TRIMESTER March/April, 2025

CSE6224 Software Requirements Engineering

PROJECT 2

Campus Accessibility Navigation System with Facilities and Event Integration

Name	Student ID	Course
Alimzhan	1231301318	Bachelor of Computer Science
GALAL OSMAN GALAL MOHAMED	1221307698	Bachelor of Computer Science
Aishah Azman	1231302725	Bachelor of Computer Science
Ahmed	1221304386	Bachelor of Computer Science

Table of Content

1.Introduction

1.1 Purpose

The purpose of this system is to provide an accessibility-focused navigation and information platform for campus staff and students. It aims to facilitate convenient access to real-time information about campus facilities, events, and individual class schedules, thereby enhancing the campus experience and addressing the needs of users with mobility challenges.

1.2 Scope

The purpose of this system is to provide a mobile-based, accessibility-enhanced navigation and information platform for MMU campus stakeholders, including students and staff. It aims to enhance mobility, inclusivity, and real-time access to campus information, especially for users with disabilities, by integrating class schedules, facility data, and event information into a unified interface.

1.3 Product overview

1.3.1 Product perspective

The system acts as a complementary digital service within the campus ecosystem, integrating with existing campus facilities management and event calendar systems. It provides a unified mobile interface supporting accessible navigation and real-time information updates.

1.3.2 Product functions

- Query detailed information and real-time status of campus facilities such as parking lots, study rooms, and restrooms.
- Browse campus maps with clickable locations linked to Google Maps for navigation.
- View detailed information on campus events including date, time, and location.
- Access personalized class schedules (students only)
- Receive push notifications about facility maintenance, event changes, and class schedule updates.
- View latest news and announcements issued by the university.
- Allow users to customize notification preferences based on urgency and type

For Administrators:

- Manage campus event listings and facility status.
- Trigger and configure system-wide push notifications.

Table of Content

1.Introduction

1.1 Purpose

The purpose of this system is to provide an accessibility-focused navigation and information platform for campus staff and students. It aims to facilitate convenient access to real-time information about campus facilities, events, and individual class schedules, thereby enhancing the campus experience and addressing the needs of users with mobility challenges.

1.2 Scope

The purpose of this system is to provide a mobile-based, accessibility-enhanced navigation and information platform for MMU campus stakeholders, including students and staff. It aims to enhance mobility, inclusivity, and real-time access to campus information, especially for users with disabilities, by integrating class schedules, facility data, and event information into a unified interface.

1.3 Product overview

1.3.1 Product perspective

The system acts as a complementary digital service within the campus ecosystem, integrating with existing campus facilities management and event calendar systems. It provides a unified mobile interface supporting accessible navigation and real-time information updates.

1.3.2 Product functions

- Query detailed information and real-time status of campus facilities such as parking lots, study rooms, and restrooms.
- Browse campus maps with clickable locations linked to Google Maps for navigation.
- View detailed information on campus events including date, time, and location.
- Access personalized class schedules (students only)
- Receive push notifications about facility maintenance, event changes, and class schedule updates.

- View latest news and announcements issued by the university.
- Allow users to customize notification preferences based on urgency and type

For Administrators:

- Manage campus event listings and facility status.
- Trigger and configure system-wide push notifications.

1.3.3 User characteristics

The system is designed for three primary user groups:

- **Students**: Technically literate users familiar with mobile apps. They rely on the system for class schedules, campus navigation, and notifications.
- **Staff**: Non-technical users who need access to facility and event information and receive timely notifications relevant to their roles.
- **Administrators**: Technically trained users responsible for managing campus data, events, and notification workflows through a restricted-access interface.

The system is optimized for users with basic to intermediate smartphone proficiency and includes features that address accessibility needs (e.g., screen reader support, contrast settings) for users with visual or motor impairments.

1.3.4 Limitations

The system has the following known limitations:

- It does not support offline map navigation; an internet connection is required for all real-time functions.
- Multi-language support is not implemented in the current version.

