services make up the system's three primary tiers. Together, these layers provide consumers across several platforms with interactive features, instructional materials, and real-time data. A vibrant, approachable setting for increasing dengue awareness and encouraging community involvement is guaranteed by the design.

Client-Side Components:

The system supports three types of client devices:

- 1. web browsers,**
- 2. mobile applications**
- 3. desktop applications.**

Server-Related Elements:

There are two main parts to the server-side architecture:

- **The Internet server:** Often known as the Web server, serves as a bridge between the backend systems and client devices. It guarantees smooth connectivity with the application server and handles user requests, including those to see dengue maps or submit reports.
- **Application Server:** The system's main business logic is managed by the application server. It handles requests from the web server, gets information from the database, and carries out essential tasks including delivering instructional material, processing reports, and validating quizzes. In order to get real-time epidemic data, the application server additionally interfaces with the third-party dengue data API.

Third-Party Services and Databases:

- **Database System:** Important information is kept in the database, including user details, test scores, learning materials, and breeding locations that have been reported. Users may review their quiz scores or obtain updated instructional materials thanks to this, which also guarantees data preservation.
- **Real-Time Dengue Data API:** Current data on dengue outbreaks is provided via a third-party API. By using this information to update the dengue map and notify users, the system is kept up to date and accurate.

5. Data Design

Table: Description of Entities in the Database

| No. | Entity Name | Description |
|-----|--------------|---|
| 1 | GUEST | Store information about users, such as their guest IDs |
| 2 | USER | Contains user details like user ID, username, password, email and profile data. |
| 3 | ADMIN | Stores admin-specific data, including admin IDs, names, passwords, logs and actions. |
| 4 | VIDEO | Manages information about videos, including video ID, title, description and URL. |
| 5 | QUIZ | Holds data about quizzes, such as quiz ID, title and associated questions. |
| 6 | QUIZ RESULT | Track quiz results with details like result ID, score, timestamp and associated user ID |
| 7 | DENGUE MAP | Maintains dengue outbreak details, including map ID, region, outbreak data and timestamp. |
| 8 | REPORT CASES | Stores report of cases, including report ID, location, and description. |

5.1 Data Dictionary

5.1.1 Entity: <GUEST>

| Attribute Name | Type | Description |
|----------------|---------|----------------------------------|
| guestID | Integer | Unique identifier for the guest. |

5.1.2 Entity: <USER>

| Attribute Name | Type | Description |
|----------------|---------|--|
| userID | Integer | Unique identifier for the user. |
| userName | String | The username of the user. |
| password | String | Password for the user's account. |
| email | String | The user's email address. |
| profileDetails | String | Additional information about the user's profile. |

5.1.3 Entity: <VIDEO>

| Attribute Name | Type | Description |
|----------------|---------|----------------------------------|
| videoID | Integer | Unique identifier for the video. |
| title | String | Title of the video. |
| description | String | Description of the video. |
| url | String | Description of the video. |

5.1.4 Entity: <DENGUE MAP>

| Attribute Name | Type | Description |
|----------------|----------|--|
| mapID | Integer | Unique identifier for the map. |
| region | String | Region associated with the map. |
| outbreakData | String | Description about the dengue outbreak in the region. |
| timestamp | DateTime | Date and time of the map data update. |

5.1.5 Entity: <QUIZ>

| Attribute Name | Type | Description |
|----------------|---------|---|
| quizID | Integer | Unique identifier for the quiz. |
| title | String | The title of the quiz. |
| questions | List | A list of questions included in the quiz. |

5.1.6 Entity: <QUIZ RESULT>

| Attribute Name | Type | Description |
|----------------|----------|--|
| resultID | Integer | Unique identifier for the quiz result. |
| score | Integer | Score achieved by the user in the quiz. |
| timestamp | DateTime | Date and time of the quiz completion. |
| quizID | Integer | Identifier for the associated quiz. |
| userID | Integer | Identifier for the user who took the quiz. |

5.1.7 Entity: <ADMIN>

| Attribute Name | Type | Description |
|-----------------------|-------------|--|
| adminID | Integer | Unique identifier for the admin. |
| adminName | String | Name of the admin. |
| password | String | Password for the admin account. |
| logs | List | List of activity logs for the admin. |
| actions | List | List of actions the admin can perform. |

5.1.8 Entity: <REPORT CASES>

| Attribute Name | Type | Description |
|-----------------------|-------------|-------------------------------------|
| reportID | Integer | Unique identifier for the report. |
| location | String | Location where the report is filed. |
| description | String | Description of the reported case. |

6. User Interface Design

The interface include all the subsystems interface, where there are subsystems for users, including registration page with the email verification, login and logout subsystem, profile page where user can manage and edit their profile, viewing dengue map page, attempting quiz and review their result, reporting mosquito breeding site. And the subsystems for admin where the admin can login to the system, land on the admin dashboard and perform admin tasks such as reviewing dengue reports, managing and reviewing quizzes.

6.1 Guest Registration Subsystem Interface

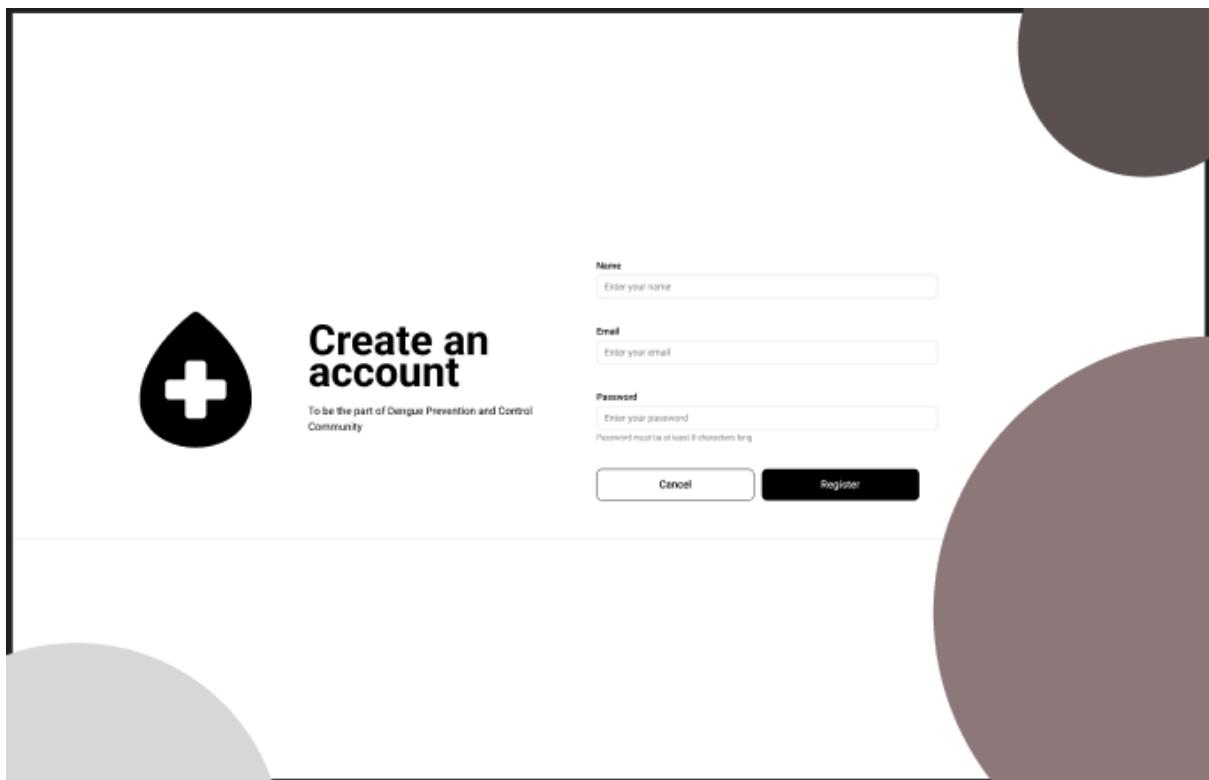


Figure 6.1: Interface for <Registration page>

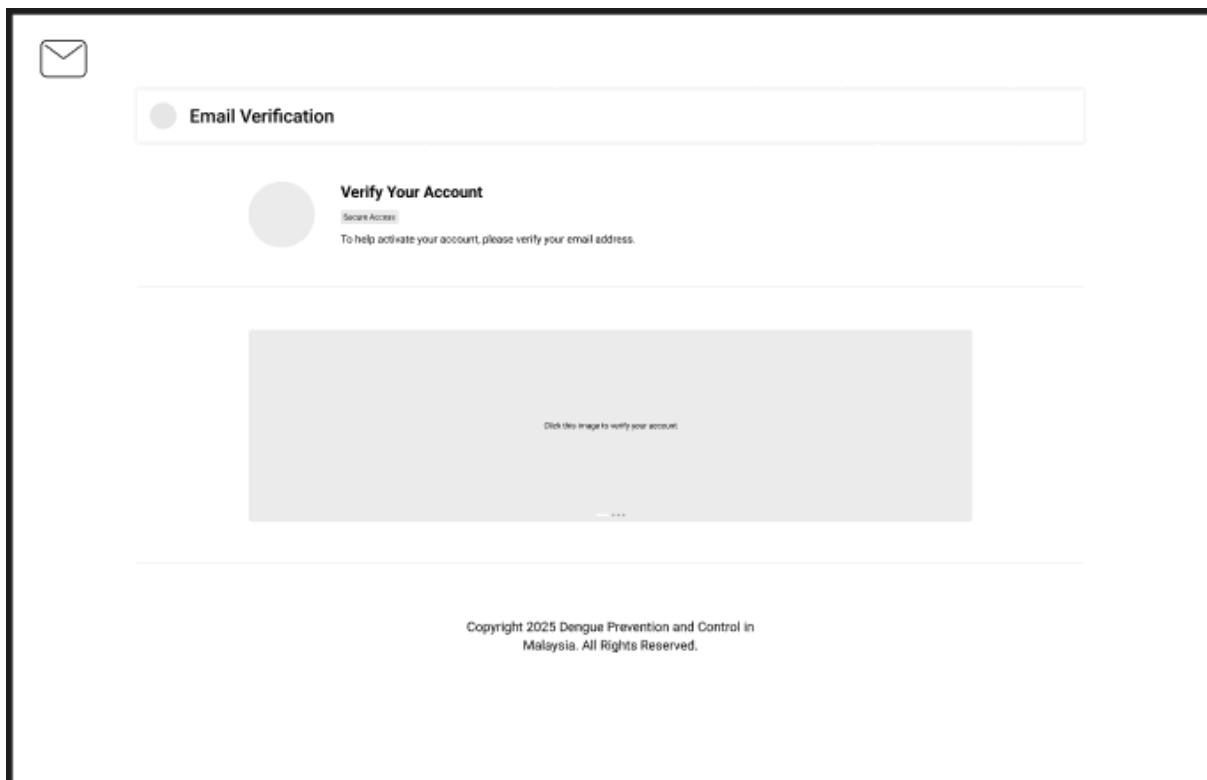


Figure 6.2: Interface for <Registration Guest/Verify Email Page>



Figure 6.3: Interface for <Registration Guest/Confirmation Verify Email Page>

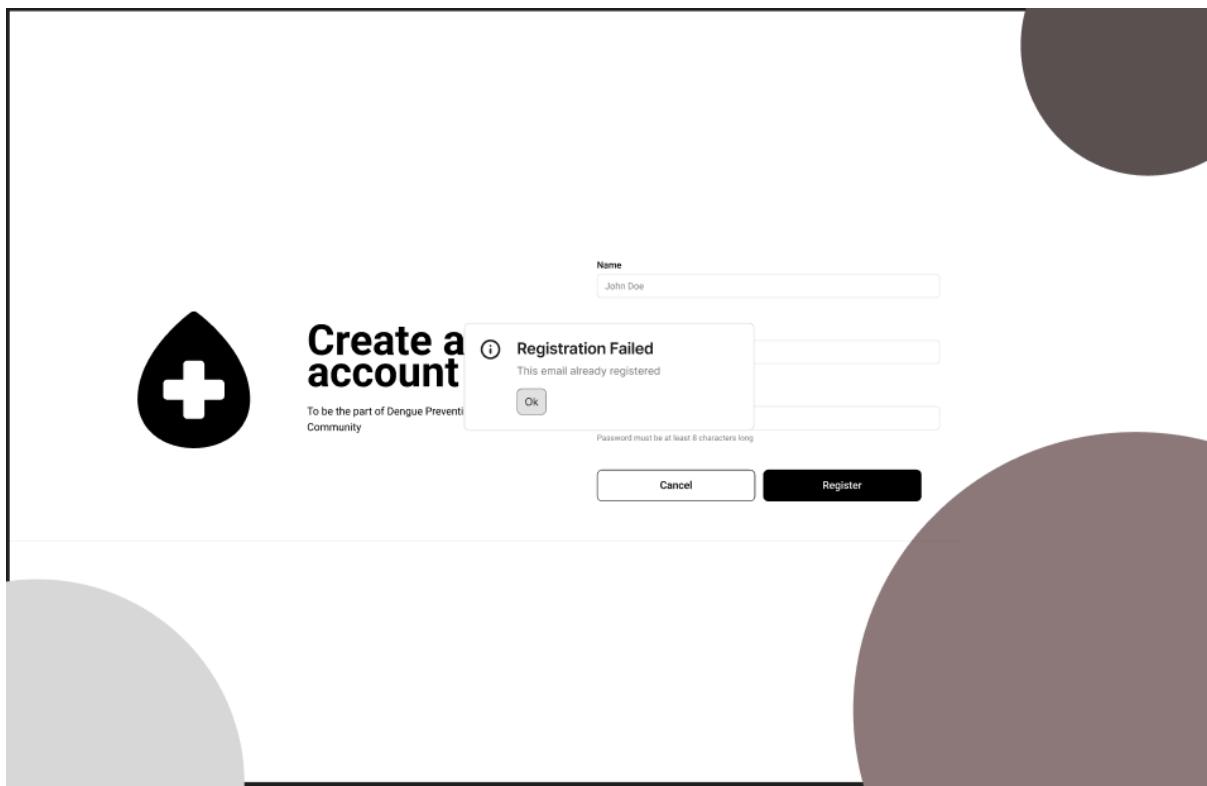


Figure 6.4: Interface for <Guest Registration Page/Failed Registration Error Message>

In this subsystem, users can register their account and get an email verification, if the registration fails, the failed message will appear.