- The system relies on third-party services such as the Google Maps API and campus management systems; any service interruptions may affect functionality.
- Notification delivery is dependent on mobile platform services such as Firebase Cloud Messaging (Android) and Apple Push Notification Service (iOS).

1.4 Definitions

- **Accessibility Navigation**: Navigation designed specifically to accommodate users with mobility challenges.
- **Facility Status**: Real-time availability and condition of campus facilities.
- **Notification System**: A module that pushes important campus messages and alerts to users.
- **Push Notifications:** Automated alerts sent to user devices to inform them of important updates.
- **Kano Model:** A model to classify requirements into Must-be, Performance, and Excitement categories based on user satisfaction.

2 References

1. IEEE Std 29148-2018, *Systems and software engineering — Life cycle processes — Requirements engineering*, IEEE Standards Association, 2018.
2. Kano, N., Seraku, N., Takahashi, F., & Tsuji, S. (1984). *Attractive quality and must-be quality*. Journal of the Japanese Society for Quality Control, 14(2), 39–48.

Development Tools and APIs:

3. Google Maps Platform. *Google Maps API Documentation*. [Online]. Available: https://developers.google.com/maps/documentation [Accessed: May 20, 2025].
4. Google Forms. Online Survey Tool. [Online]. Available: https://forms.gle/HVwtyEFH2FwHaKaq9 [Accessed: May 18, 2025].
5. Figma. *Design and Prototyping Tool*. [Online]. Available: https://figma.com [Accessed: May 15, 2025].

Legal Compliance:

6. Personal Data Protection Act (PDPA) 2010, Malaysia. [Online]. Available: https://www.pdp.gov.my [Accessed: May 2025].
7. General Data Protection Regulation (GDPR), European Union, 2016. [Online]. Available: https://gdpr.eu [Accessed: May 2025].
8. 8. Personal interviews and prototyping feedback sessions conducted with MMU academic and administrative staff, May 2025.

3 Requirement

3.1 Functions

- Users shall be able to query detailed information and real-time status of campus facilities (e.g., parking availability, restrooms).
- Users shall be able to browse the campus map and click on locations to open Google Maps navigation.
- The system shall display comprehensive campus event information, including date, time, location, and details.
- Students shall be able to view their individual class schedules.
- The system shall deliver notifications regarding facility maintenance, event updates, and class schedule changes.
- Users shall be able to access the latest campus news and announcements.
- Users shall be able to customize notification preferences by type and urgency.

3.1.1 Requirements Prioritization and Classification

- Requirements are classified using the **Kano Model**, with reference to user feedback gathered during prototyping.

	ID	Kano Category	Requirement Description	Primary User Role
1	FR - 01	Basic	The system shall provide navigation to campus facilities and event locations.	Student/Staff
2	FR - 02	Basic	The system shall display accurate, accessible maps with route guidance for disabled users.	Student
3	FR - 03	Basic	The system shall display event locations and timing	Student/Staff
4	FR - 04	Basic	The interface shall provide	All

			clear digital signage and labels for facility status and events.	
5	FR - 05	Performance	The system shall provide real-time updates on facility status and event changes.	All
6	FR - 06	Performance	The system shall automatically sync with university calendars and timetables.	All
7	FR - 07	Performance	The system shall support personalized scheduling and reminders for students and staff.	Student/Staff
8	FR - 08	Performance	The system shall prioritize alerts based on user role and urgency.	All
9	FR - 09	Excitement	The system shall support indoor navigation within complex buildings.	Student
10	FR - 10	Excitement	The system shall support voice or AR-assisted navigation features.	Student
11	FR - 11	Excitement	The user shall be able to customize notification filters by urgency and category.	Student/Staff
12	FR - 12	Excitement	The system shall offer smart	Student

			suggestions like reminders based on user proximity and schedule.	
13	FR - 13	Indifferent	The system shall show general campus news unrelated to navigation or events.	All
14	FR - 14	Indifferent	The system shall offer basic issue reporting not tied to navigation (e.g., feedback forms).	All
15	FR - 15	Reverse	The system shall not send excessive or irrelevant notifications without filtering options.	All
16	FR - 16	Reverse	The system shall avoid tracking users without explicit consent.	All
17	FR - 17	Reverse	The system shall not require complex setup just to access maps or events.	All

3.2 Performance requirements

This section outlines the system's expected performance metrics, including response time, availability, scalability, and real-time processing. These requirements ensure the application remains responsive, reliable, and efficient under normal and peak usage conditions.