6.2 Customer Login Subsystem Interface

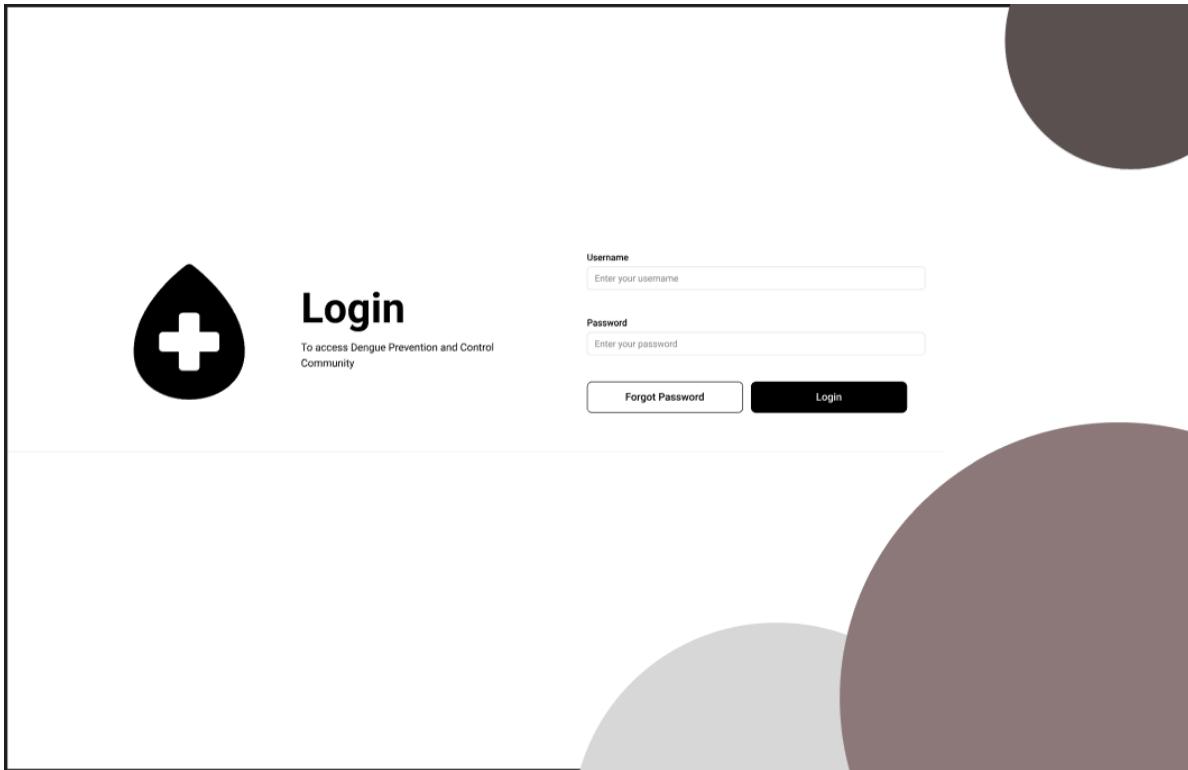


Figure 6.5: Interface for <Login page>

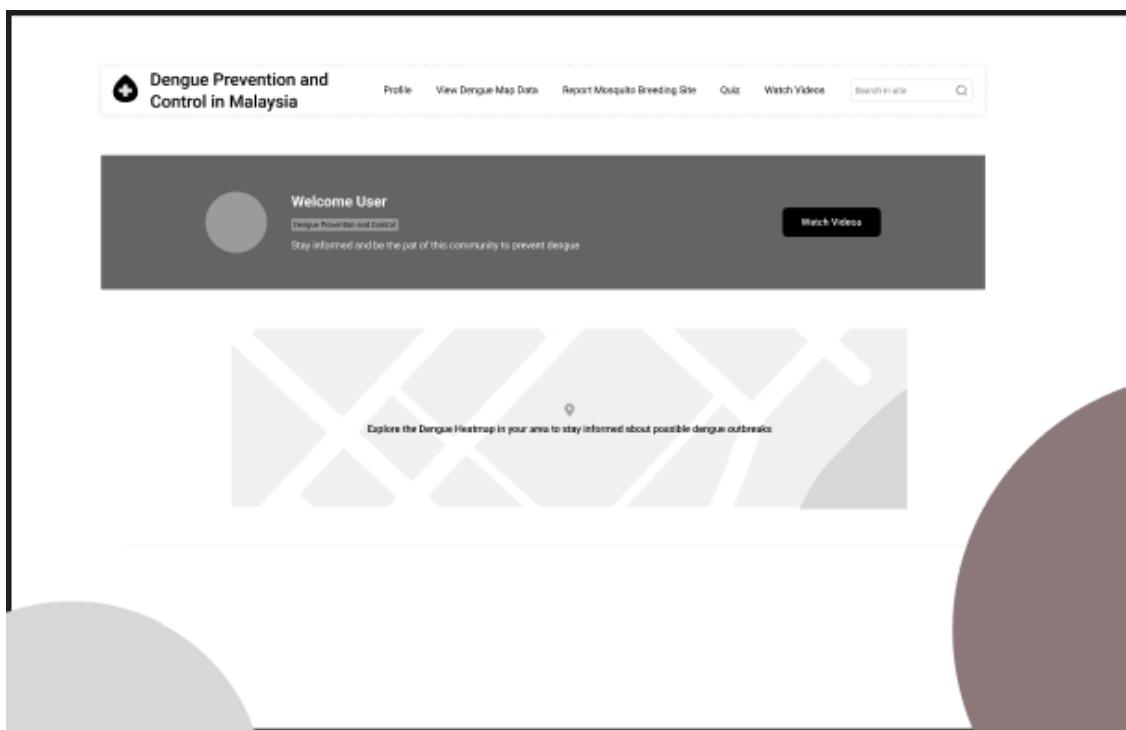


Figure 6.6: Interface for <Homepage>

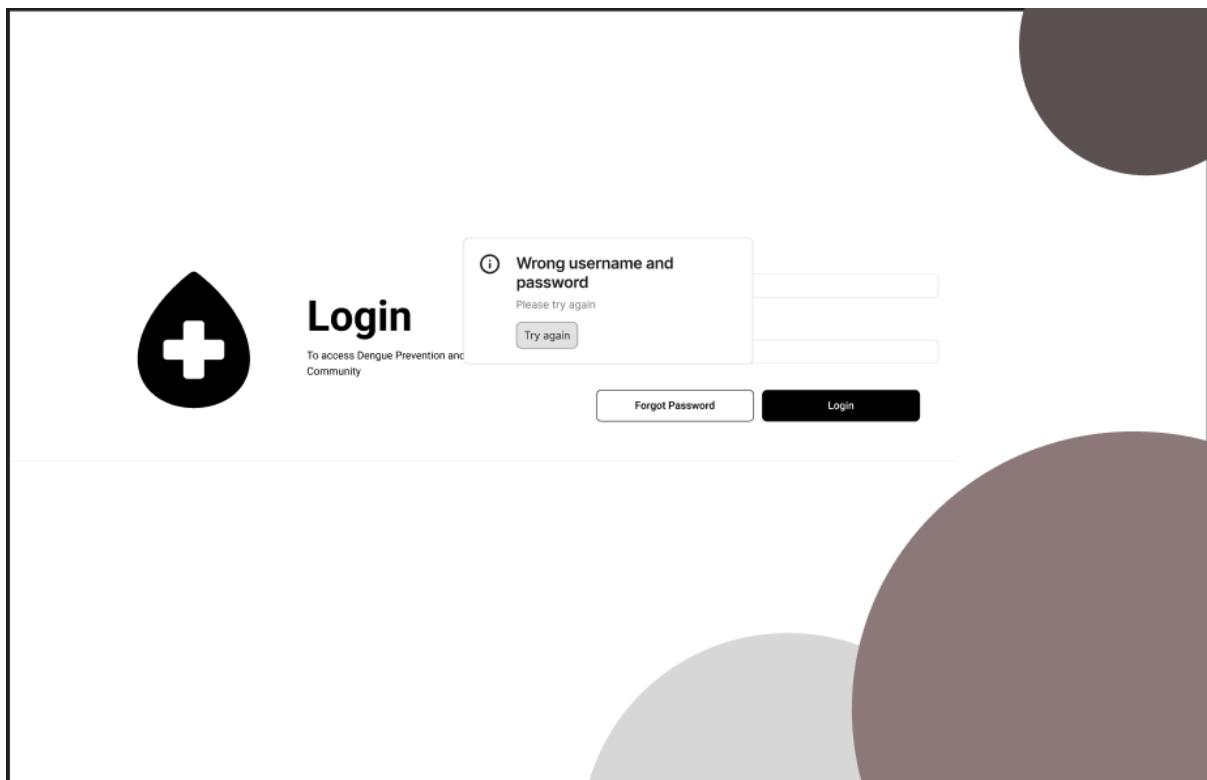


Figure 6.7: Interface for <Login/ Failed Login Error Message>

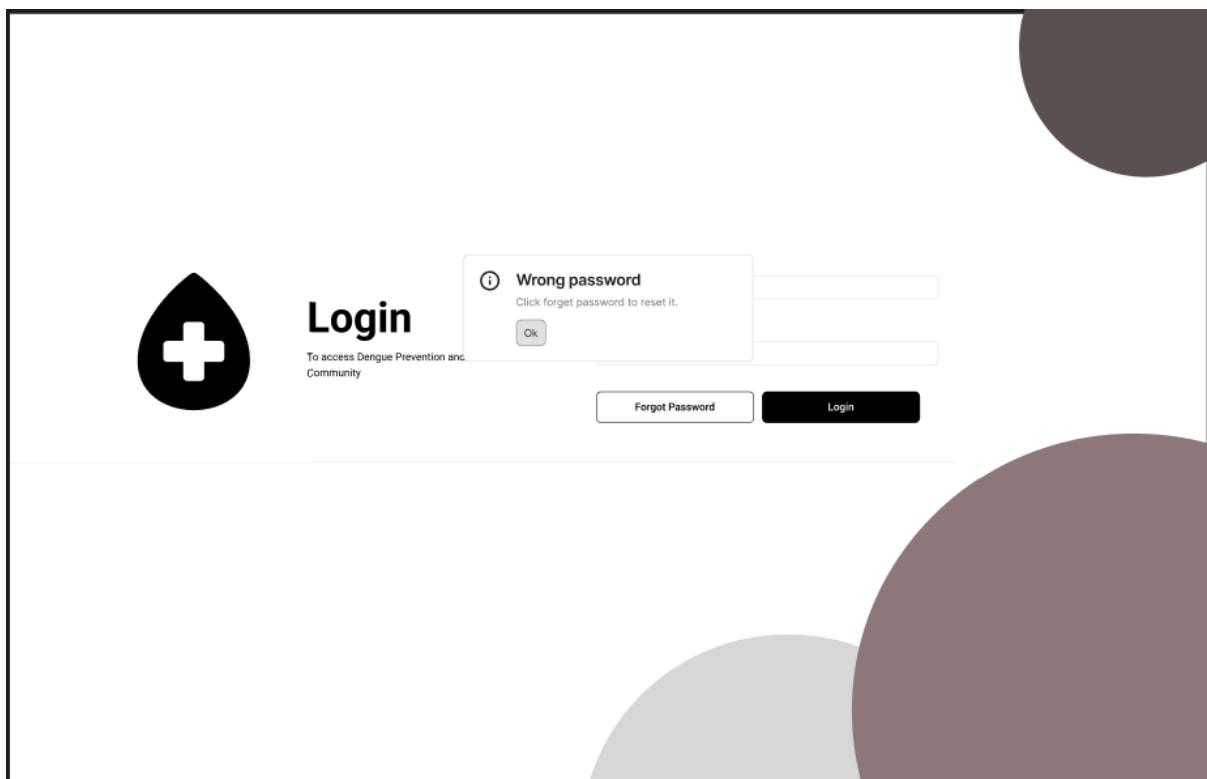


Figure 6.8: Interface for <Login/ Wrong Password Error Message>

In this subsystem, users can login into their account and if it is successful, it will land on the home page, but if the login attempt fails, the failed message will appear or if it fails 5 times, the system will suggest to change password.

6.3 Email Verification Subsystem Interface

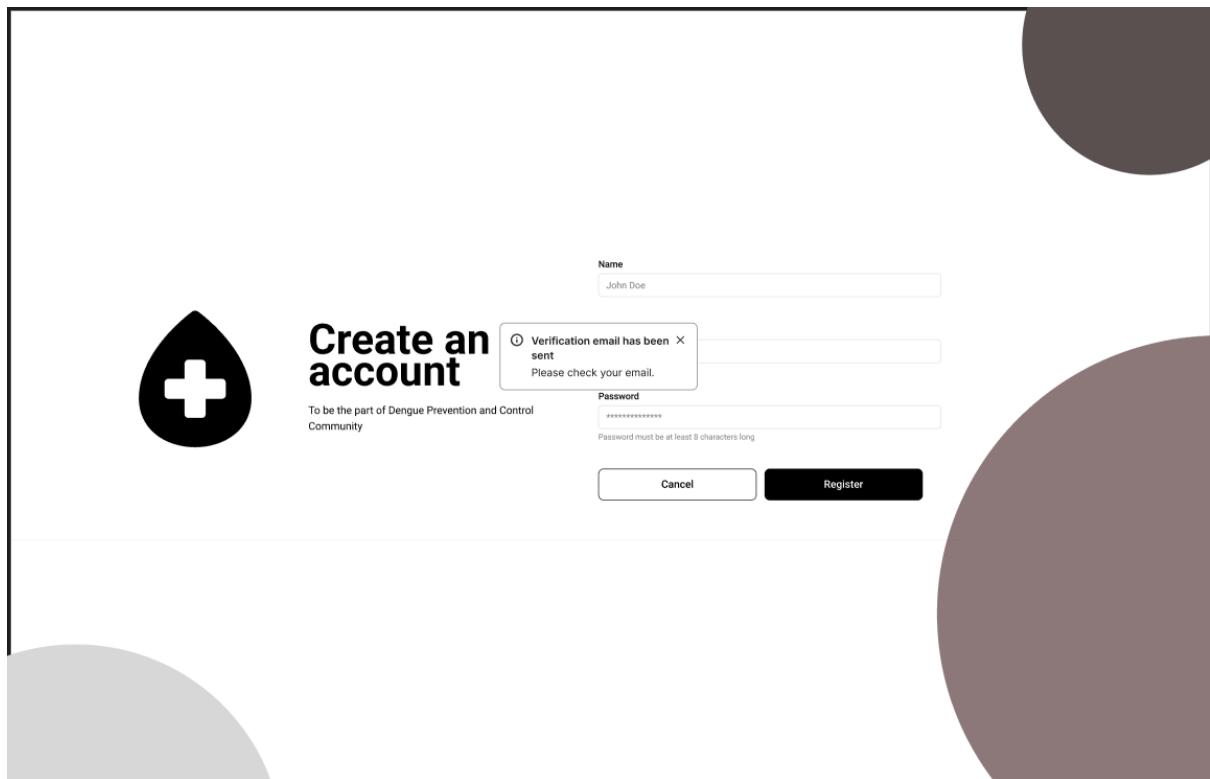


Figure 6.9: Interface for <Email Verification/Confirmation Message box>

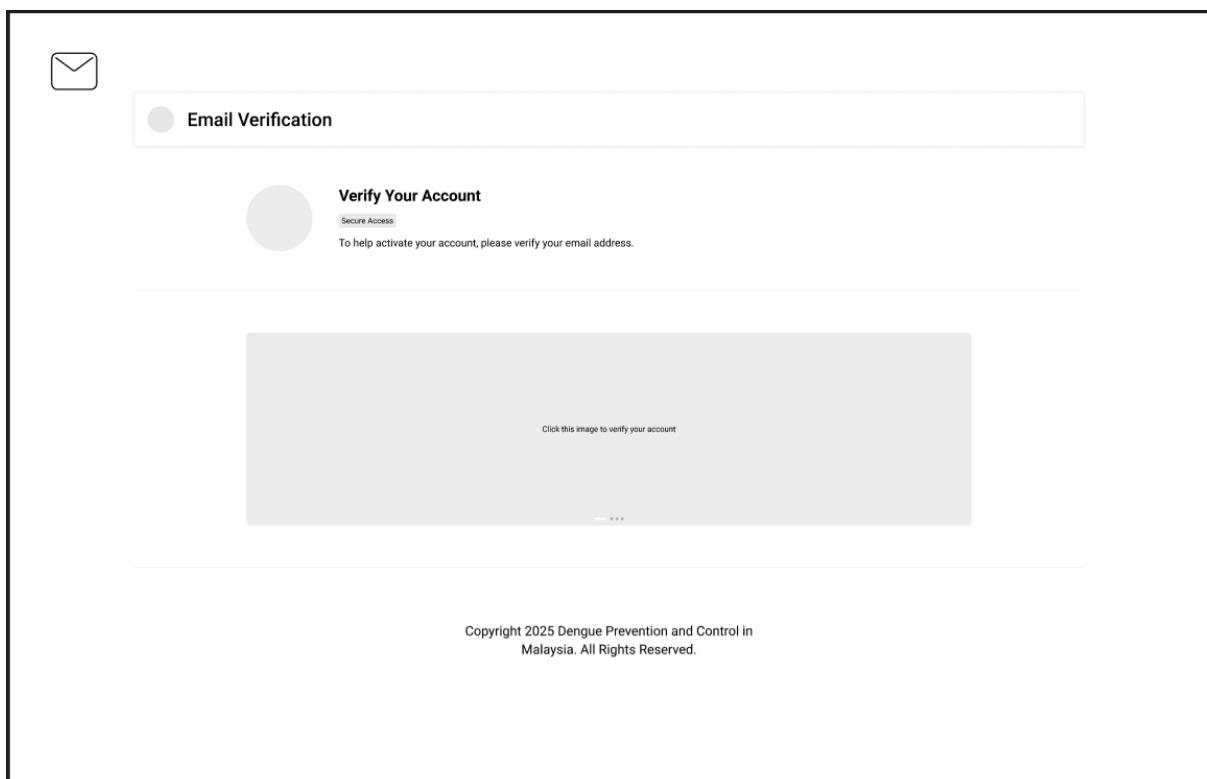


Figure 6.10: Interface for <Email Verification page>



Figure 6.11: Interface for <Email Verification/ Confirmation page>



DENGUE PREVENTION AND CONTROL IN MALAYSIA

Link Expired
Please click resend options
[Resend Email](#)

Figure 6.12: Interface for <Email verification/ Link Expired>



DENGUE PREVENTION AND CONTROL IN MALAYSIA

Error has Occured
Please try again

Figure 6.13: Interface for <Email verification/ Error Occurred>

In this subsystem, users can verify their account via email verification, if the verifying attempt is successful, it will be redirected to the DPCM and a successful message appears, but if the verifying attempt is unsuccessful or the link has expired, the failed message will appear.

6.4 Customer Logout Subsystem Interface

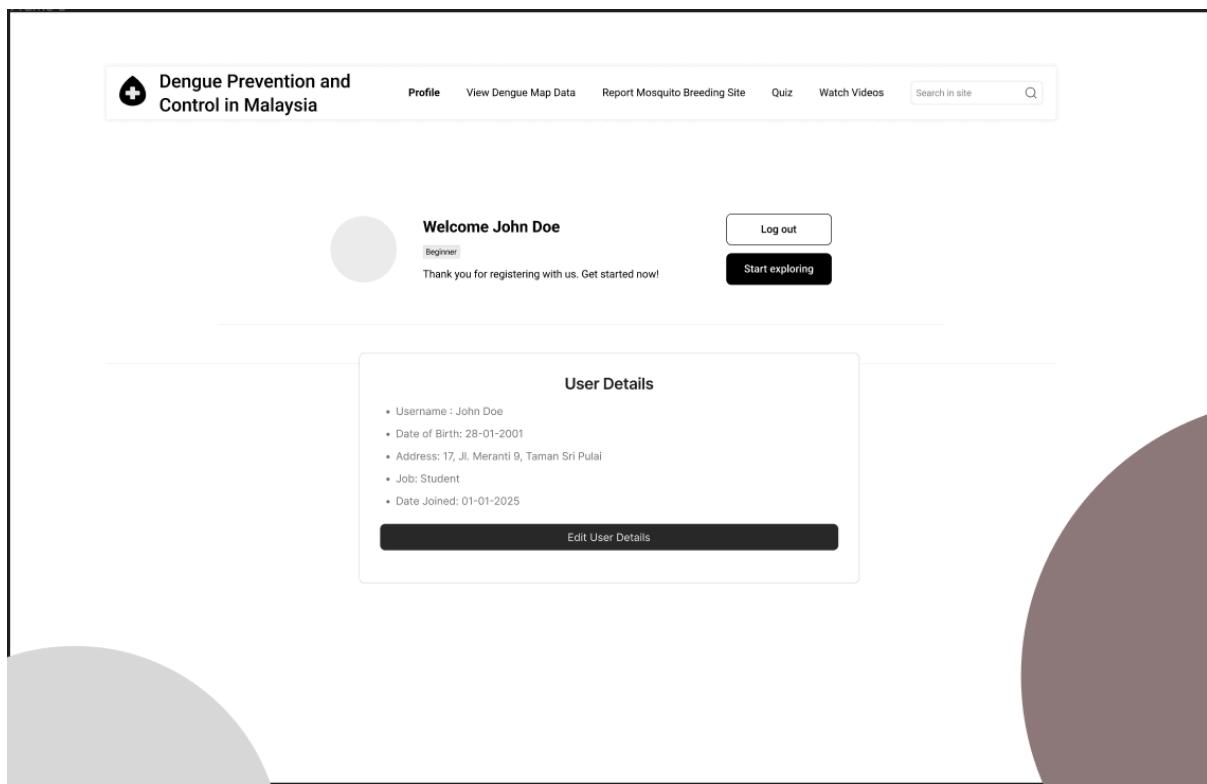


Figure 6.14: Interface for <Customer Log Out>



DENGUE PREVENTION AND CONTROL IN MALAYSIA

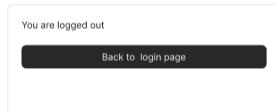
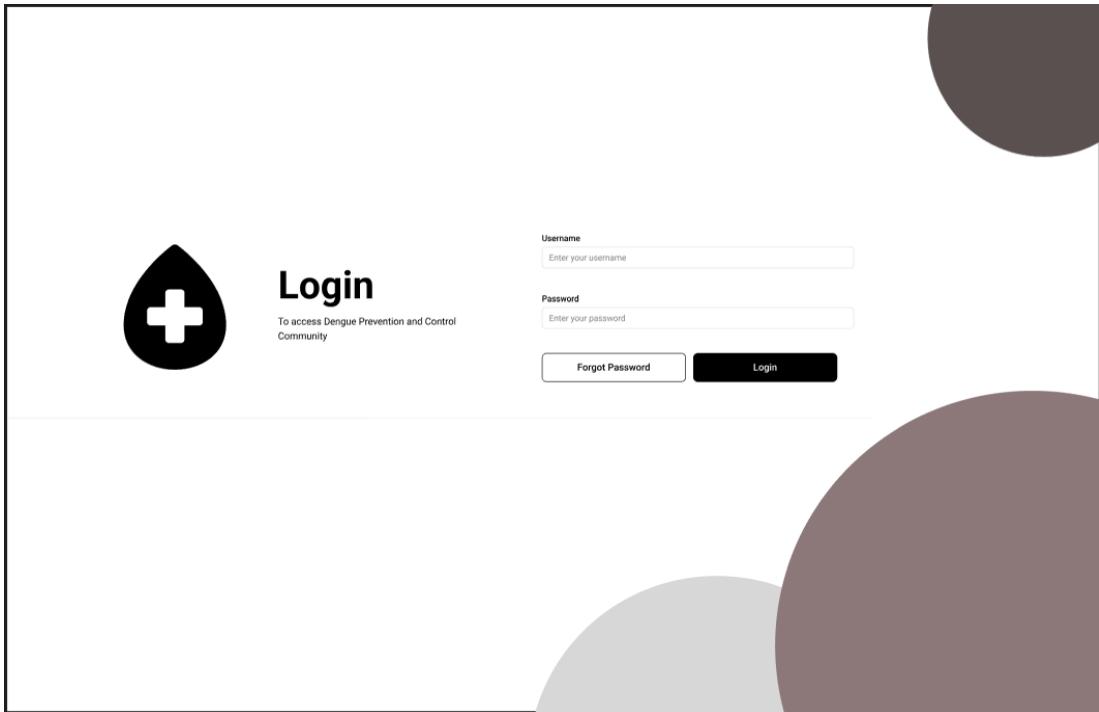


Figure 6.15: Interface for <Customer Log Out/ Success Logout>



The image shows the login interface for the Dengue Prevention and Control Community. It features a logo of a black teardrop with a white cross inside. The word "Login" is prominently displayed in the center. Below it, a sub-header reads "To access Dengue Prevention and Control Community". To the right, there are fields for "Username" and "Password", each with an input placeholder. Below these fields are "Forgot Password" and "Login" buttons.

Figure 6.16: Interface for <Customer Logout/ Back to Login>

In this subsystem, users can log out their account by clicking the logout button in the profile page, then the log out message will appear and the users will be redirected to the login page again.

6.5 Viewing Homepage Subsystem Interface

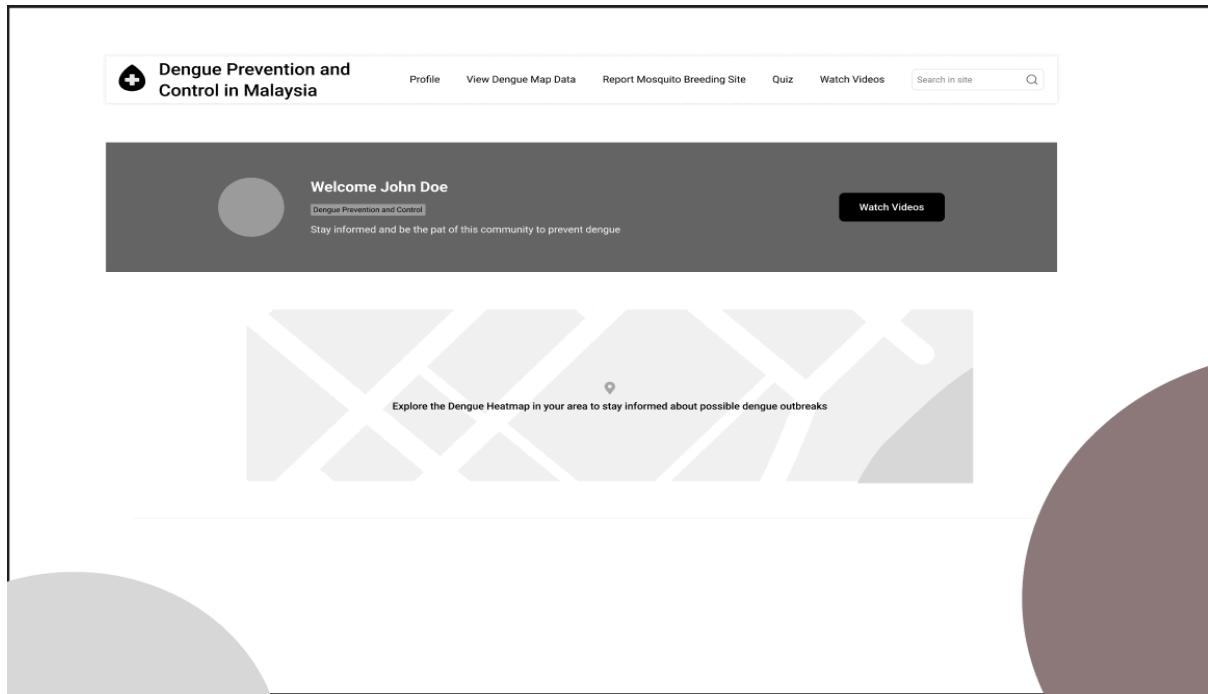


Figure 6.17: Interface for <Viewing Homepage>

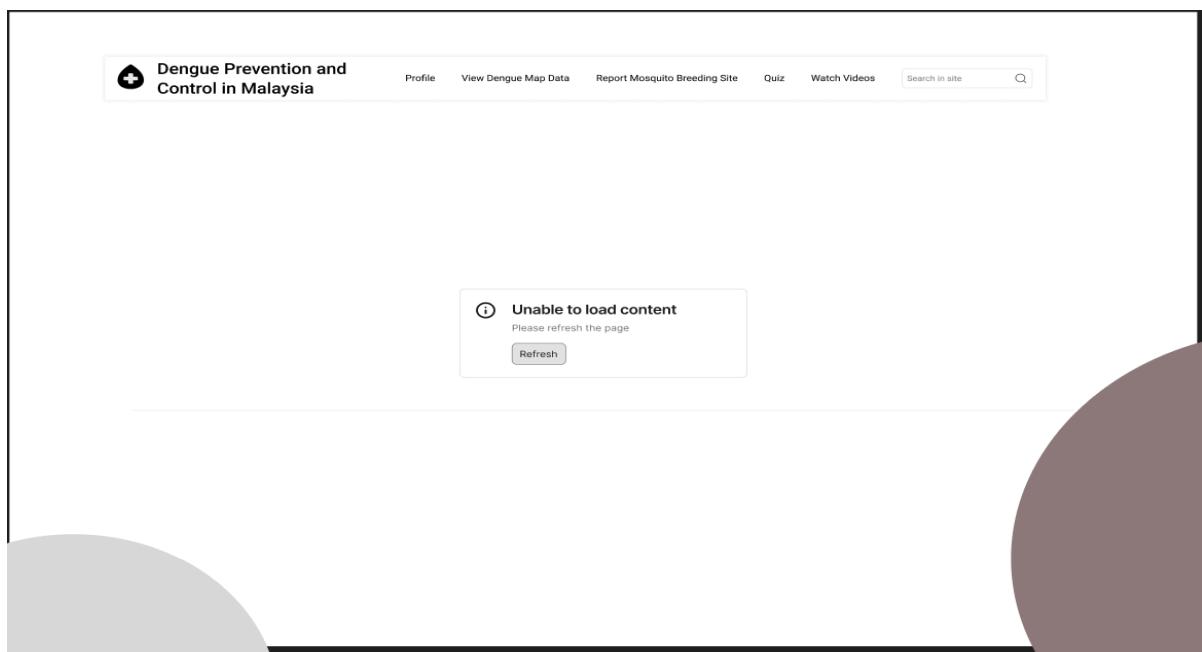


Figure 6.18: Interface for <Viewing Homepage/ Error Popup Message Unable to Load Content>

In this subsystem, after the user's success on login attempt they will be landing on the home page, and if the server cannot load the home page, an error message will appear.

6.6 Viewing Menu Subsystem Interface

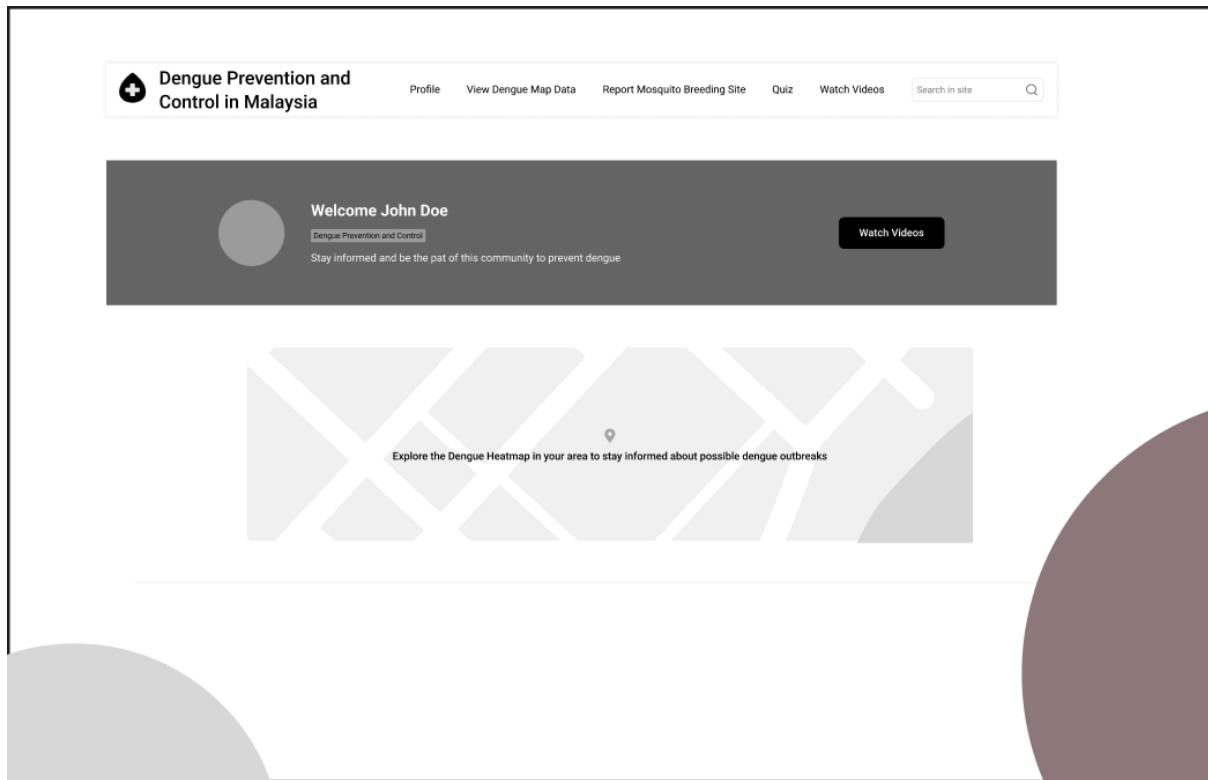


Figure 6.19: Interface for <Viewing Menu>

In this subsystem, users will be landed on the home page and viewing menus after a successful login attempt.

6.7

Forgot Password Subsystem Interface

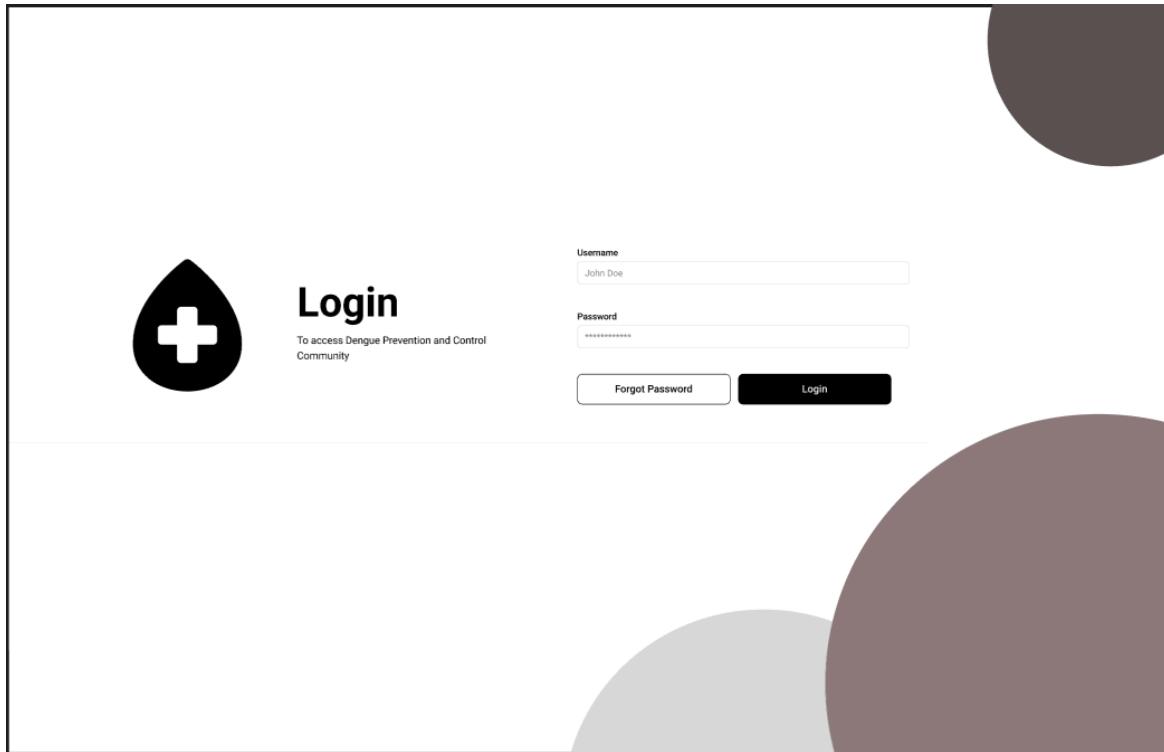


Figure 6.20: Interface for <Forgot Password>

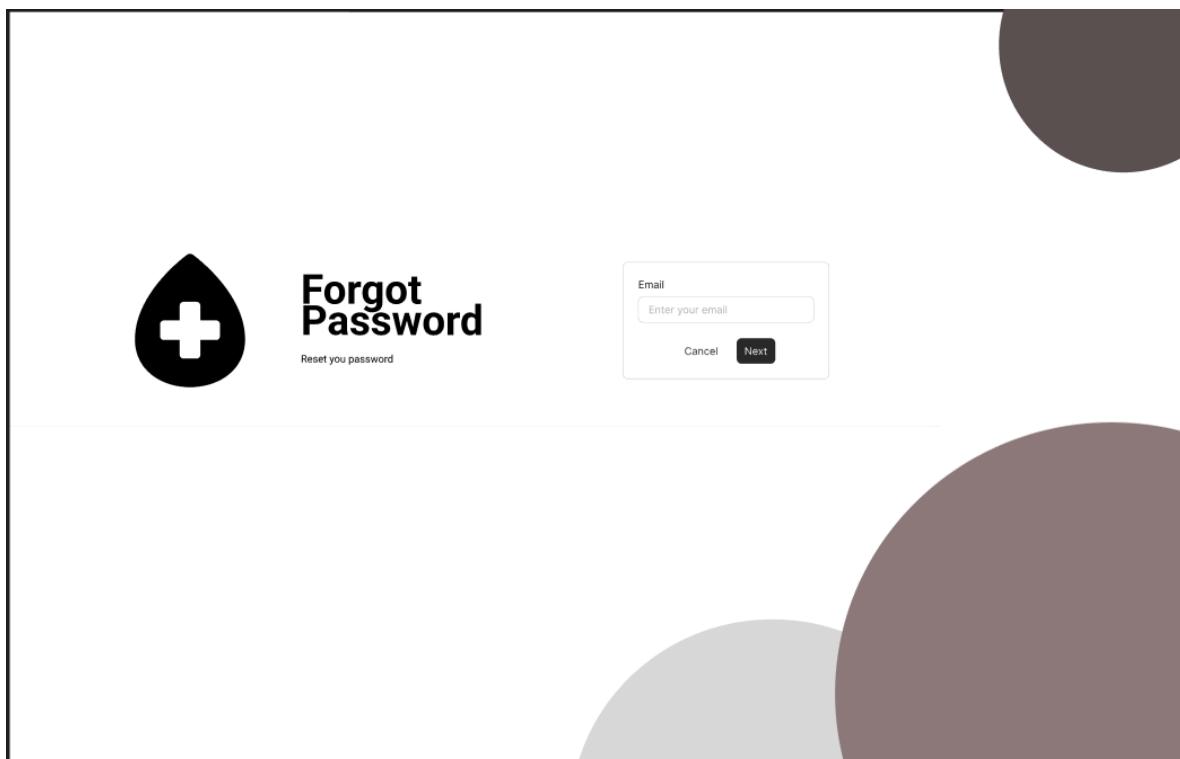


Figure 6.21: Interface for <Forgot Password/ Reset password by email>

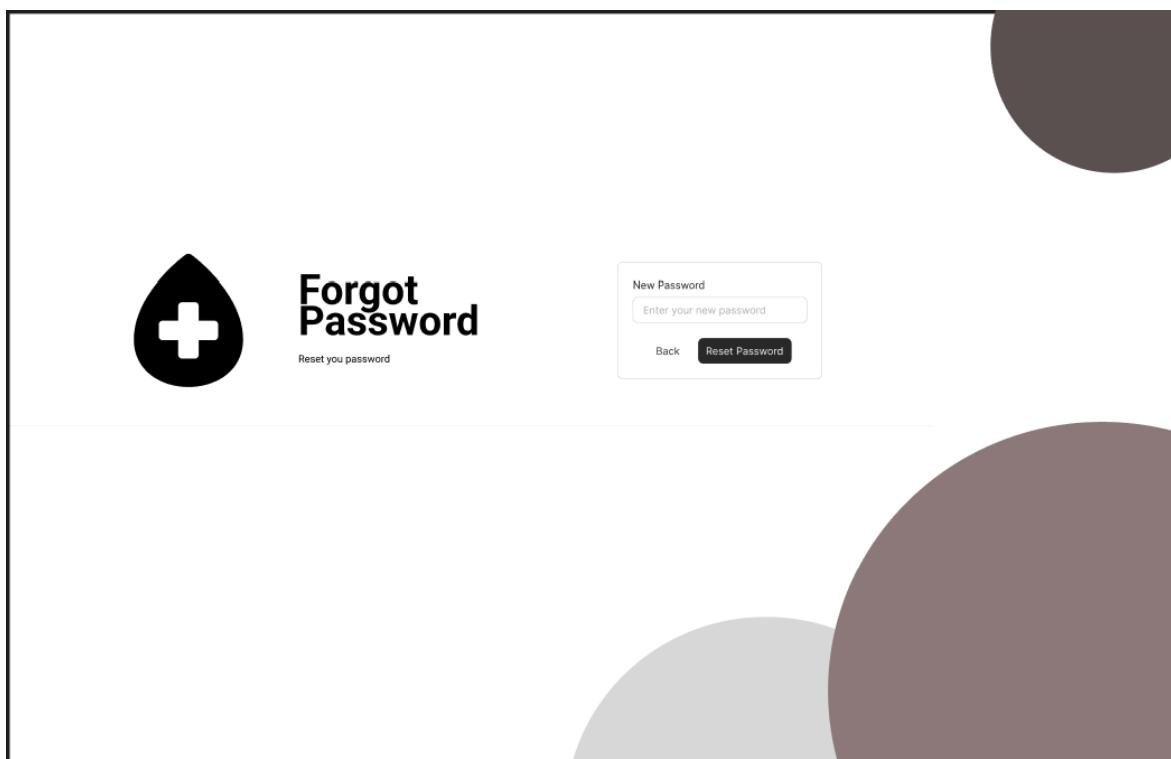


Figure 6.22: Interface for <Forgot Password/ Reset password>

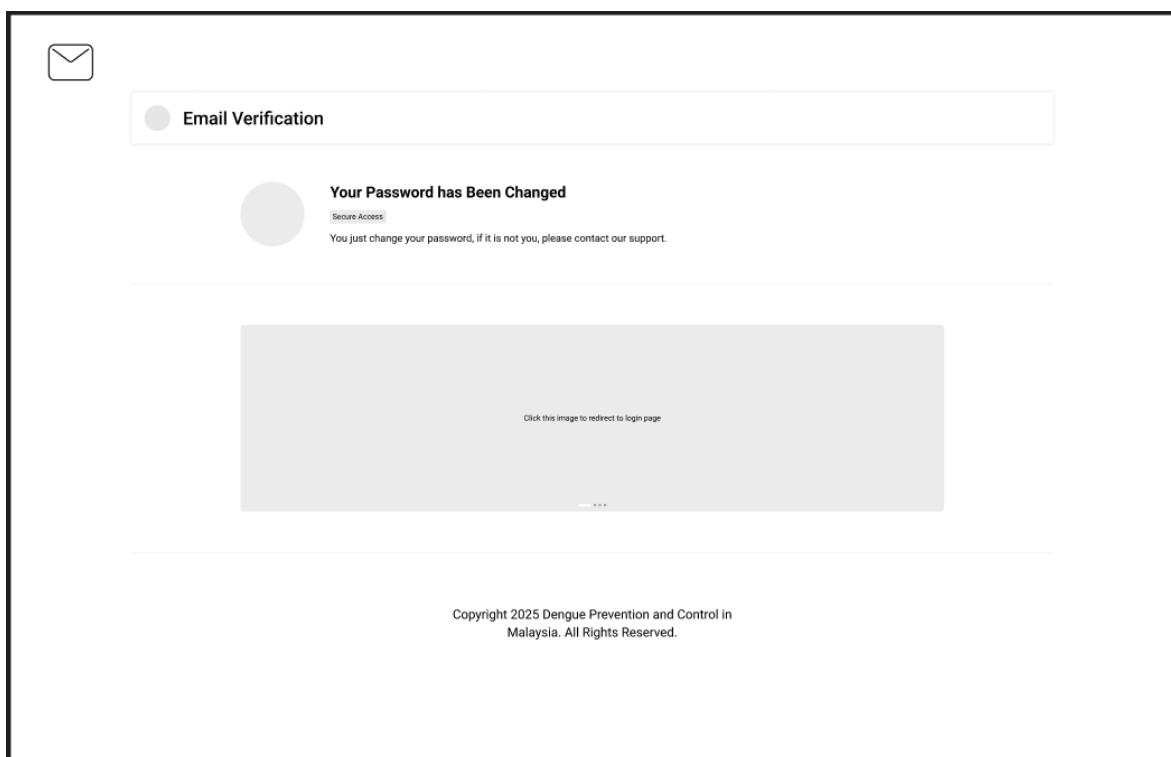


Figure 6.23: Interface for <Forgot Password/ Confirmation Password Changed>

In this subsystem, users can choose the forgot password in the login page, replacing their password and the successful password change email will appear if the users change their password.