ID	Requirement Description	Type	Target Metric
NFR-01	The system shall respond to 95% of user requests within 3 seconds.	Response Time	≤ 3 seconds

NFR-02	The system shall deliver 95% of notifications within 1 minute of trigger event.	Notification Latency	≤ 1 minute
NFR-03	The system shall maintain an availability rate of at least 99.5% during operating hours.	Availability	≥ 99.5%
NFR-04	The system shall support 1000 concurrent users without functional degradation.	Scalability / Load	1000 concurrent users (min)

3.3 Usability requirements

This section defines the usability standards the system must meet to ensure a smooth, accessible, and user-friendly experience. It focuses on interface design, accessibility features, ease of navigation, and personalization to accommodate diverse user needs, including users with disabilities.

ID	Requirements Description	Usability Target	Primary Role
UR-01	The system shall provide an intuitive, mobile-first interface with large touch targets and minimal scrolling.	Designed for single-hand mobile use, WCAG 2.1 compliant	All
UR-02	The system shall support accessibility features including screen reader compatibility and high-contrast mode.	Meet WCAG 2.1 Level AA compliance	Students with disabilities
UR-03	The system shall require no more than 3 taps to access core features (e.g., map, schedule, events).	≤ 3 steps to access major functions	All
UR-04	Onboarding screens shall introduce app features with simple illustrations and tooltips.	First-use tutorial required	All

UR-05	All icons and labels shall use standard design conventions familiar to mobile users.	Familiar UI components, Material Design or iOS HIG	All
UR-06	The app shall remember user preferences (notification settings, language, etc.) across sessions.	Persistent user settings	Student/Staff
UR-07	The admin panel shall group actions logically and display real-time system status.	≤ 2 clicks to access critical management actions	Admin

3.4 Interface requirements

- The system shall integrate with the campus facilities management database via a RESTful API.
- The system shall integrate with the campus event management system via a RESTful API.
- The system shall use the Google Maps API for navigation features.
- Data transmission between the app and backend services shall be secured using HTTPS and encryption.

3.5 Logical database requirements

- The system shall maintain data tables for:
    - Facilities (including real-time status, capacity, and accessibility metadata)
    - Events (with attendee accessibility requirements)
    - Class schedules (linked to accessible locations)
    - Notifications (prioritized by user role and urgency)
- Relationships between these entities shall be logically defined to support efficient queries (e.g., indexing on facility_id).