6.8 **Changing Password Subsystem**

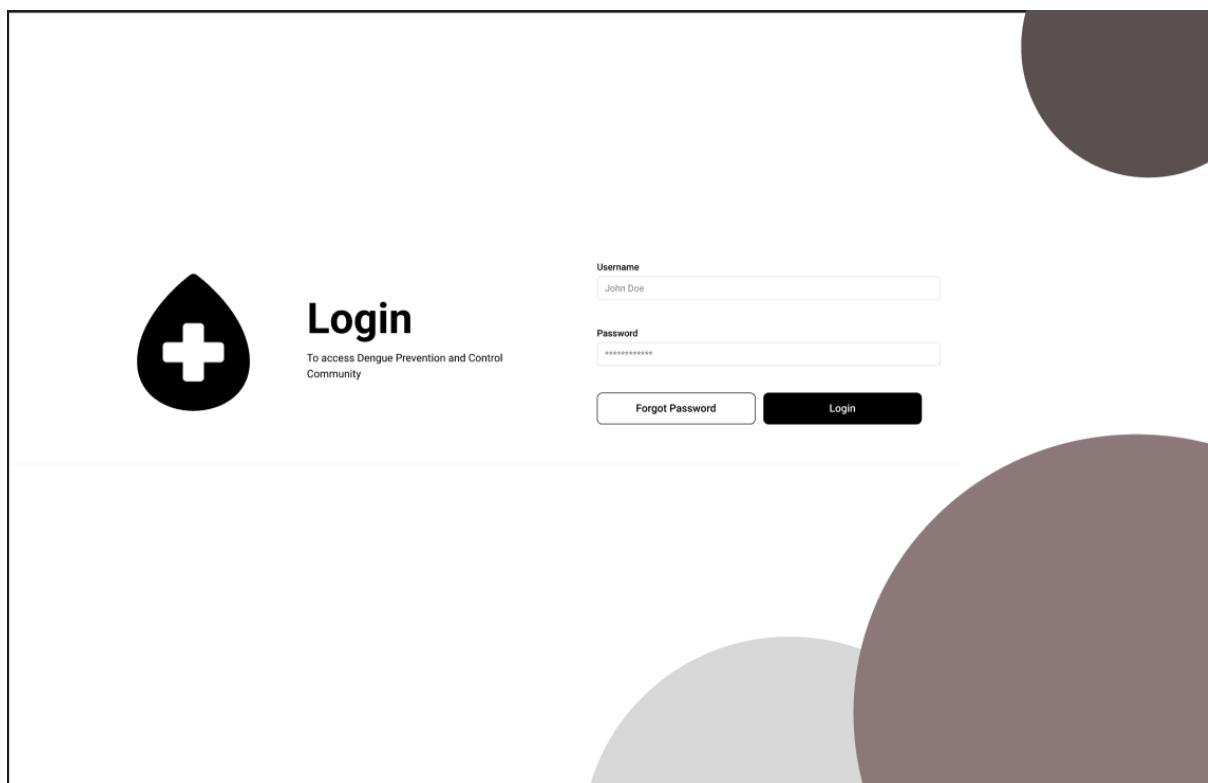


Figure 6.24: Interface for <Changing Password/Login page>

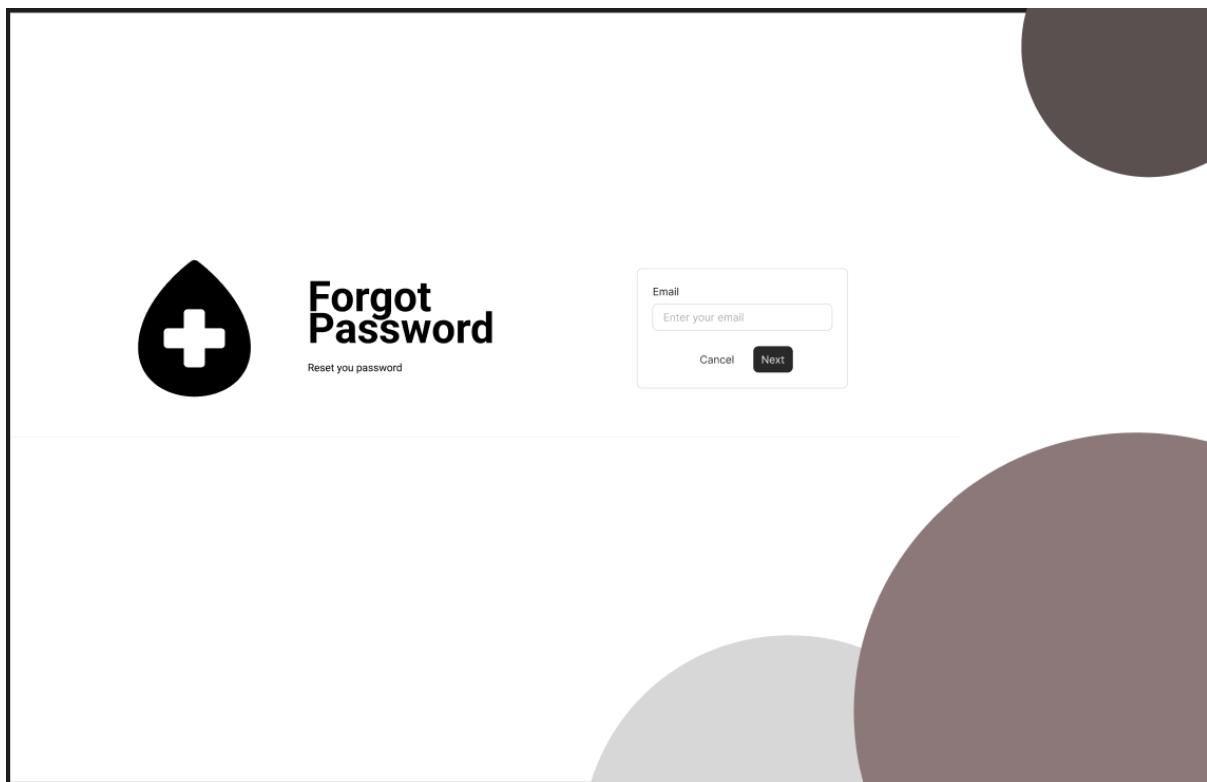


Figure 6.25: Interface for <Changing Password/ Reset password by email>

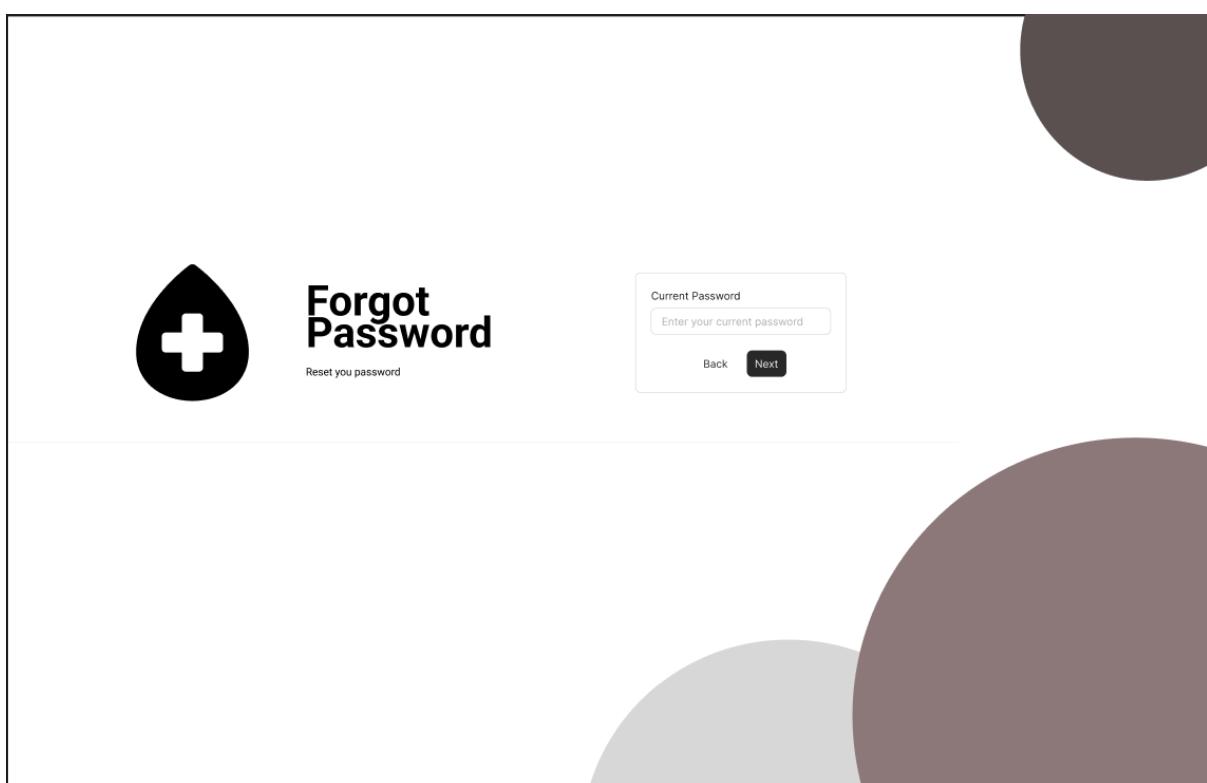


Figure 6.26: Interface for <Changing Password/ Fill in Current Password>

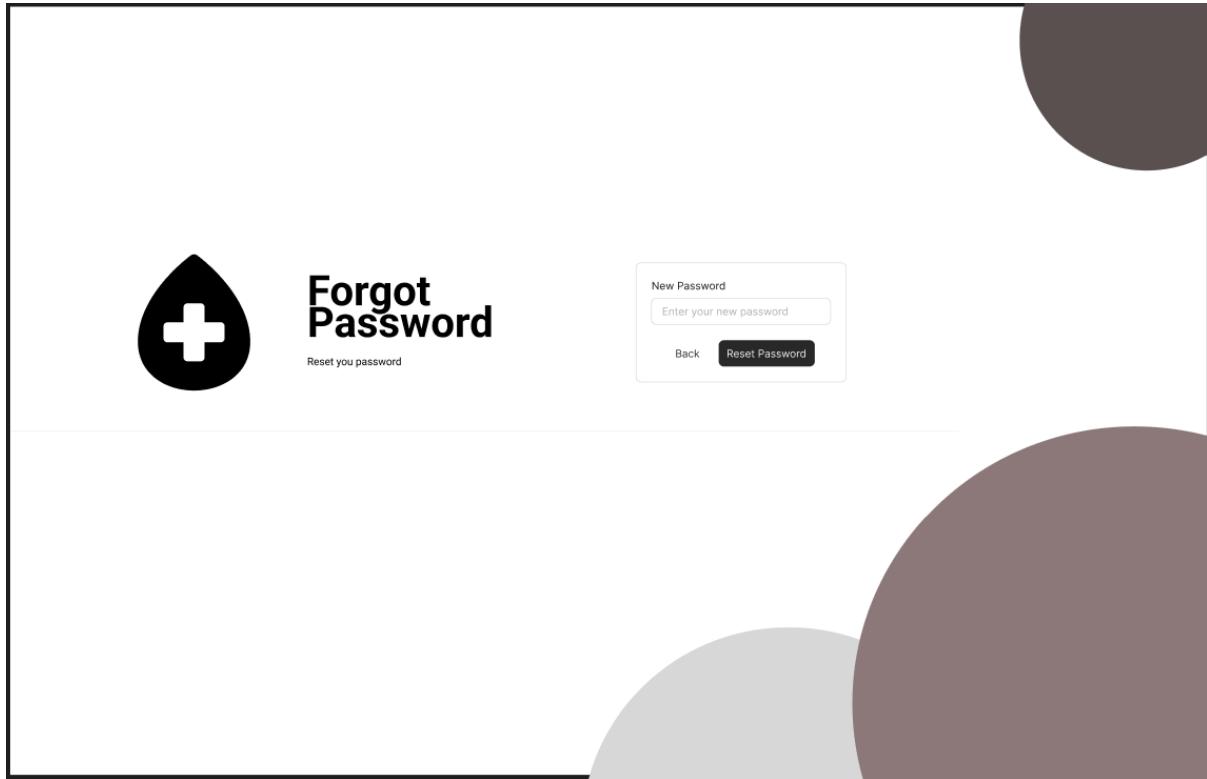


Figure 6.27: Interface for <Changing Password/ Change for New Password>

In this subsystem, users can choose the forgot password to change their password in the login page, replacing their password and the successful password change email will appear if the users change their password.

6.9 Viewing Profile Subsystem Interface

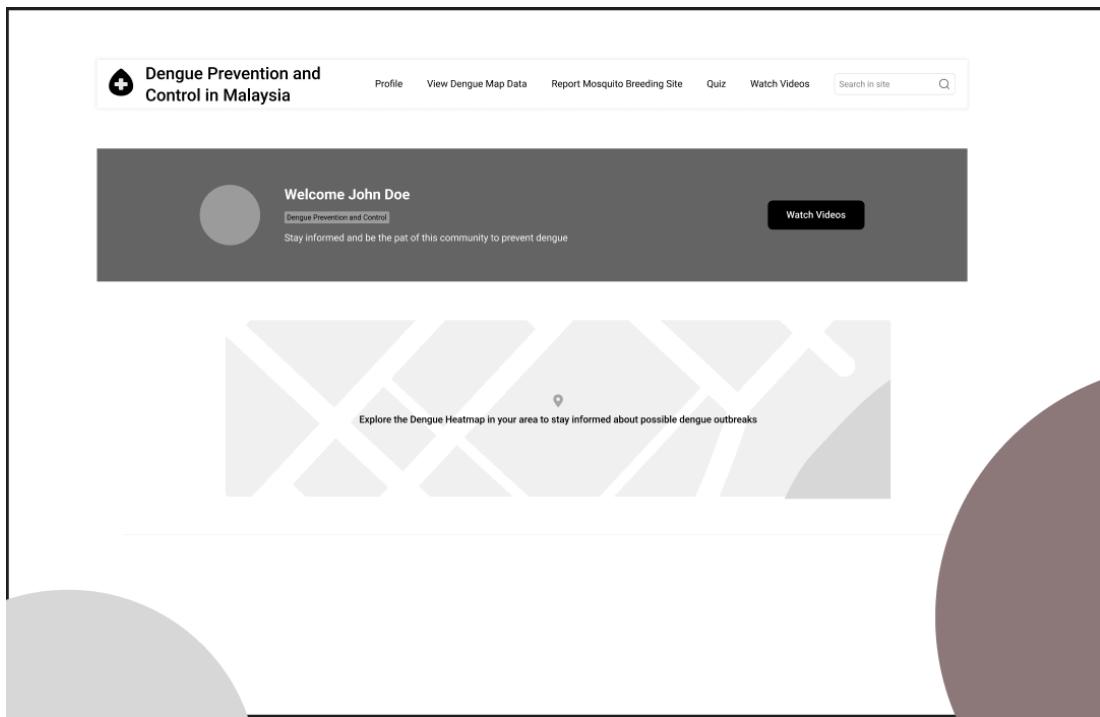


Figure 6.28: Interface for <Viewing Profile/ Homepage>

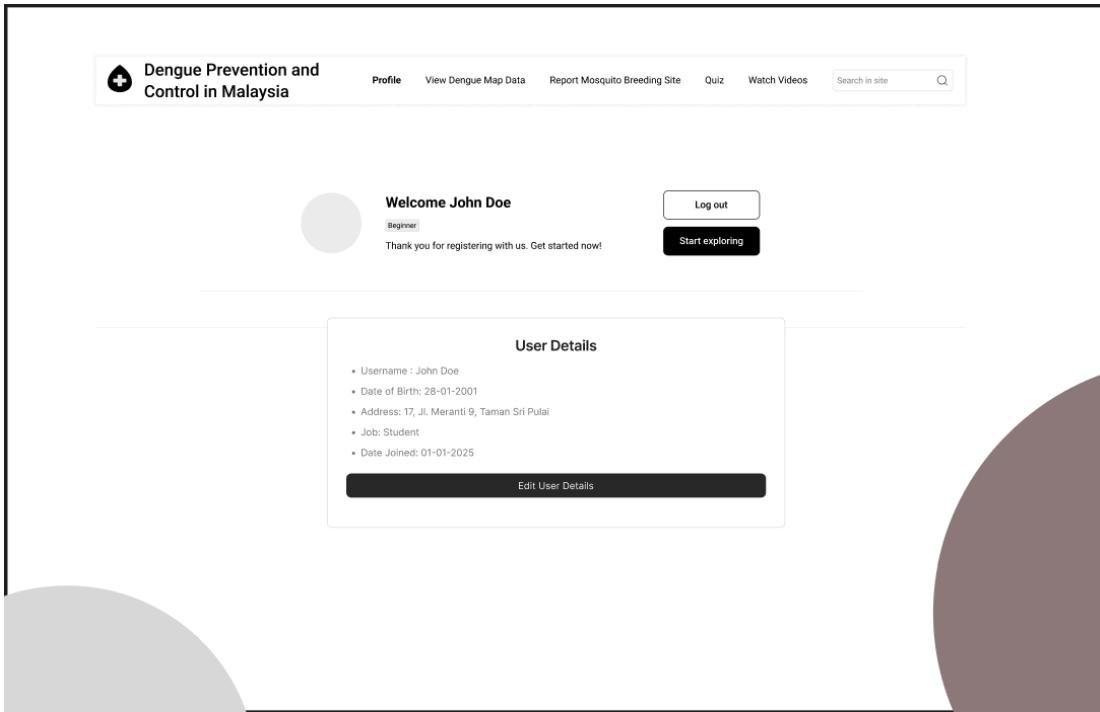


Figure 6.29: Interface for <Viewing Profile/ Profile Information>

In this subsystem, users can choose the profile in the menu to view their profile page.

6.10 Editing Profile Subsystem Interface

The screenshot shows the 'Edit Profile' page. At the top left is the logo 'Dengue Prevention and Control in Malaysia'. To its right are navigation links: Profile, View Dengue Map Data, Report Mosquito Breeding Site, Quiz, Watch Videos, and a search bar labeled 'Search in site' with a magnifying glass icon. The main title 'Edit Profile' is centered above a sub-instruction 'Fill in or change the details below'. Below this are several input fields: 'Username' (John Doe), 'Email' (john.doe@example.com), 'Date of Birth' (28-01-2001), 'Address' (John 17, Jl. Meranti 9, Taman Sri Pulai), 'Mobile Number' (+62 813 1713 4563), 'Country' (USA, Canada, UK, Australia selected), and 'Gender' (Male, Female). At the bottom are 'Reset' and 'Update Profile' buttons.

Figure 6.30: Interface for <Editing Profile>

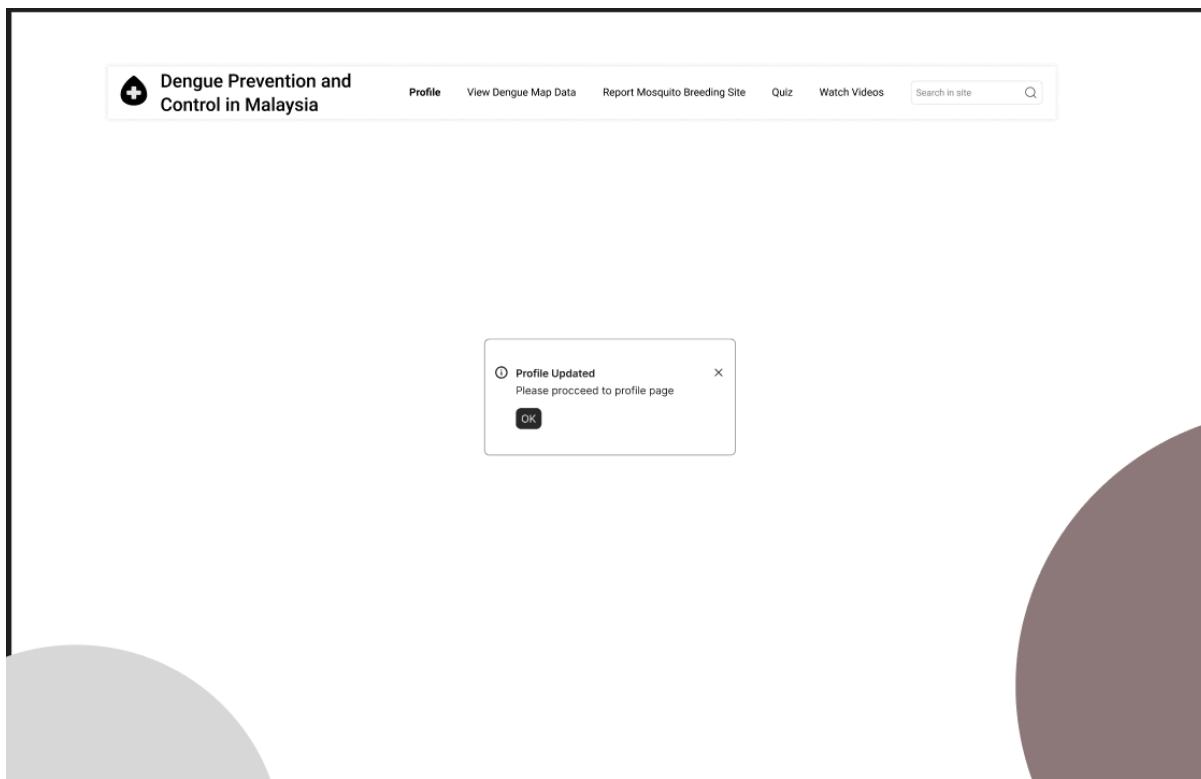


Figure 6.31: Interface for <Editing Profile/ Success Update Profile Popup Message>

In this subsystem, users can choose to edit user details in the profile page, update their personal details and the successful message will appear if the users update their profile successfully.

6.11 Admin Login Subsystem Interface

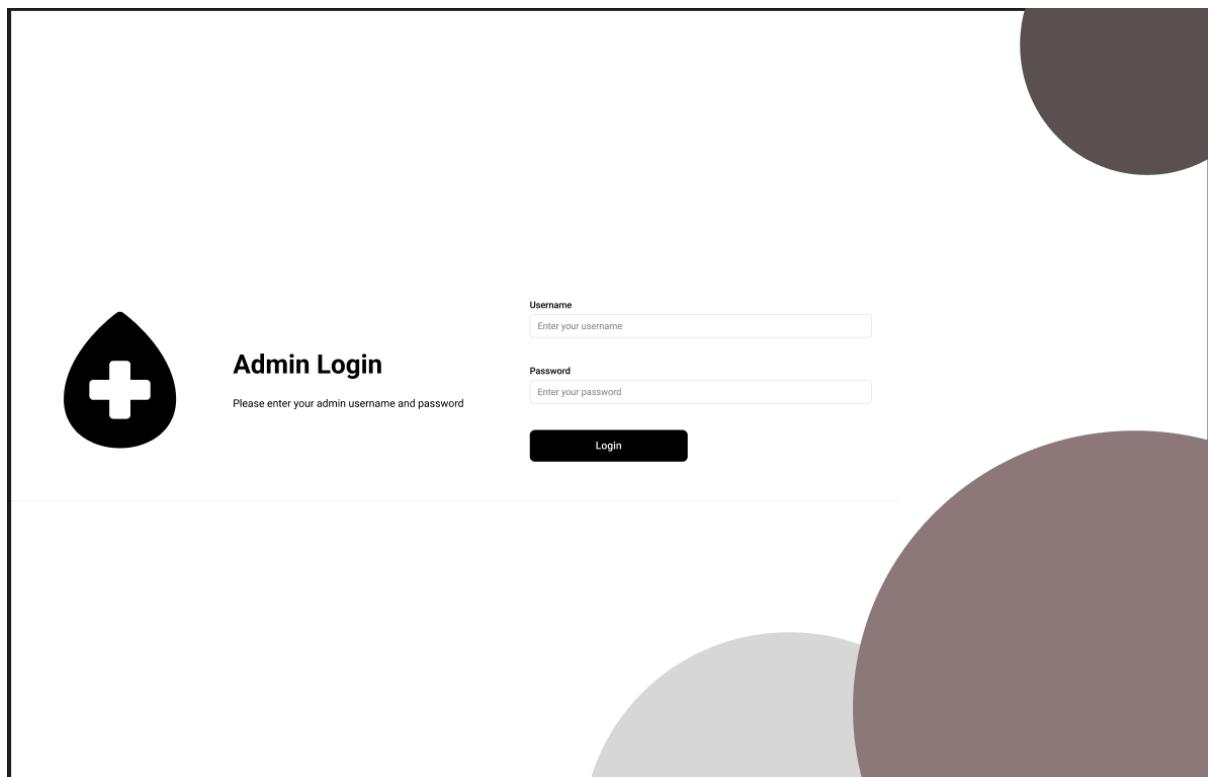


Figure 6.32: Interface for <Admin Login/Login page>

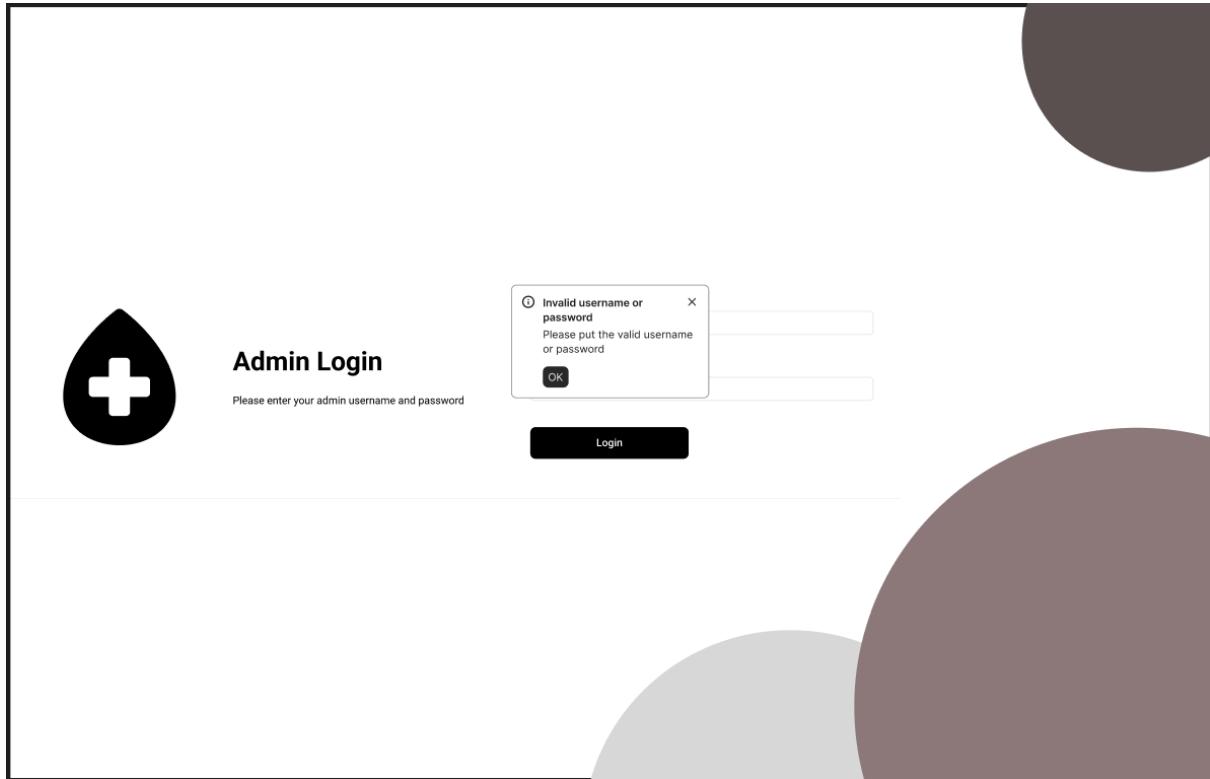


Figure 6.33: Interface for <Admin Login/ Error Popup Message>

In this subsystem, Admin can login using their username and password, and land on the admin dashboard. But if the login was unsuccessful, the failed message will appear.

6.12 Admin Logout Subsystem Interface

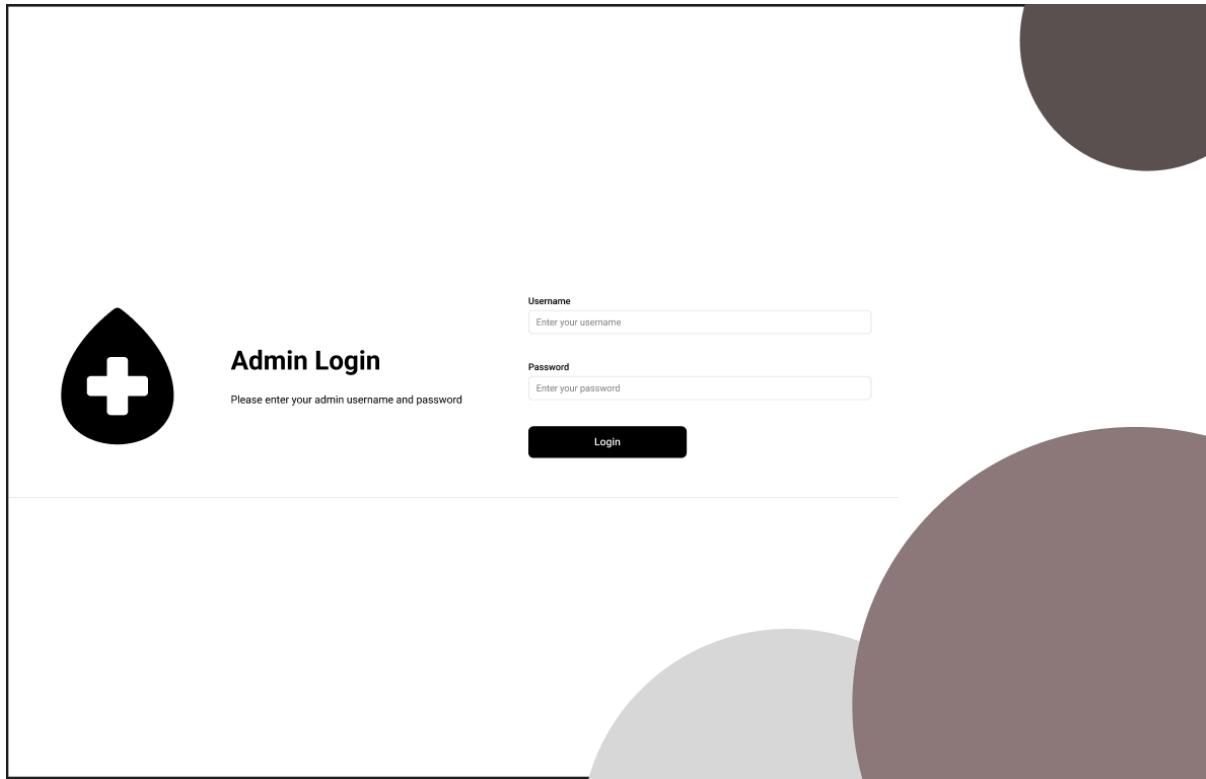


Figure 6.34: Interface for <Admin Logout/ Login for Admin>

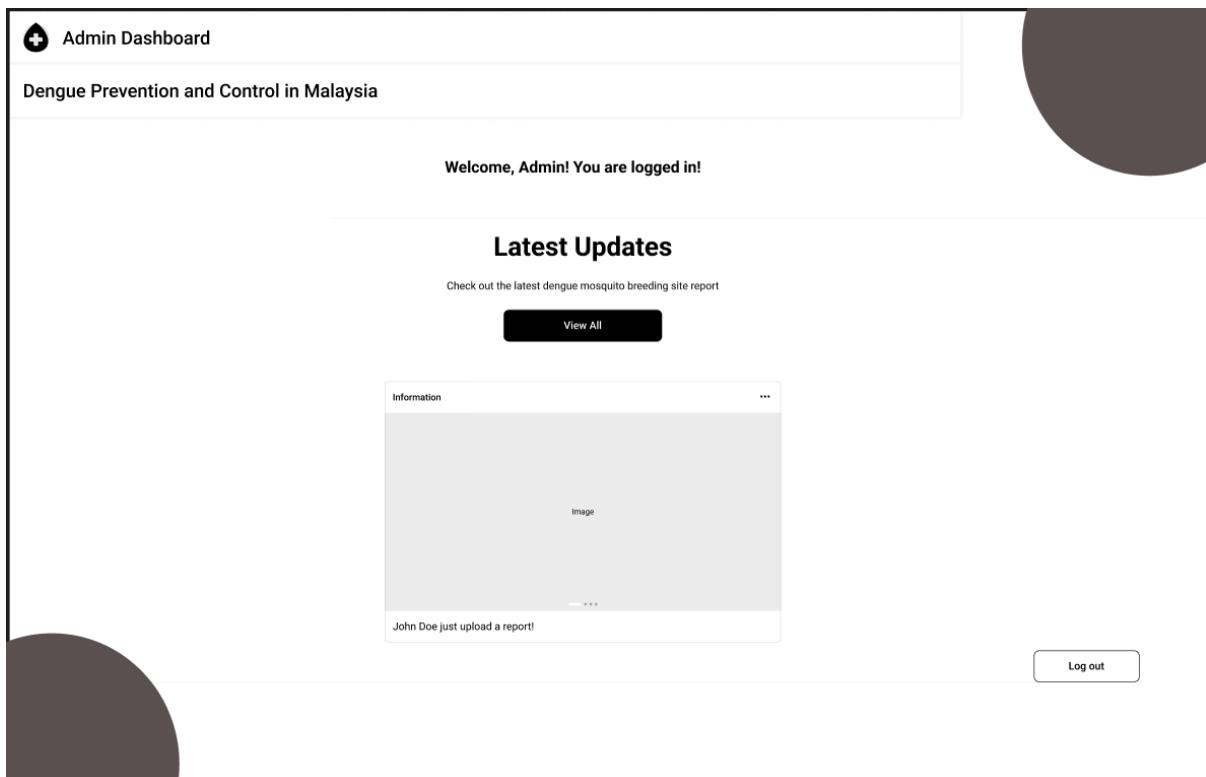


Figure 6.35: Interface for <Admin Logout/ Dashboard for Admin>

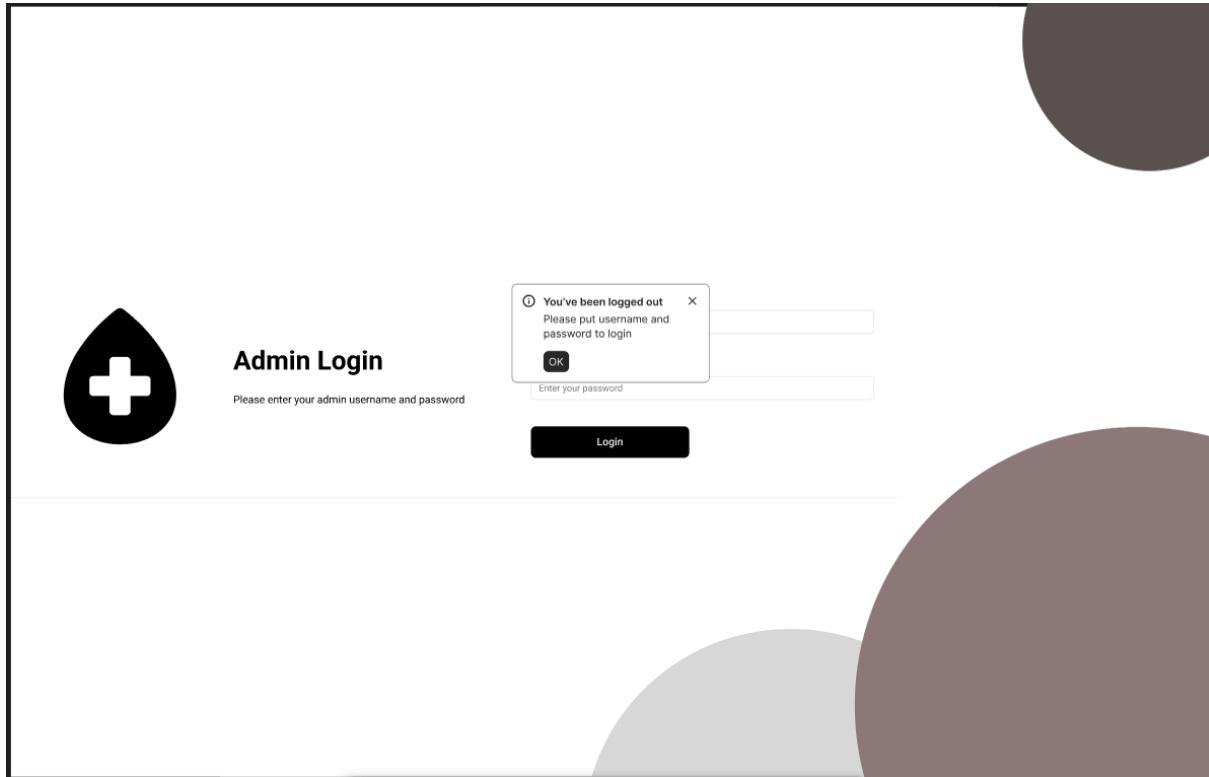


Figure 6.36: Interface for <Admin Logout/ Success Popup Message>

In this subsystem, Admin can choose the log out button in the admin dashboard, and the admin will be redirected to the login page, as the logged out message appears.

6.13

Dengue Map Subsystem Interface

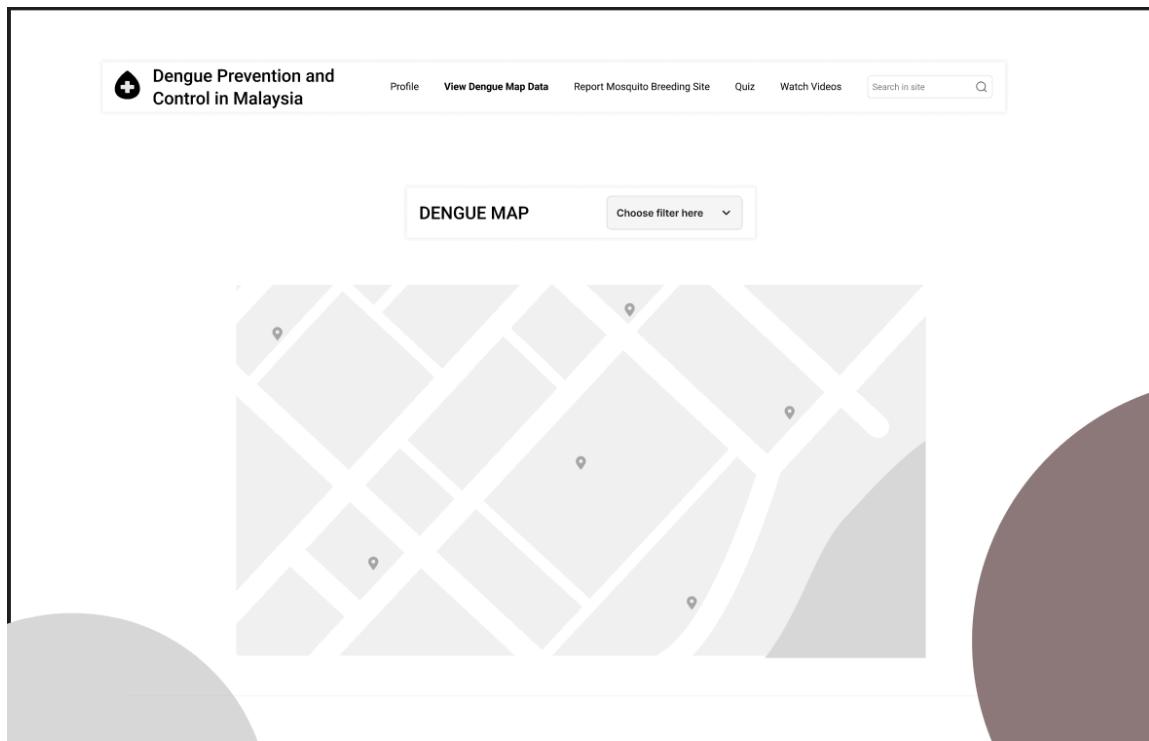


Figure 6.37: Interface for <Dengue Map>

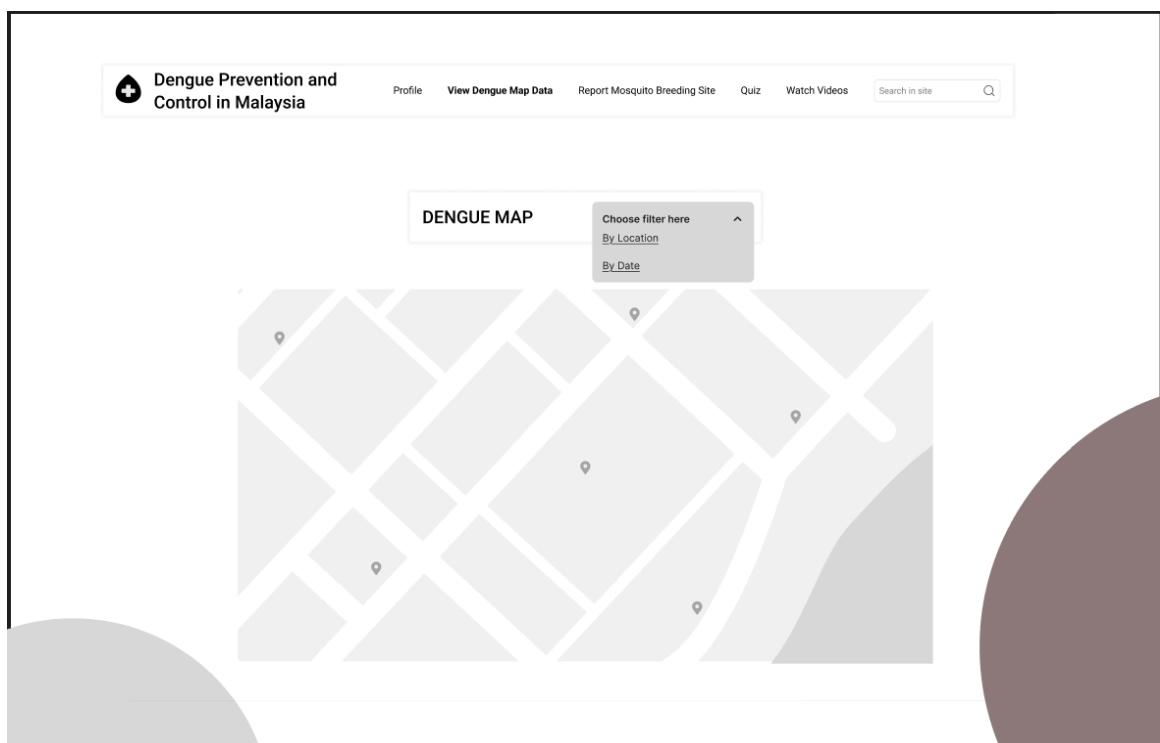


Figure 6.38: Interface for <Dengue Map/ Filter Map>

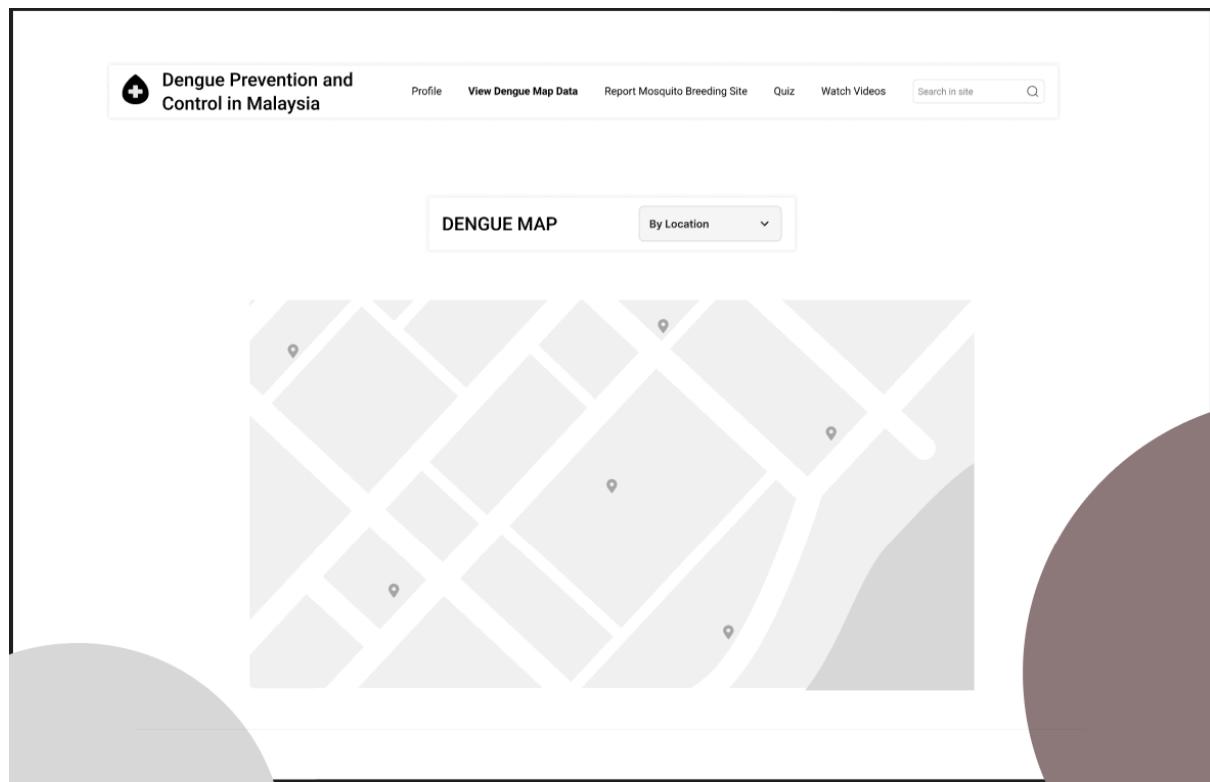


Figure 6.39: Interface for <Dengue Map/ Filter By Location>

In this subsystem, users can see the mosquito breeding site in maps by choosing a dengue map on the menu, users can apply a filter in the map as well.

6.14 Reporting Mosquito Breeding Sites Subsystem Interface

The screenshot shows the 'Report Dengue' form on a website. At the top, there is a navigation bar with links for Profile, View Dengue Map Data, Report Mosquito Breeding Site, Quiz, Watch Videos, and a search bar. The main form area has a title 'Report Dengue' and a subtitle 'Fill in the details below'. It contains three input fields: 'Location' (with placeholder 'Enter location'), 'Date' (with placeholder 'MM/DD/YYYY'), and 'Upload Photo' (with placeholder 'Attach a photo of breeding site'). Below these fields are two buttons: 'Cancel' and 'Submit Report'.

Figure 6.40: Interface for <Reporting Mosquito Breeding Sites/ Make Report>

This screenshot is similar to Figure 6.40, showing the 'Report Dengue' form. However, a modal dialog box is overlaid on the right side of the page, containing a success message: 'Report submitted successfully' and 'You will be directed to the homepage'. There is an 'OK' button at the bottom of the dialog.

Figure 6.41: Interface for <Reporting Mosquito Breeding Sites/ Success Popup Message >

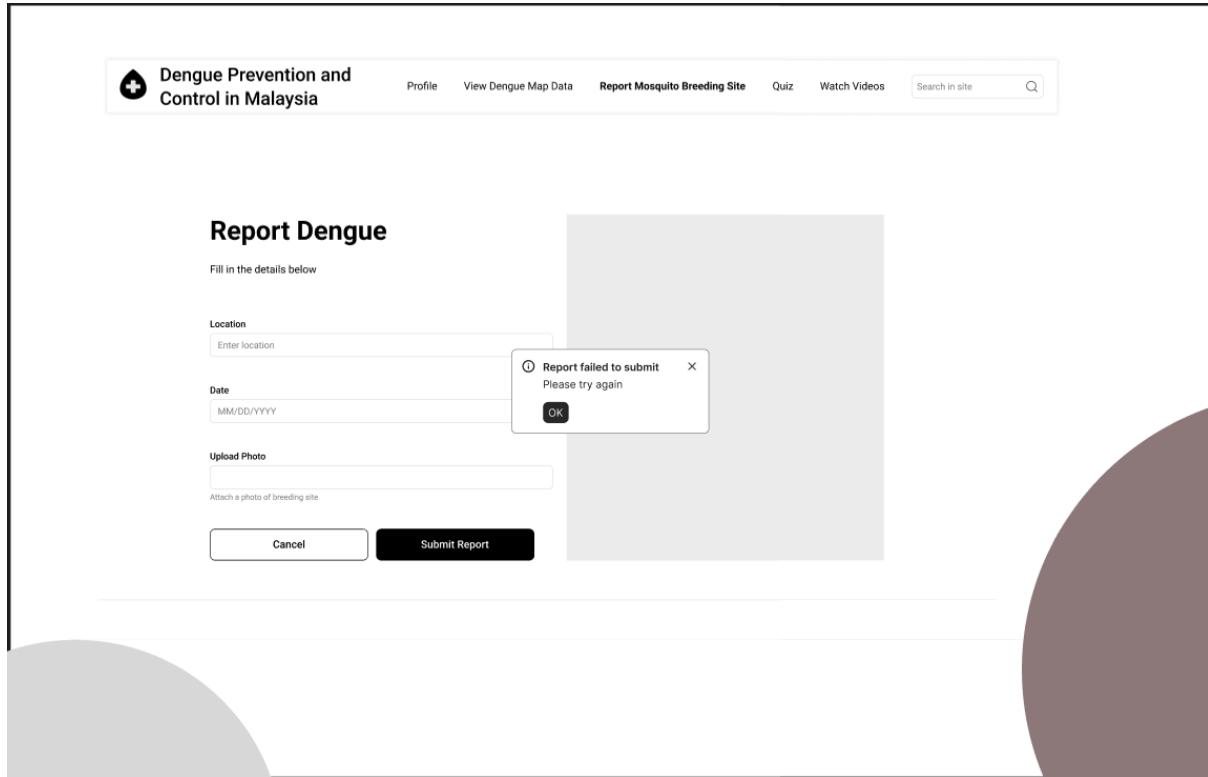


Figure 6.42: Interface for <Reporting Mosquito Breeding Sites/ Failed Popup Message >

In this subsystem, users can choose to report mosquito breeding sites on the menu, and the user will be redirected to the report form to fill their report and attach the photo. After a successful submission, the successful message will appear, as a failed message will appear when the submission is failed.