1.1.1. Entity Relationship Diagram

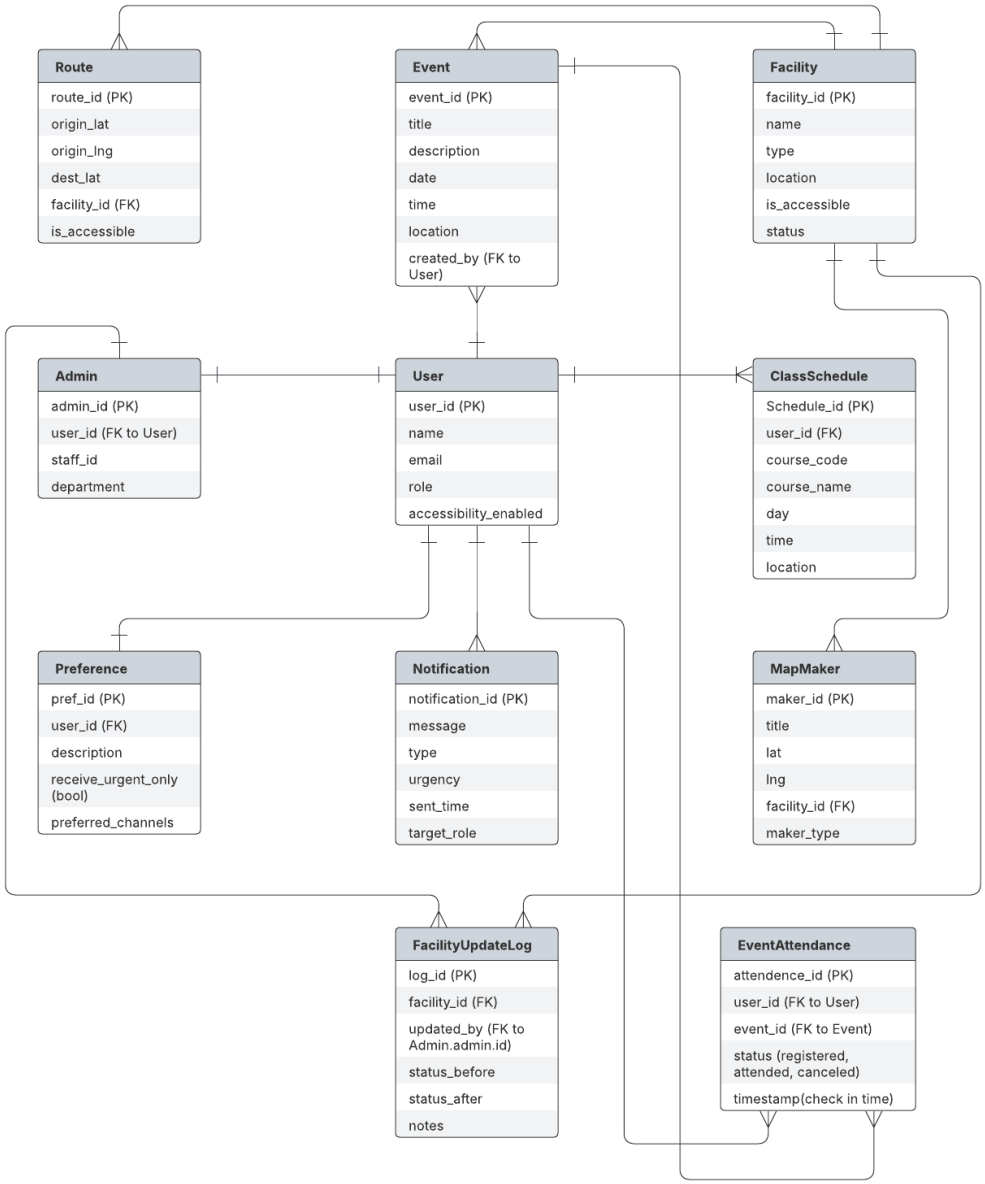


Diagram 3.1 : Entity Relationship Diagram

The Entity-Relationship Diagram (ERD) for the Campus Accessibility Navigation System models a structured, role-based architecture that supports the platform's core features. At the center of this design is the generalized `User` entity, which accommodates **students**, **staff**, **visitors**, and **admins** through a `role` attribute. Each role interacts with the system differently based on permissions and functional needs.

Students and staff access **class schedules**, **events**, **facility information**, and receive **notifications** personalized through the `Preference` entity. Visitors, on the other hand, are limited to viewing publicly accessible information like events and maps, without access to personalized features such as class schedules or preferences.

The `Admin` entity is an extension of `User`, containing additional administrative metadata such as `staff_id` and `department`. Admins have elevated permissions that allow them to manage `Facilities` and `Events`, and their actions are logged in the `FacilityUpdateLog` entity for traceability and audit purposes.

The system supports interactive campus navigation using the `Route` entity, which provides accessible pathways linked to campus `Facilities`. Each facility may have multiple `MapMarker` references for location rendering and visualization.