6.15 Viewing Educational Videos Subsystem Interface

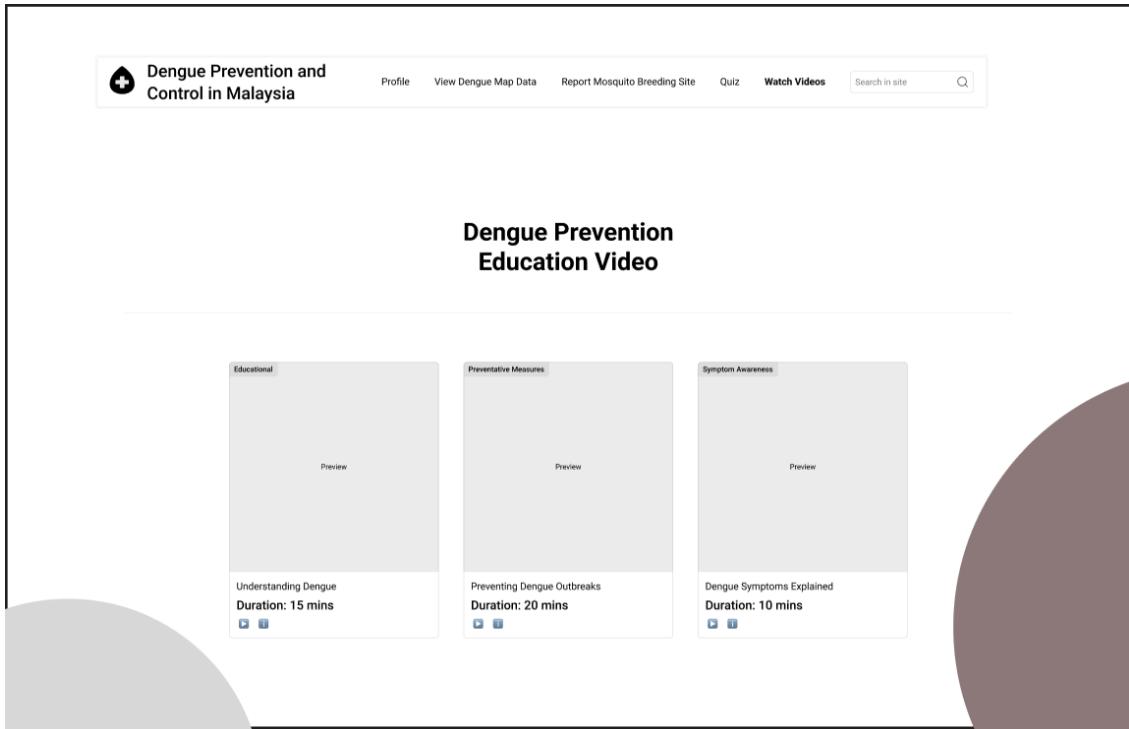


Figure 6.43: Interface for <Viewing Educational Videos/ Videos>

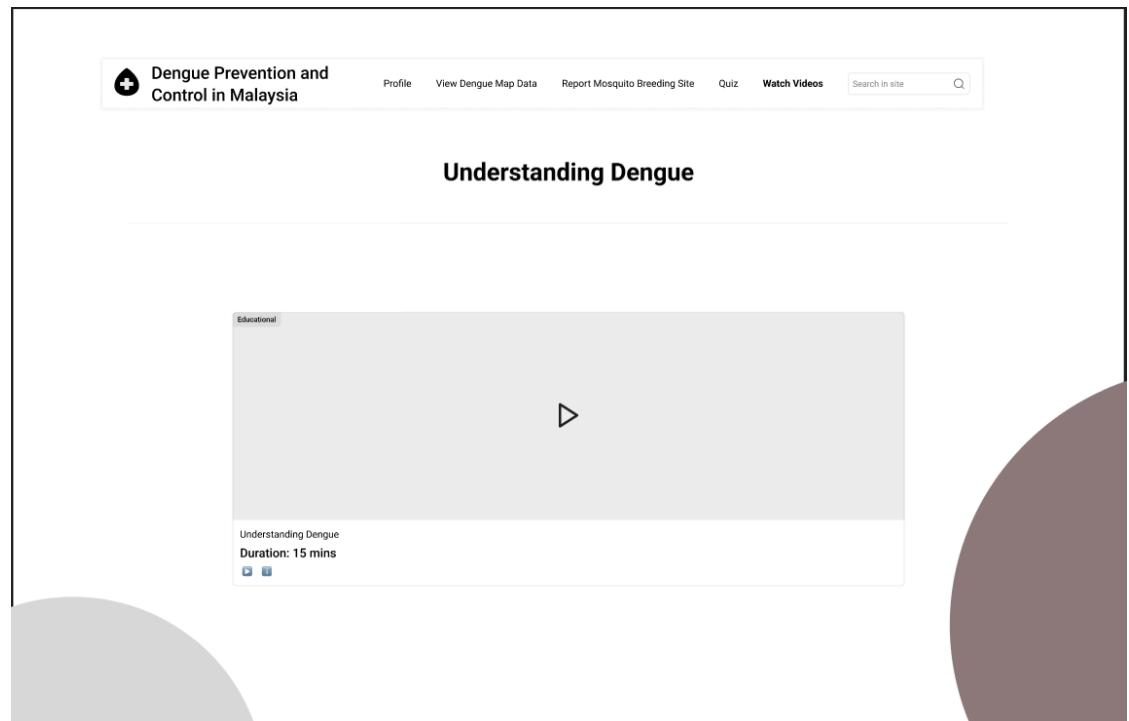


Figure 6.44: Interface for <Viewing Educational Videos/ Select Video>

In this subsystem, users can choose the watch video on the menu to see the dengue videos.

6.16 Taking Quizzes Subsystem Interface

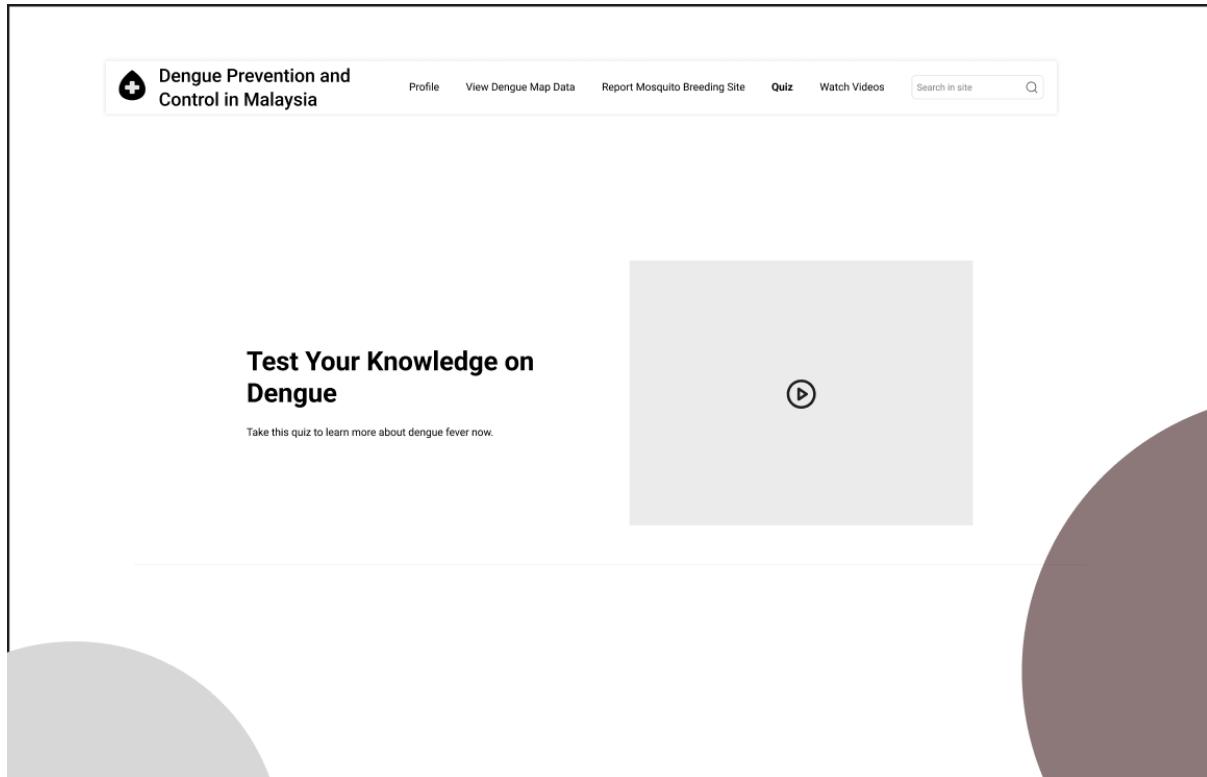


Figure 6.45: Interface for <Taking Quizzes/ Quiz page>

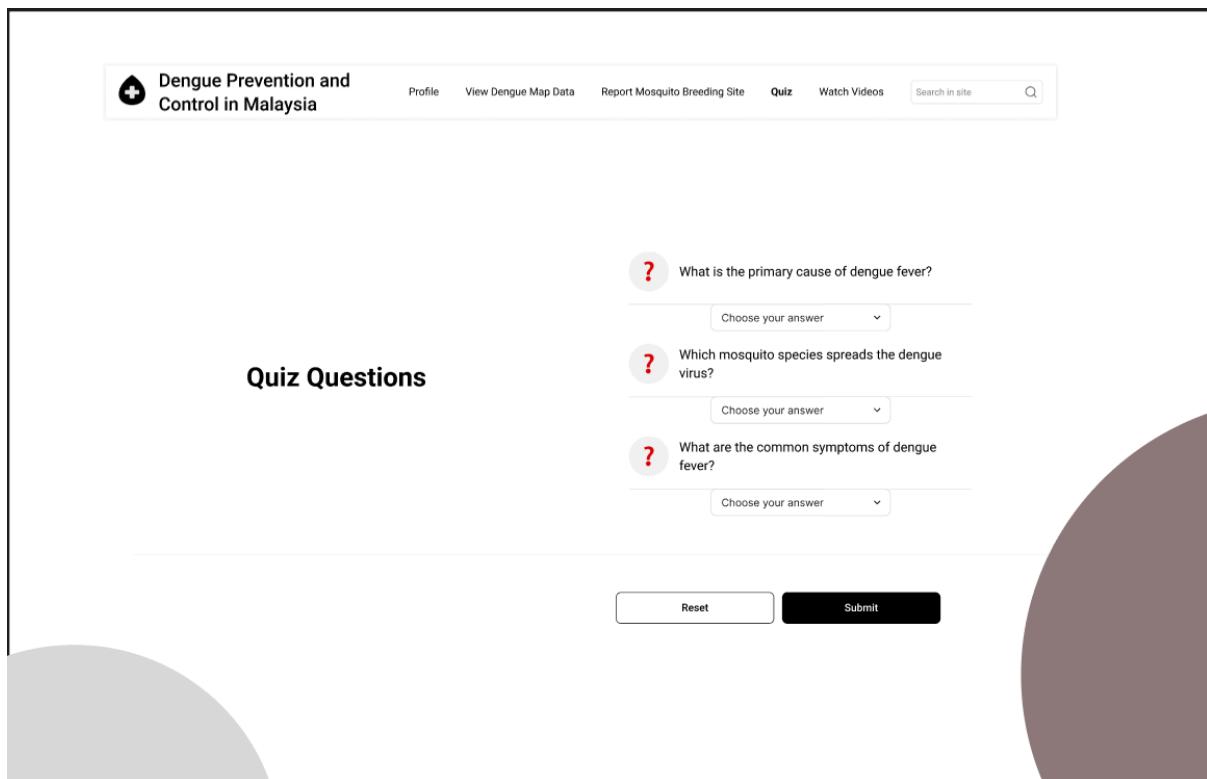


Figure 6.46: Interface for <Taking Quizzes/ Quiz Questions>

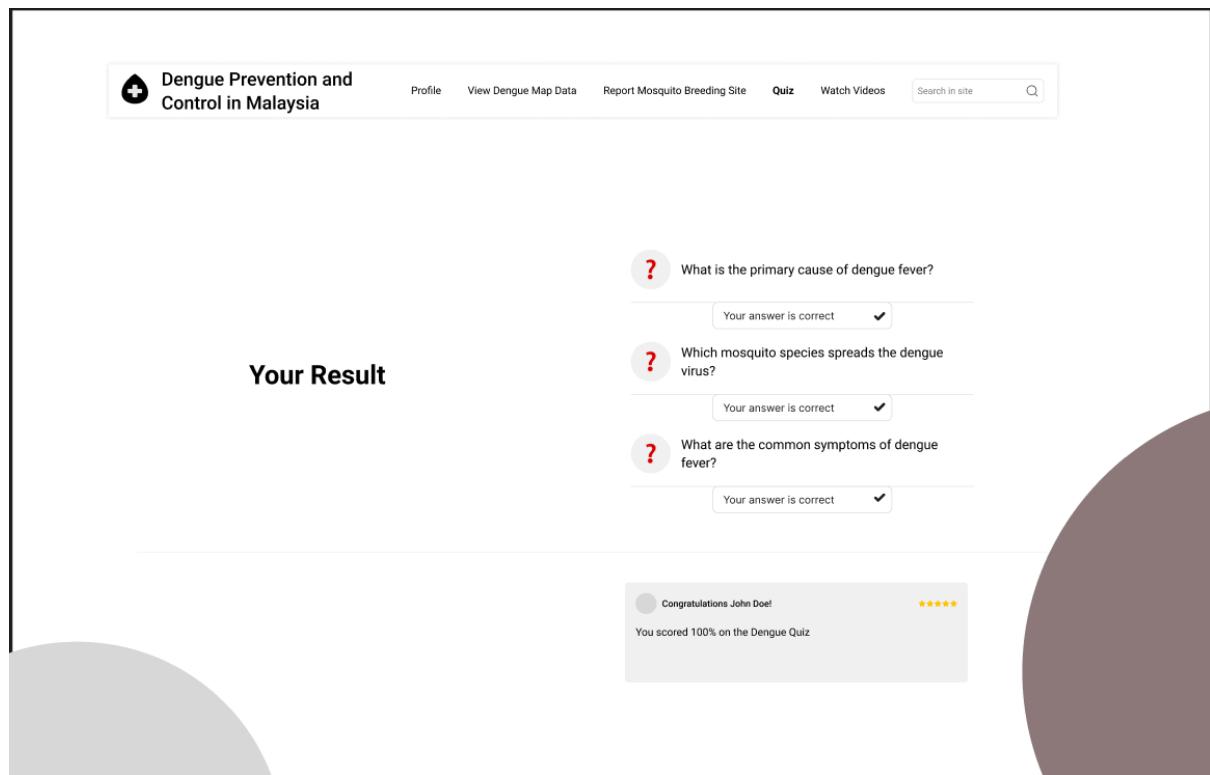


Figure 6.47: Interface for <Taking Quizzes/ Result Quiz>

In this subsystem, users can answer dengue quizzes by selecting a quiz on the menu.

6.17 Admin Managing Reports Subsystem Interface

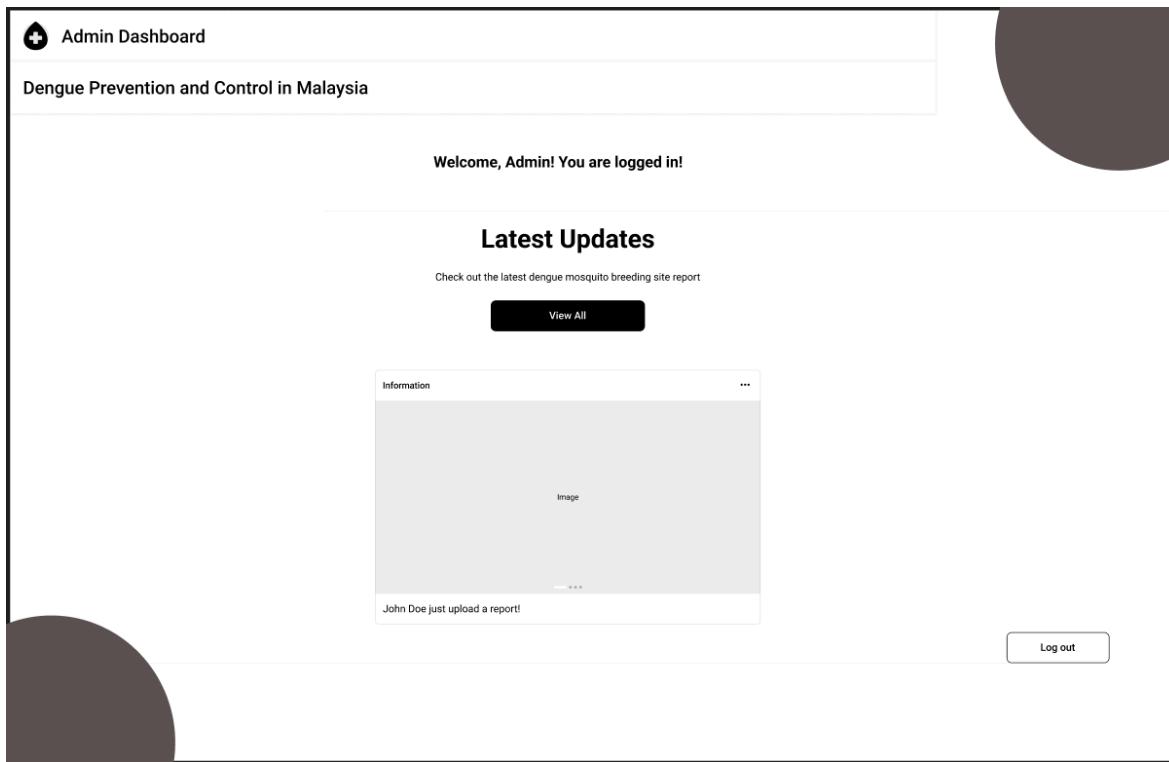


Figure 6.48: Interface for <Admin Managing Reports/ Dashboard Admin>

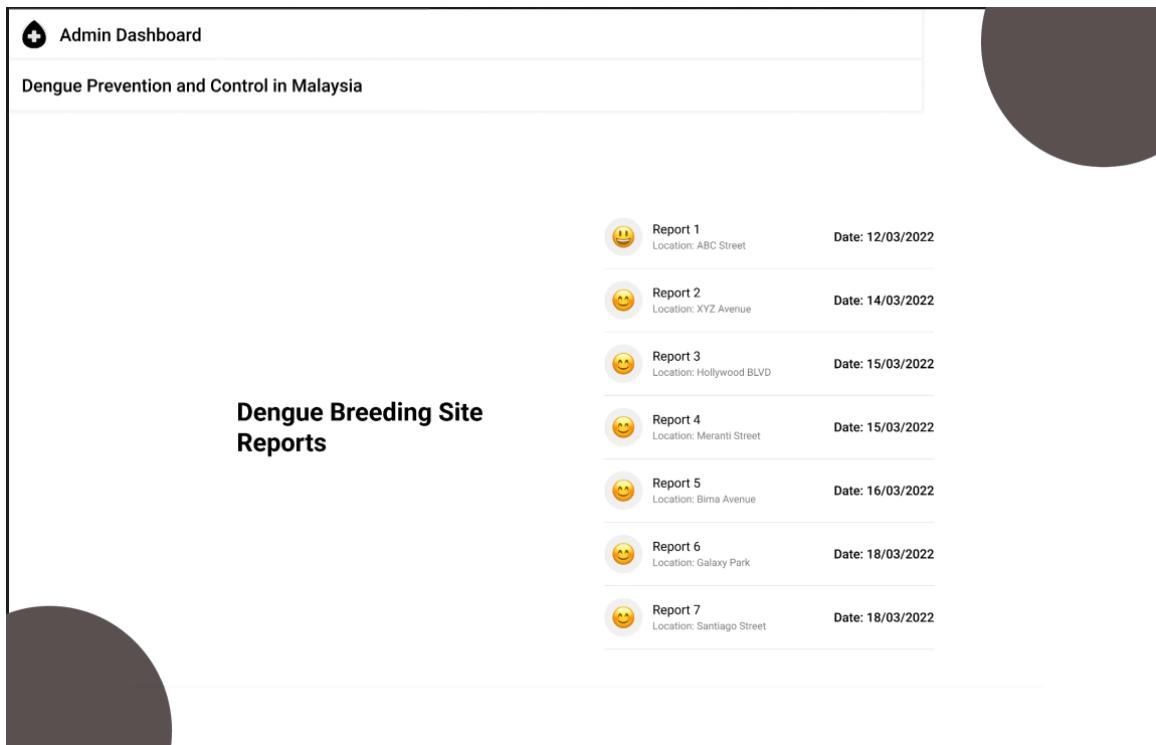


Figure 6.49: Interface for <Admin Managing Reports/List Reports Dengue Breeding>

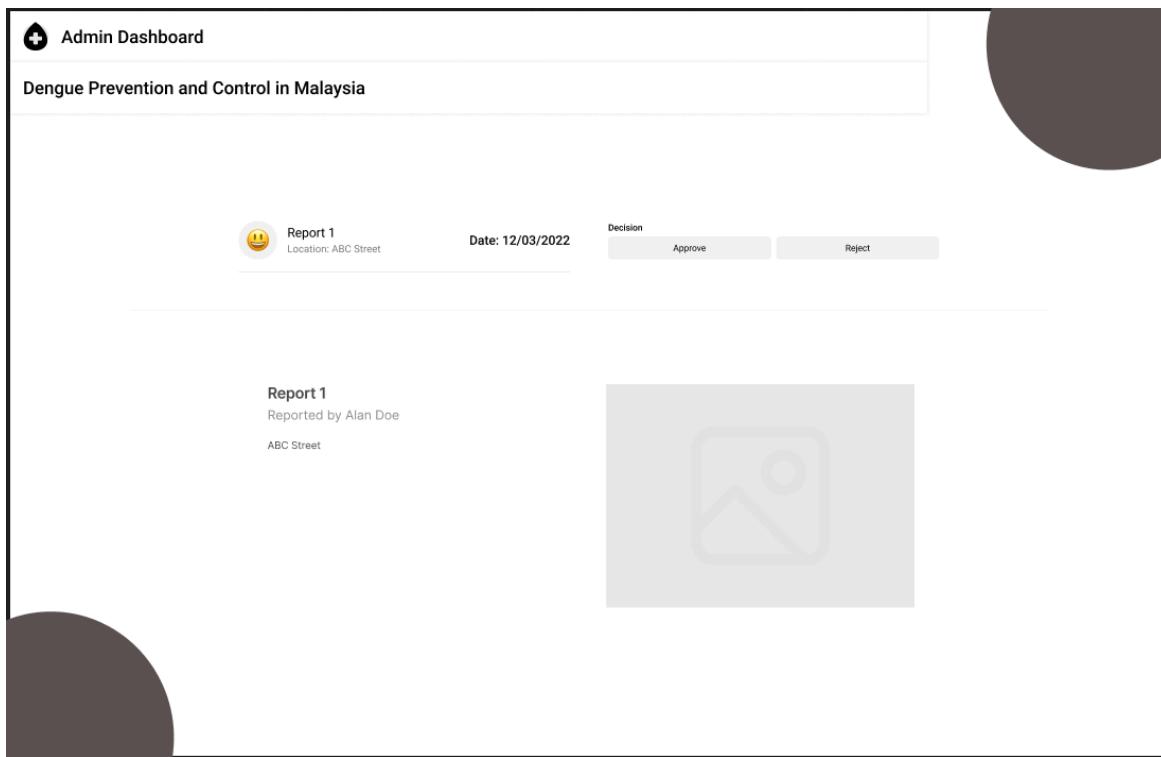


Figure 6.50: Interface for <Admin Managing Reports/View Report Dengue Breeding for Approval>