Events are created and managed via the `Event` entity, and participation is tracked using a Many-to-Many relationship between `User` and `Event`, resolved through the `EventAttendance` entity. This enables the system to register, track, and manage user attendance across multiple events.

The `Notification` entity delivers targeted push notifications to users based on urgency, type, and individual preferences. These messages may relate to event updates, facility disruptions, or class changes.

Overall, this ERD ensures normalized data flow, clean role separation, and supports all major platform functions such as personalized scheduling, accessible navigation, facility monitoring, event participation, and secure administrative operations — in alignment with MMU's accessibility and inclusivity goals.

3.6 Design constraints

- The system depends on real-time synchronization with existing campus management systems.
- The system shall be compatible with Android and iOS mobile platforms.
- The system must comply with relevant data privacy regulations (PDPA, GDPR).

3.7 Software system attributes

- **Reliability**

  The system shall ensure accurate and timely delivery of information and notifications.
- **Security**

  The system shall protect user data privacy and prevent unauthorized access.
- **Maintainability**

  The system shall be modularly designed to facilitate future updates and maintenance.
- **Privacy**

Ensure compliance with data protection laws, implement consent management and data anonymization.

- **Scalability**

  Support growing number of users and data without performance loss.

3.8 Supporting information

Documentation / Reference Sources:

- WCAG 2.1 guidelines for accessibility
- Google Maps API documentation
- Firebase Cloud Messaging (FCM) documentation
- PDPA / GDPR compliance notes
- MMU Campus Infrastructure Data (if applicable)

Tools/Design Sources:

- Figma (for UI prototypes)
- GitHub repo (version control and collaboration)
- Draw.io (for diagrams like ERD, Use Case)

3.9 Behavioural Models

3.9.1 Use Case Diagram

The *Navigate to Campus Facility* use case allows students and staff to search for and receive real-time directions to specific campus facilities such as restrooms, parking lots, or study rooms. Upon selecting a facility, the system displays a route using campus maps integrated with external navigation services. If the user has accessibility mode enabled, the system extends this behavior by prioritizing accessible pathways and enabling features such as screen reader compatibility and voice-guided navigation. This use case supports both general and inclusive mobility, ensuring users can move efficiently across campus environments.

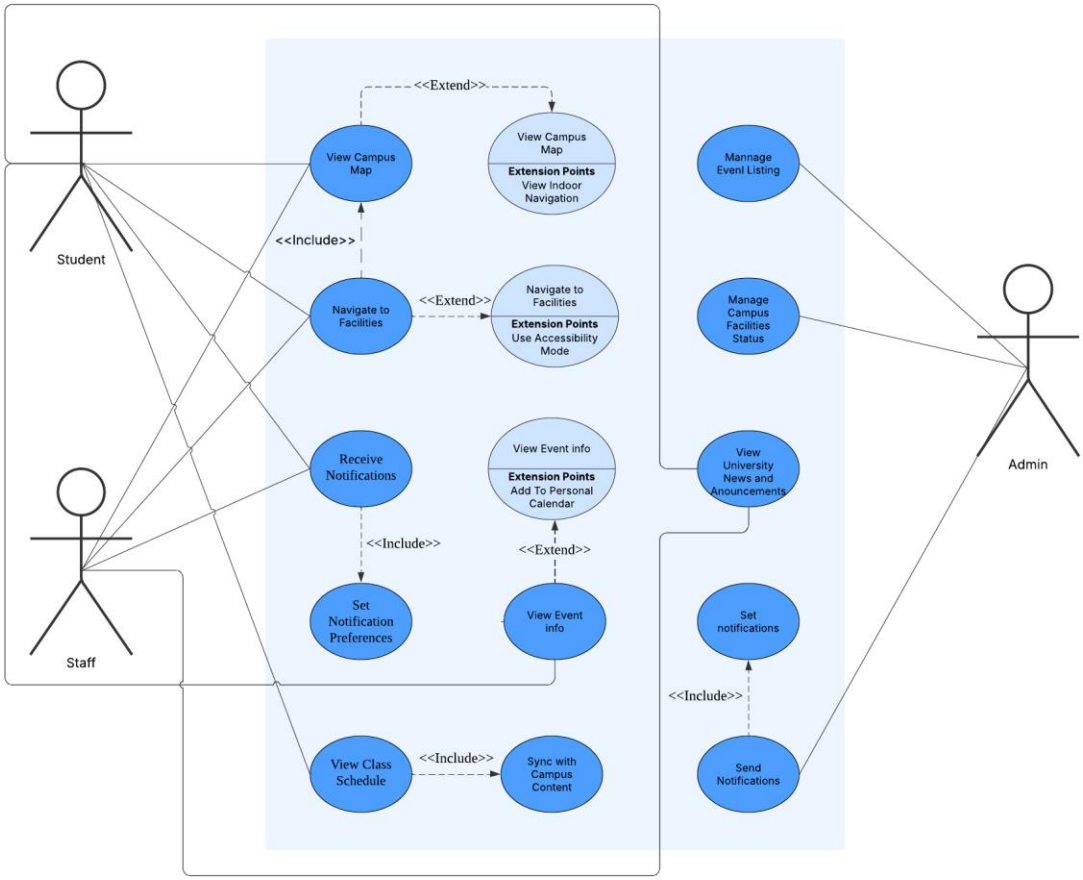


Diagram 3.2 : Use Case Diagram

3.9.2

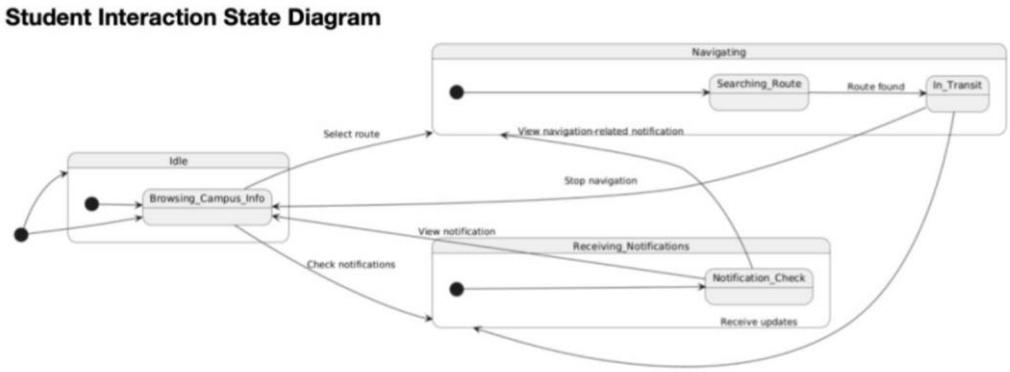


Diagram 3.3 : Student Interaction State Diagram

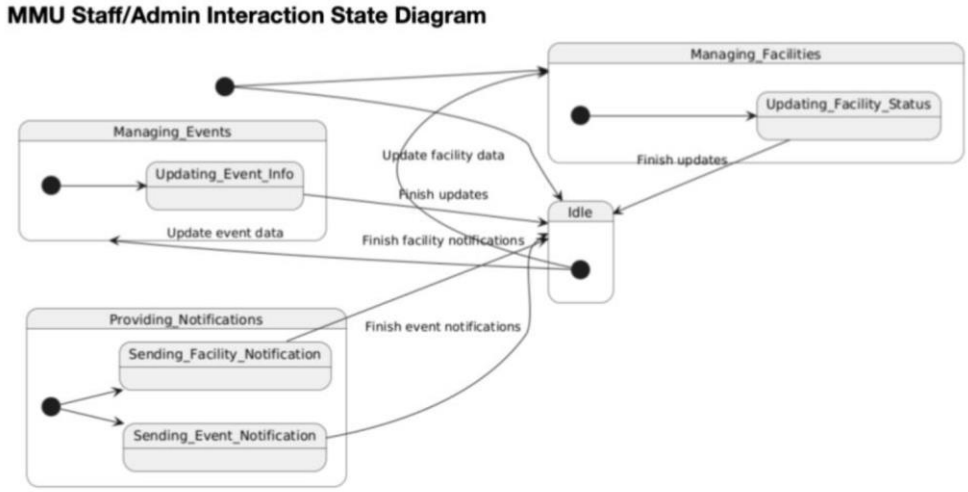


Diagram 3.4 : MMU Staff /Admin Interaction State Diagram

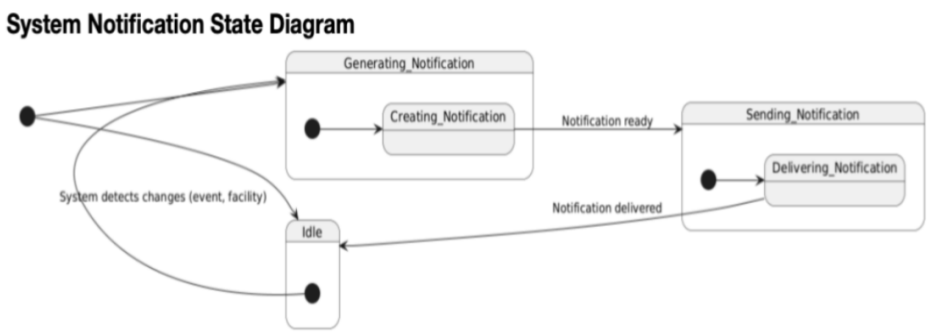


Diagram 3.5 : System Notification State Diagram