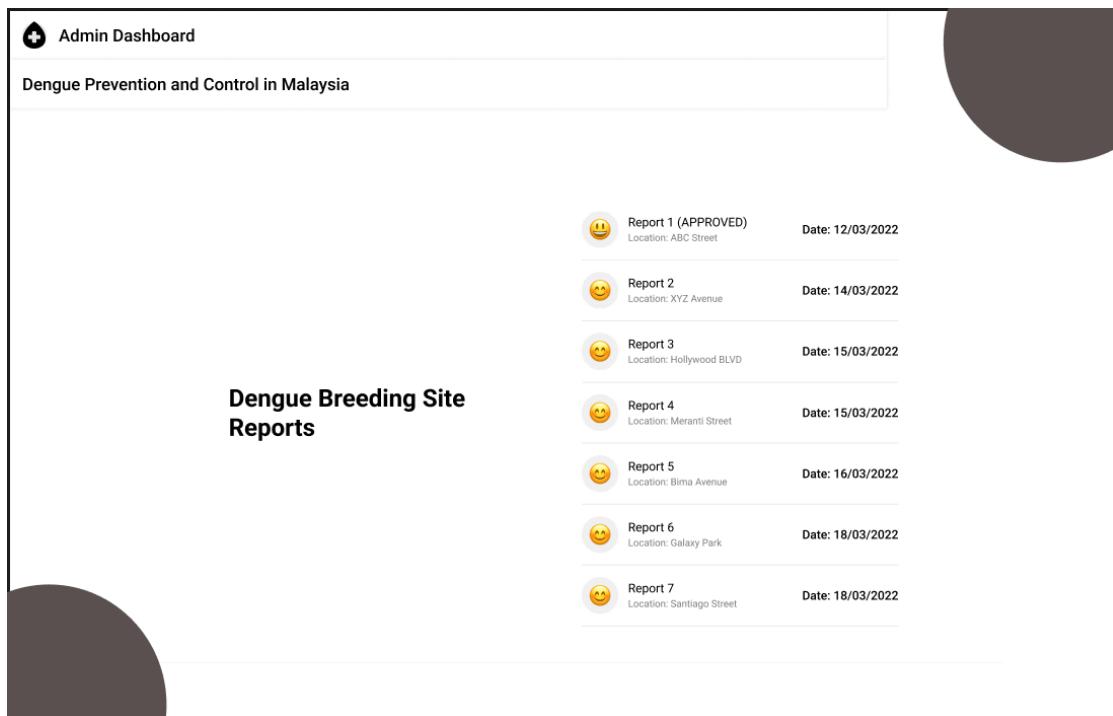


Figure 6.51: Interface for <Admin Managing Reports/Approve Report Dengue Breeding>

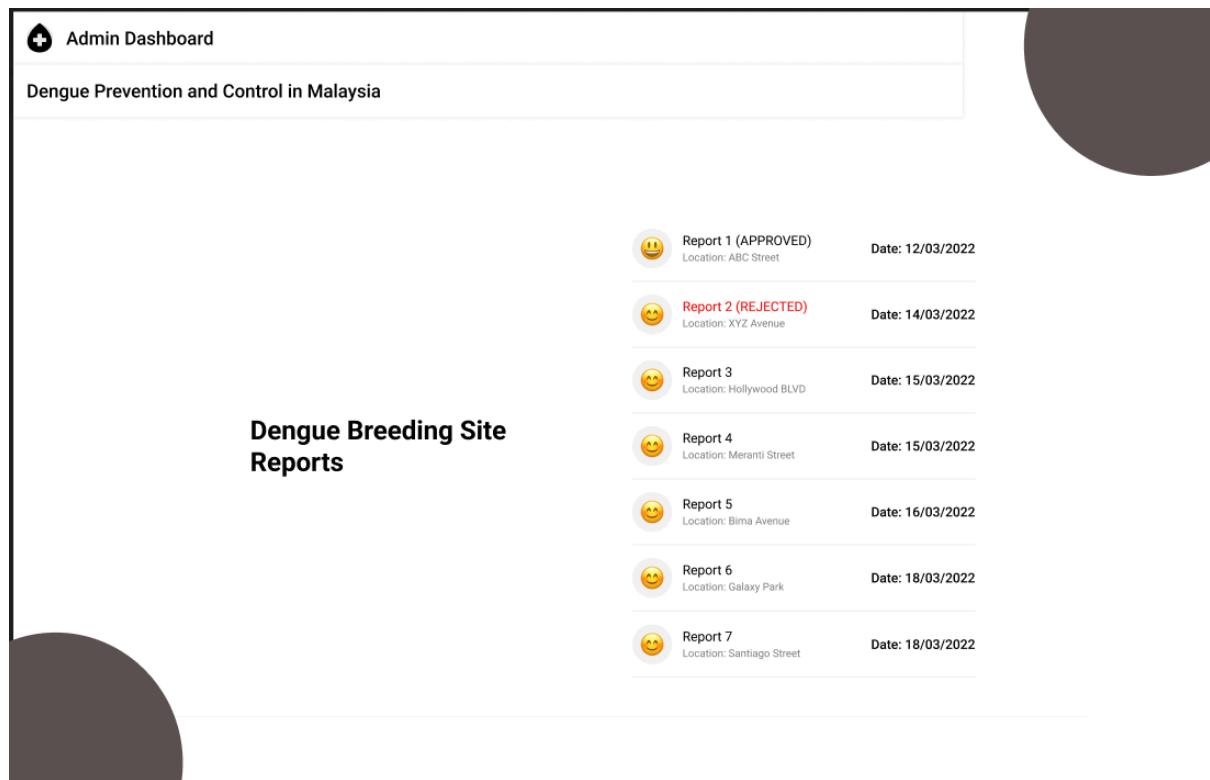


Figure 6.52: Interface for <Admin Managing Reports/Reject Report Dengue Breeding>

In this subsystem, the admin can manage reports, see the details about the report, approve or reject the report, and the report will be marked as the admin's decision.

6.18

Admin Managing Quizzes Subsystem Interface

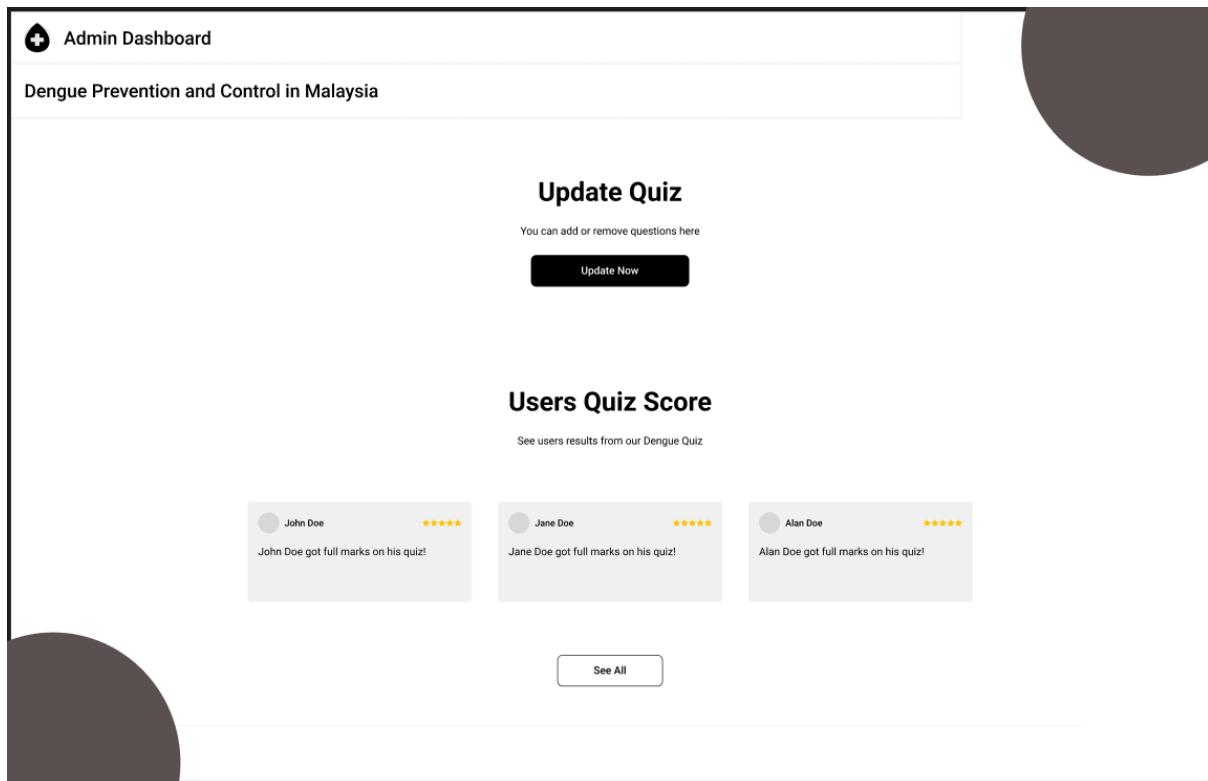


Figure 6.53: Interface for <Admin Managing Quizzes>

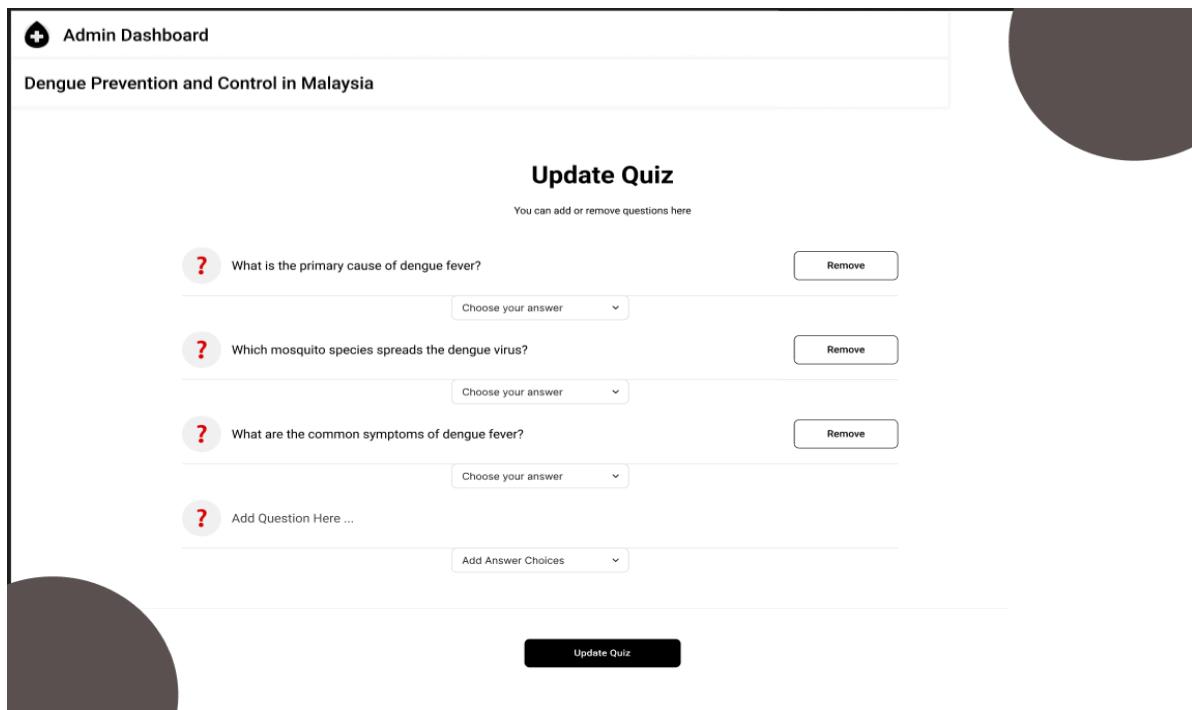


Figure 6.54: Interface for <Admin Managing Quizzes/ Update Quiz Questions>

The screenshot shows the Admin Dashboard for the Dengue Prevention and Control in Malaysia system. The main title is "Admin Dashboard" and the sub-section is "Dengue Prevention and Control in Malaysia". The central heading is "Users Quiz Score" with the sub-instruction "See users results from our Dengue Quiz". Below this, there are eight user profiles, each consisting of a small profile picture, the user's name, and five yellow stars indicating a perfect score. Each profile also includes a success message stating that the user got full marks on their quiz.

| User | Score | Message |
|-------------------|-------|---|
| John Doe | ★★★★★ | John Doe got full marks on his quiz! |
| Jane Doe | ★★★★★ | Jane Doe got full marks on his quiz! |
| Alan Doe | ★★★★★ | Alan Doe got full marks on his quiz! |
| Brandon Doe | ★★★★★ | Brandon Doe got full marks on his quiz! |
| Rafif Doe | ★★★★★ | Rafif Doe got full marks on his quiz! |
| Fairuz Doe | ★★★★★ | Fairuz Doe got full marks on his quiz! |
| Zai | ★★★★★ | Zai got full marks on his quiz! |
| Franklin B. | ★★★★★ | Franklin B. got full marks on his quiz! |
| Cristiano Ronaldo | ★★★★★ | Cristiano Ronaldo got full marks on his quiz! |

Figure 6.55: Interface for <Admin Managing Quizzes/ List Users Quiz Score>

In this subsystem, admin can manage dengue quizzes, adding or removing questions and review users' results.

7. Traceability

Traceability refers to the ability to follow the flow of requirements and changes across all project phases, ensuring full coverage, effective change management, and systematic validation. By connecting requirements to their corresponding design, code, and tests, traceability helps reduce risks, manage changes, and ensure that the end product meets stakeholder expectations.

Table 7.1: Traceability

| Sprint # | Subsystem/Package | User Story ID |
|----------|-----------------------------|---------------|
| Sprint 1 | Package User Management | US01 |
| | | US02 |
| | | US03 |
| Sprint 2 | Package Dengue Information | US04 |
| | | US05 |
| Sprint 3 | Package Learning Engagement | US06 |
| | | US07 |
| | | US08 |
| Sprint 4 | Package Admin | US09 |
| | | US10 |
| Sprint 5 | Package Admin | US11 |
| | | US12 |
| | | US13 |

8. Test Cases

8.1 TC001: Test Package User Management Subsystem: Guest Registration (US001)

This test contains the following test cases:

- (a) **TC001_01**: Test the normal registration process from sequence diagram 1 (SD001).
- (b) **TC001_02**: Test an alternate flow where the user makes a choice for Google registration in SD001.
- (c) **TC001_03**: Test an exception flow where required fields are not filled in SD001.

8.1.1 TC001_01: Test Normal Scenario of Sequence Diagram 1 (SD001)

TC001_01_01 : Test Normal Flow of Guest Registration

| Test Case ID (TC001_01_01) | Test Steps | Input Data | Expected Results | Actual Results | PASS/FAIL | Tested By | Tester Comments |
|--------------------------------------|-----------------------------------|--|---------------------------------|----------------|-----------|-----------|-----------------------|
| 1 | Navigate to the registration page | http://demo.guru99.com | The landing page should be open | As Expected | PASS | ALIF | expected page show up |
| 2 | enter valid data | Name: Alif Azman Email: alif2505@gmail.com Password: SecureP@ss123 | Credential can be entered | As Expected | PASS | ALIF | successful |
| 3 | submit form | | submit button can be click | As Expected | PASS | ALIF | submit successful |
| 4 | verify email | | complete verify email | As Expected | PASS | ALIF | no flows |
| 5 | login | | Customer is logged in | As Expected | PASS | ALIF | successful |

TC001_01_02 : Test Email Verification Delay

| Test Case ID (TC001_01_02) | Test Steps | Input Data | Expected Results | Actual Results | PASS/FAIL | Tested By | Tester Comments |
|--------------------------------------|------------------------|------------|-------------------------------------|----------------|-----------|-----------|-------------------------|
| 1 | Complete registration | | successfully registered | As Expected | PASS | ALIF | successfully registered |
| 2 | wait for delayed email | | email page show up | As Expected | PASS | ALIF | expected page show up |
| 3 | Click verify | | Registration succeeds despite delay | As Expected | PASS | ALIF | verification delay |

TC001_01_03 :Test Duplicate Email Registration

| Test Case ID (TC001_01_03) | Test Steps | Input Data | Expected Results | Actual Results | PASS/FAIL | Tested By | Tester Comments |
|--------------------------------------|--|---|---|----------------|-----------|-----------|-------------------------|
| 1 | Attempt registration with existing email | email: alif2505@gmail.com (already registered). Password: AnotherP@ssword456 | Error message displayed:: "This email is already registered." | As Expected | PASS | ALIF | Error message displayed |

8.2 TC002: Test Package User Management Subsystem: Customer Login (US002)

This test contains the following test cases:

- (a) **TC002_01**: Test login with valid credentials (SD002).
- (b) **TC002_02**: Test login with invalid credentials (SD002).
- (c) **TC002_03**: Test login with account lockout after multiple failed attempts (SD002).

8.2.1 TC002_01: Test Normal Login Scenario

| Test Case ID (TC002_01_01) | Test Steps | Input Data | Expected Results | Actual Results | PASS/FAIL | Tested By | Tester Comments |
|---------------------------------------|-------------------------------------|---|--|----------------|-----------|-----------|-----------------------|
| 1 | Navigate to login page | | The landing page should be open | As Expected | PASS | ALIF | expected page show up |
| 2 | enter correct username and password | Name: Alif Azman Password: SecureP@ss123 | No error message | As Expected | PASS | ALIF | Successful |
| 3 | submit | | User is logged in and redirected to homepage | As Expected | PASS | ALIF | expected page show up |

8.2.2 TC002_02: Test Invalid Login

| Test Case ID (TC002_02_01) | Test Steps | Input Data | Expected Results | Actual Results | PASS/FAIL | Tested By | Tester Comments |
|---------------------------------------|-------------------------------|---|---|----------------|-----------|-----------|-----------------|
| 1 | Enter wrong username/password | Name: John Doe Password: SecureP@ss123 | Error message displayed:: "wrong username or password,please try again" | As Expected | PASS | ALIF | As Expected |

8.2.3 TC002_03: Account Lockout

| Test Case ID (TC002_03_01) | Test Steps | Input Data | Expected Results | Actual Results | PASS/FAIL | Tested By | Tester Comments |
|---------------------------------------|---|---|--|----------------|-----------|-----------|-----------------|
| 1 | Attempt login with wrong password 5 times | Name: John Doe Password: SecurePass123 | Error message displayed:: "wrong password,click forget password to reset it" | As Expected | PASS | ALIF | As Expected |

8.3 TC003: Test Package User Management Subsystem: Email Verification (US002)

This test contains the following test cases:

- (a) **TC003_01:** Test normal email verification (SD002).
- (b) **TC003_02:** Test expired email verification link (SD002).
- (c) **TC003_03:** Test invalid email verification link (SD002).

8.3.1 TC003_01: Normal Email Verification

| Test Case ID (TC003_01_01) | Test Steps | Input Data | Expected Results | Actual Results | PASS/FAIL | Tested By | Tester Comments |
|---------------------------------------|---------------|------------|---------------------------------|----------------|-----------|-----------|-----------------|
| 1 | Register | | The landing page should be open | As Expected | PASS | ALIF | As Expected |
| 2 | receive email | | No error message | As Expected | PASS | ALIF | As Expected |
| 3 | click verify | | Account activated | As Expected | PASS | ALIF | As Expected |

8.3.2 TC003_02: Expired Link

| Test Case ID (TC003_02_01) | Test Steps | Input Data | Expected Results | Actual Results | PASS/FAIL | Tested By | Tester Comments |
|--------------------------------------|-------------------------|------------|---|----------------|-----------|-----------|-----------------|
| 1 | Register | | The landing page should be open | As Expected | PASS | ALIF | As Expected |
| 2 | wait until link expires | | No error message | As Expected | PASS | ALIF | As Expected |
| 3 | click verify | | Error message displayed:"link expired,please click resend option" | As Expected | PASS | ALIF | As Expected |

8.3.3 TC003_03: Invalid Link

| Test Case ID (TC003_03_01) | Test Steps | Input Data | Expected Results | Actual Results | PASS/FAIL | Tested By | Tester Comments |
|--------------------------------------|----------------------|------------|---------------------------------|----------------|-----------|-----------|-----------------|
| 1 | Register | | The landing page should be open | As Expected | PASS | ALIF | As Expected |
| 2 | click incorrect link | | Error message displayed | As Expected | PASS | ALIF | As Expected |

8.4 TC004: TestPackage User Management Subsystem: Customer Logout (US002)

This test contains the following test cases:

- (a) **TC004_01**: Test normal Logout (SD002).

8.4.1 TC004_01: Normal Logout

| Test Case ID (TC004_01_01) | Test Steps | Input Data | Expected Results | Actual Results | PASS/ FAIL | Tested By | Tester Comments |
|---------------------------------------|---------------------|------------|---|----------------|------------|-----------|-----------------|
| 1 | login | | The landing page should be open | As Expected | PASS | ALIF | As Expected |
| 2 | click logout button | | User is logged out and redirected to login page | As Expected | PASS | ALIF | As Expected |

8.5 TC005: Test Package User Management Subsystem: Viewing Homepage (US002)

This test contains the following test cases:

- (a) **TC005_01**: Test homepage display for logged-in user (SD002).

- (b) **TC005_02**: Test error display when homepage fails to load (SD002).

8.5.1 TC005_01: Normal Homepage Display

| Test Case ID (TC005_01_01) | Test Steps | Input Data | Expected Results | Actual Results | PASS/F AIL | Tested By | Tester Comments |
|---------------------------------------|----------------------|------------|---------------------------------|----------------|------------|-----------|-----------------|
| 1 | login | | The landing page should be open | As Expected | PASS | ALIF | As Expected |
| 2 | navigate homepage to | | All content loads correctly | As Expected | PASS | ALIF | As Expected |

8.5.2 TC005_02: Error Display

| Test Case ID (TC005_02_01) | Test Steps | Input Data | Expected Results | Actual Results | PASS/F FAIL | Tested By | Tester Comments |
|--------------------------------------|------------------------|------------|--|----------------|-------------|-----------|-----------------|
| 1 | Simulate server outage | | Error message displayed : Unable to load content | As Expected | PASS | ALIF | As Expected |

8.6 TC006: Test Package User Management Subsystem: Viewing Menu (US002)

This test contains the following test cases:

(a) **TC006_01:** Test menu visibility for logged-in user (SD002).

8.6.1 TC006_01: Normal Menu Visibility

| Test Case ID (TC006_01_01) | Test Steps | Input Data | Expected Results | Actual Results | PASS/F FAIL | Tested By | Tester Comments |
|--------------------------------------|-----------------------|------------|---------------------------------|----------------|-------------|-----------|-----------------|
| 1 | Log in | | The landing page should be open | As Expected | PASS | ALIF | As Expected |
| 2 | navigate to menu page | | Menu items displayed correctly | As Expected | PASS | ALIF | As Expected |

8.7 TC007: Test Package User Management Subsystem: Forgot Password (US002)

This test contains the following test cases:

- (a) **TC007_01**: Test changing password functionality (SD002).

8.7.1 TC007_01: Normal Forgot Password Flow

| Test Case ID (TC007_01_01) | Test Steps | Input Data | Expected Results | Actual Results | PASS/FAIL | Tested By | Tester Comments |
|---------------------------------------|-------------------------|------------|--|----------------|-----------|-----------|-----------------|
| 1 | Click "Forgot Password" | | no error message | As Expected | PASS | ALIF | As Expected |
| 2 | enter email | | No error message | As Expected | PASS | ALIF | As Expected |
| 3 | reset password | | password reset email sent and user can log in with a new password. | As Expected | PASS | ALIF | As Expected |

8.8 TC008: Test Package User Management Subsystem: Changing Password (US002)

This test contains the following test cases:

- (a) **TC008_01**: Test changing password functionality (SD002).

8.8.1 TC008_01: Normal Password Change

| Test Case ID (TC007_01_01) | Test Steps | Input Data | Expected Results | Actual Results | PASS/FAIL | Tested By | Tester Comments |
|---------------------------------------|-----------------------------|------------|------------------|----------------|-----------|-----------|-----------------|
| 1 | login | | no error message | As Expected | PASS | ALIF | As Expected |
| 2 | navigate to change password | | No error message | As Expected | PASS | ALIF | As Expected |

| | | | | | | | |
|---|--------------------------------|--|-------------------------------|-------------|------|------|-------------|
| 3 | enter current and new password | | Password updated successfully | As Expected | PASS | ALIF | As Expected |
|---|--------------------------------|--|-------------------------------|-------------|------|------|-------------|

8.9 TC009: Test Package User Management Subsystem: Viewing Profile (US003)

This test contains the following test cases:

(a) **TC009_01**: Test profile visibility for logged-in user (SD003).

8.9.1 TC009_01: Normal Profile Viewing

| Test Case ID (TC009_01_01) | Test Steps | Input Data | Expected Results | Actual Results | PASS/FAIL | Tested By | Tester Comments |
|---------------------------------------|--------------------------|------------|--|----------------|-----------|-----------|-----------------|
| 1 | login | | The landing page should be open | As Expected | PASS | ALIF | As Expected |
| 2 | navigate to profile page | | All profile information displayed correctly. | As Expected | PASS | ALIF | As Expected |

8.10 TC010: Test Package User Management Subsystem: Editing Profile (US003)

This test contains the following test cases:

(a) **TC010_01**: Test editing profile functionality (SD003).

8.10.1 TC010_01: Normal Profile Edit

| Test Case ID (TC010_01_01) | Test Steps | Input Data | Expected Results | Actual Results | PASS/ FAIL | Tested By | Tester Comments |
|---------------------------------------|--|------------|--|----------------|------------|-------------------|---|
| 1 | login | | The landing page should be open | As Expected | PASS | SITI NUR SYAKIRAH | As Expected |
| 2 | navigate to edit profile | | All profile information displayed correctly. | As Expected | PASS | SITI NUR SYAKIRAH | As Expected |
| 3 | click "Edit User Detail" button | | Information that can be edit: - Username - Email - Date of Birth - Address - Mobile Number - Country - Gender | As Expected | PASS | SITI NUR SYAKIRAH | As Expected |
| 4 | save changes by click on "Update Profile" button | | Profile updated successfully | As Expected | PASS | SITI NUR SYAKIRAH | Successful update and message box success appears |

8.11 TC011 : Test Package Admin Subsystem : Admin Login (US009)

This test contains the following test cases:

(a) **TC011_01**: Test admin login with valid credentials (SD009).

(b) **TC011_02**: Test admin login with invalid credentials (SD009).

8.11.1 TC011_01: Normal Admin Login

| Test Case ID (TC011_01_01) | Test Steps | Input Data | Expected Results | Actual Results | PASS/FAIL | Tested By | Tester Comments |
|-------------------------------|-------------------------|---|------------------|----------------|-----------|-------------------|------------------|
| 1 | valid username/password | email: syakirah12@ mail.com (already registered). Password: AnotherA@ssword456 | Admin logged in. | As Expected | PASS | SITI NUR SYAKIRAH | Success to login |

8.11.2 TC011_02: Invalid Admin Login

| Test Case ID (TC011_02_01) | Test Steps | Input Data | Expected Results | Actual Results | PASS/FAIL | Tested By | Tester Comments |
|-------------------------------|-----------------------------|--|-------------------------|----------------|-----------|-------------------|--|
| 1 | incorrect username/password | email: unknown@ mail.com (already registered). Password: AnotherA@Psword456 | Error message displayed | As Expected | PASS | SITI NUR SYAKIRAH | Message box show as invalid username or password |

8.12 TC012 : Test Package Admin Subsystem : Admin Logout (US011)

This test contains the following test cases:

- (a) **TC012_01**: Test admin logout functionality (SD011).

8.12.1 TC012_01: Normal Admin Logout

| Test Case ID (TC012_01_01) | Test Steps | Input Data | Expected Results | Actual Results | PASS /FAIL | Tested By | Tester Comments |
|---------------------------------------|-----------------|---|--|----------------|------------|-------------------|--|
| 1 | Log in as admin | email: syakirah12@ g mail.com (already registered). Password: AnotherA@ss word456 | The landing page should be open | As Expected | PASS | SITI NUR SYAKIRAH | Success to login |
| 2 | click logout | | Admin is logged out and redirected to login page | As Expected | PASS | SITI NUR SYAKIRAH | Message box appear to state that user have log out |

8.13 TC013: Test Package Dengue Information Subsystem : Dengue Map (US004)

This test contains the following test cases:

- (a) **TC013_01**: Test viewing the dengue map with data (SD004).

- (b) **TC013_02**: Test filtering dengue map by location or date (SD004).

8.13.1 TC013_01: Normal Dengue Map Viewing

| Test Case ID (TC013_01_01) | Test Steps | Input Data | Expected Results | Actual Results | PASS/ FAIL | Tested By | Tester Comments |
|---------------------------------------|---|------------|---------------------------------|----------------|------------|-------------------|---|
| 1 | Log in | | The landing page should be open | As Expected | PASS | SITI NUR SYAKIRAH | Success to login |
| 2 | navigate to dengue map by click "View Dengue Map" | | Dengue data displayed correctly | As Expected | PASS | SITI NUR SYAKIRAH | As Expected, it display dengue map data |

| | | | | | | | |
|--|-------------------------|--|--|--|--|--|--|
| | Data” at navigation bar | | | | | | |
|--|-------------------------|--|--|--|--|--|--|

8.13.2 TC013_02: Filtering Map Data

| Test Case ID (TC013_02_01) | Test Steps | Input Data | Expected Results | Actual Results | PASS/FAIL | Tested By | Tester Comments |
|---------------------------------------|-------------------------------|---|-------------------------|----------------|-----------|-------------------|-----------------|
| 1 | Apply filters (location/date) | Location: Malaysia Date: 20/11/20 24 | Filtered displayed data | As Expected | PASS | SITI NUR SYAKIRAH | As Expected |

8.14 TC014: Test Package Dengue Information Subsystem : Reporting Mosquito Breeding Sites (US005)

This test contains the following test cases:

(a) TC014_01: Test submitting a report with valid details (SD005).

(b) TC014_02: Test error handling for invalid report submissions (SD005).

8.14.1 TC014_01: Normal Report Submission

| Test Case ID (TC014_01_01) | Test Steps | Input Data | Expected Results | Actual Results | PASS /FAIL | Tested By | Tester Comments |
|--------------------------------------|------------------|---------------------------------------|--|----------------|------------|-------------------|-----------------|
| 1 | Fill report form | Click “Report Mosquito Breeding Site” | The landing page should be open | As Expected | PASS | SITI NUR SYAKIRAH | As Expected |
| 2 | Fill in detail | | Information that can be fill in: - Location - Date - Upload Photo | As Expected | PASS | SITI NUR SYAKIRAH | As Expected |

| | | | | | | | |
|---|--------------|---------------------------------|--------------------------------------|-------------|------|-------------------|--|
| 3 | attach photo | | All information displayed correctly. | As Expected | PASS | SITI NUR SYAKIRAH | As Expected |
| 4 | submit | Click on “Submit Report” button | Report submitted successfully | As Expected | PASS | SITI NUR SYAKIRAH | Message box appear show that the report have been successful submitted |

8.14.2 TC014_02: Invalid Report Submission

| Test Case ID (TC014_02_01) | Test Steps | Input Data | Expected Results | Actual Results | PASS/FAIL | Tested By | Tester Comments |
|----------------------------|-----------------------------|---------------------------------|---------------------------|----------------|-----------|-------------------|--|
| 1 | Leave required fields blank | | invalid report submission | As Expected | PASS | SITI NUR SYAKIRAH | As Expected |
| 2 | submit | Click on “Submit Report” button | Error message displayed. | As Expected | PASS | SITI NUR SYAKIRAH | Message box appear show that the report failed to submit |

8.15 TC015: Test Package Learning Engagement Subsystem : Viewing Educational Videos (US008)

This test contains the following test cases:

(a) **TC015_01**: Test video playback functionality (SD008).

8.15.1 TC015_01: Normal Video Playback

| Test Case ID (TC015_01_01) | Test Steps | Input Data | Expected Results | Actual Results | PASS/FAIL | Tested By | Tester Comments |
|----------------------------|------------|--|--|----------------|-----------|-------------------|-----------------|
| 1 | View video | Click on “Watch Video” on navigation bar | There are many video that related to dengue can be watched | As Expected | PASS | SITI NUR SYAKIRAH | As Expected |

| | | | | | | | |
|---|--------------|--|--|-------------|------|-------------------|-------------|
| 2 | Select video | | The landing page should should be open | As Expected | PASS | SITI NUR SYAKIRAH | As Expected |
| 3 | play | | Video plays without interruption. | As Expected | PASS | SITI NUR SYAKIRAH | As Expected |

8.16 TC016: Test Package Learning Engagement Subsystem : Taking Quizzes (US006)

This test contains the following test cases:

- (a) **TC016_01**: Test completing a quiz (SD006).
- (b) **TC016_02**: Test viewing quiz results (SD006).

8.16.1 TC016_01: Normal Quiz Completion

| Test Case ID (TC016_01_01) | Test Steps | Input Data | Expected Results | Actual Results | PASS/FAIL | Tested By | Tester Comments |
|--------------------------------------|-----------------------|------------|--|----------------|-----------|-------------------|-----------------|
| 1 | Answer quiz questions | | The landing page should should be open | As Expected | PASS | SITI NUR SYAKIRAH | As Expected |
| 2 | submit answer | | Quiz submitted and score displayed immediately | As Expected | PASS | SITI NUR SYAKIRAH | As Expected |

8.16.2 TC016_02: Viewing Quiz Results

| Test Case ID (TC016_02_01) | Test Steps | Input Data | Expected Results | Actual Results | PASS/FAIL | Tested By | Tester Comments |
|--------------------------------------|---------------|------------|-----------------------------------|----------------|-----------|-------------------|-----------------|
| 1 | Complete quiz | | The landing page should be open | As Expected | PASS | SITI NUR SYAKIRAH | As Expected |
| 2 | view results | | Quiz results displayed correctly. | As Expected | PASS | SITI NUR SYAKIRAH | As Expected |

8.17 TC017 : Test Package Admin Subsystem : Admin Managing Reports (US0012)

This test contains the following test cases:

- (a) **TC017_01:** Test approving a report (SD012).
- (b) **TC017_02:** Test rejecting a report (SD012).

8.17.1 TC017_01: Approve Report

| Test Case ID (TC017_01_01) | Test Steps | Input Data | Expected Results | Actual Results | PASS/FAIL | Tested By | Tester Comments |
|--------------------------------------|----------------|------------|---------------------------------|----------------|-----------|-------------------|-----------------|
| 1 | Review report | | The landing page should be open | As Expected | PASS | SITI NUR SYAKIRAH | As Expected |
| 2 | click approve. | | Report marked as approved | As Expected | PASS | SITI NUR SYAKIRAH | As Expected |

8.17.2 TC017_02: Reject Report

| Test Case ID (TC017_02_01) | Test Steps | Input Data | Expected Results | Actual Results | PASS/FAIL | Tested By | Tester Comments |
|--------------------------------------|---------------|------------|---------------------------------|----------------|-----------|-------------------|-----------------|
| 1 | Review report | | The landing page should be open | As Expected | PASS | SITI NUR SYAKIRAH | As Expected |
| 2 | click reject | | Report marked as rejected | As Expected | PASS | SITI NUR SYAKIRAH | As Expected |

8.18 TC018: Test Package Admin Subsystem : Manage Quiz (US018)

This test contains the following test cases:

- (a) **TC018_01**: Test adding a new quiz (Normal flow).
- (b) **TC018_02**: Test editing an existing quiz (Alternate flow).
- (c) **TC018_03**: Test deleting a quiz (Error handling)

8.18.1 TC018_01: Add New Quiz

| Test Case ID (TC018_01_01) | Test Steps | Input Data | Expected Results | Actual Results | PASS/FAIL | Tested By | Tester Comments |
|--------------------------------------|--------------------------|--|------------------------------------|----------------|-----------|-------------------|-----------------|
| 1 | Navigate to Quiz Section | | The quiz management page is opened | As Expected | PASS | SITI NUR SYAKIRAH | As Expected |
| 2 | Click Add Quiz | Quiz title, description, and questions | Quiz is successfully added | As Expected | PASS | SITI NUR SYAKIRAH | As Expected |

8.18.2 TC018_02: Add An Existing Quiz

| Test Case ID (TC018_02_01) | Test Steps | Input Data | Expected Results | Actual Results | PASS/FAIL | Tested By | Tester Comments |
|--------------------------------------|--------------------------|-----------------------------|------------------------------------|----------------|-----------|-------------------|-----------------|
| 1 | Navigate to Quiz Section | | The quiz management page is opened | As Expected | PASS | SITI NUR SYAKIRAH | As Expected |
| 2 | Click Edit Quiz | Modified title or questions | Quiz details updated successfully | As Expected | PASS | SITI NUR SYAKIRAH | As Expected |

8.18.3 TC018_02: Delete Quiz

| Test Case ID (TC018_03_01) | Test Steps | Input Data | Expected Results | Actual Results | PASS/FAIL | Tested By | Tester Comments |
|--------------------------------------|--------------------------|-------------------------|------------------------------------|----------------|-----------|-------------------|-----------------|
| 1 | Navigate to Quiz Section | | The quiz management page is opened | As Expected | PASS | SITI NUR SYAKIRAH | As Expected |
| 2 | Click Delete Quiz | Selected quiz to delete | Quiz removed from the list | As Expected | PASS | SITI NUR SYAKIRAH | As Expected |

9. Traceability Matrix

A Traceability Matrix is a traceability that establishes traceable links between various project elements, such as user requirements, system requirements, design documents, test cases, and defects. With the traceability of this matrix, it allows stakeholders to track the progress of requirements from their origin to the final implementation and testing phase. Below we provide a table traceability matrix related to Use Case Diagram, User Story, Sequence Diagram and Test Case.

Table 9.1: Traceability Matrix for <DPCM System>

| Sprint No | Package Item | User Story ID | User Story Description | Sequence Diagram ID | Sequence Diagram Description | Test Case ID |
|-----------|-------------------------|---------------|--|---------------------|--|---|
| Sprint 1 | Package User Management | US01 | As a guest, I want to sign up as user for application So that i can test the full application | SD001 | Illustrates the steps for guest registration, including entering personal details, submitting the registration form, and receiving confirmation of account creation. | TC001 <ul style="list-style-type: none"> • TC001_01_01 • TC001_02_01 • TC001_03_01 |
| Sprint 1 | Package User Management | US02 | | SD002 | Depicts the user login process, including inputting | TC002 <ul style="list-style-type: none"> • TC002_01_01 • TC002_02_01 • TC002_03_01 |

| | | | | | | |
|----------|----------------------------|------|---|-------|--|--|
| | | | | | <p>credentials, verifying authentication, and granting access to the system.</p> | <p>TC003</p> <ul style="list-style-type: none"> • TC003_01_01 • TC003_02_01 • TC003_03_01 <p>TC004</p> <ul style="list-style-type: none"> • TC004_01_01 <p>TC005</p> <ul style="list-style-type: none"> • TC005_01_01 • TC005_02_01 <p>TC006</p> <ul style="list-style-type: none"> • TC006_01_01 <p>TC007</p> <ul style="list-style-type: none"> • TC007_01_01 <p>TC008</p> <ul style="list-style-type: none"> • TC008_01_01 |
| Sprint 1 | Package User Management | US03 | As a user application DPCM I want to manage profile as user in application So that i can view and edit user profile | SD003 | | <p>TC009</p> <ul style="list-style-type: none"> • TC009_01_01 <p>TC010</p> <ul style="list-style-type: none"> • TC010_01_01 |
| Sprint 2 | Package Dengue Information | US04 | As a user application DPCM I would like to see about dengue data So that i can know the latest dengue data | SD004 | Shows the process to access dengue map data | <p>TC013</p> <ul style="list-style-type: none"> • TC013_01_01 • TC013_02_01 |
| Sprint 2 | Package Dengue Information | US05 | As a user application DPCM I want to report about mosquitoes breeding sites | SD005 | Shows the process of entering report details and saving report details | <p>TC014</p> <ul style="list-style-type: none"> • TC014_01_01 • TC014_02_01 |

| | | | | | | |
|----------|-----------------------------|-------|--|-------|--|------------------------|
| | | | So that application can update data from user | | | |
| Sprint 3 | Package Learning Engagement | US006 | As a user application DPCM I want to take quizzes So that i can test my knowledge about dengue | SD006 | Shows the process of accessing quizzes, answering the questions in the quiz and saving the answers | TC016 • TC006_01_01 |
| Sprint 3 | Package Learning Engagement | US007 | As a user application DPCM I want to see the result for quizzes So that i can know the score for quizzes | SD007 | Shows the process of answering quiz questions and displaying the results | TC016 • TC006_02_01 |
| Sprint 3 | Package Learning Engagement | US008 | As a user application DPCM I want to see videos about dengue so that I understand more about dengue fever and dengue prevention | SD008 | Shows the process for selecting a video, requesting a video stream, and viewing a video | TC015 • TC015_01_01 |

| | | | | | | |
|----------|---------------|-------|---|-------|---|--|
| Sprint 4 | Package Admin | US009 | As a admin application DPCM I want to login as admin for application So that i can manage the admin dashboard | SD009 | Depicts the user login process, including inputting credentials, verifying authentication, and granting access to the system. | TC011 <ul style="list-style-type: none"> • TC011_01_01 • TC011_02_01 |
| Sprint 4 | Package Admin | US010 | As a admin application DPCM I want to login as admin So that i can manage the application | SD010 | Shows the process to manage the user profile by changing the picture, password and personal information and saving the changes. | TC010 <ul style="list-style-type: none"> - TC010_01 |
| Sprint 5 | Package Admin | US011 | As an admin application DPCM I want to see and verify the report So that I can update the dengue map data | SD011 | Shows the process of selecting data and retrieving the data, approve the report and update the report or reject the report | TC011 <ul style="list-style-type: none"> • TC011_01_01 • TC011_02_01 TC012 <ul style="list-style-type: none"> • TC012_01_01 |

| | | | | | | |
|----------|---------------|-------|--|-------|---|---|
| Sprint 5 | Package Admin | US012 | As a admin for application DPCM I want to update the data about dengue case So that user can see the up to date data | US12 | Shows the process of accessing dengue page, display dengue statistics form, change dengue data and save changes | TC017 <ul style="list-style-type: none"> • TC017_01_01 • TC017_02_01 |
| Sprint 5 | Package Admin | US013 | As a admin for application DPCM I want to manage quizzes question So that user can see different question in quiz | SD013 | Show the process of accessing the quiz management page, displaying the list of existing quizzes, adding new quiz questions, editing existing questions, and deleting unwanted questions | TC018 <ul style="list-style-type: none"> • TC018_01_01 • TC018_02_01 • TC018_03_01 |

Appendix