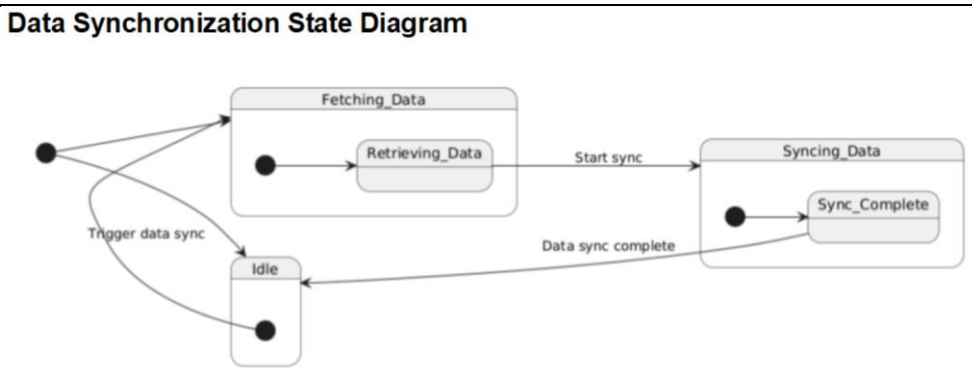


Diagram 3.6 : Data Synchronization State Diagram

3.9.3 Activity Diagram

User Journey - Requesting and Navigating a Route (with Parking and Facilities Availability)

This diagram will reflect the user flow, including parking and facilities availability, which directly affects the route planning.

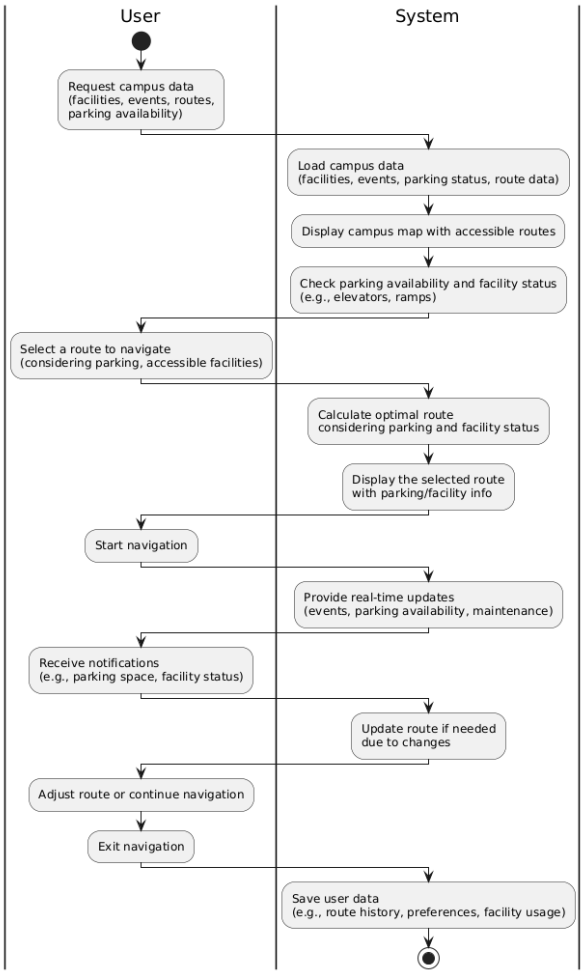


Diagram 3.8 : User journey activity diagram

Admin Workflow - Updating Campus Data (Events, Facilities)

This diagram will describe the administrator's process of updating the campus event and facility data in the system.

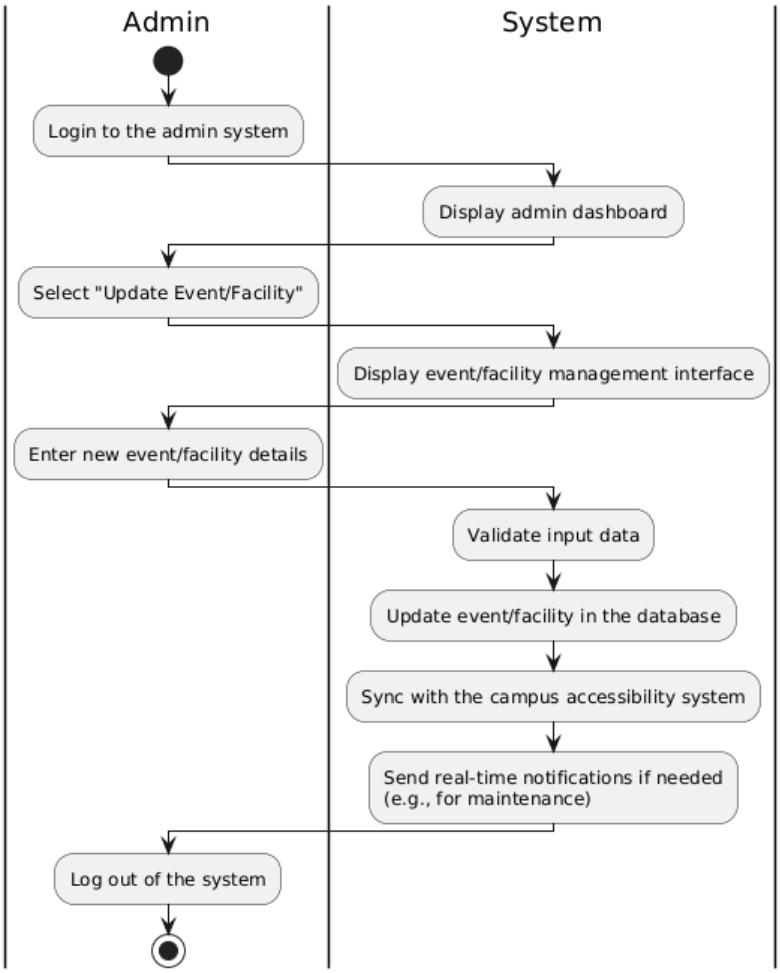


Diagram 3.9 : Admin workflow activity diagram

Real-Time Updates and Notifications

This diagram represents the process of the system sending real-time updates to the users regarding campus events, maintenance updates, etc

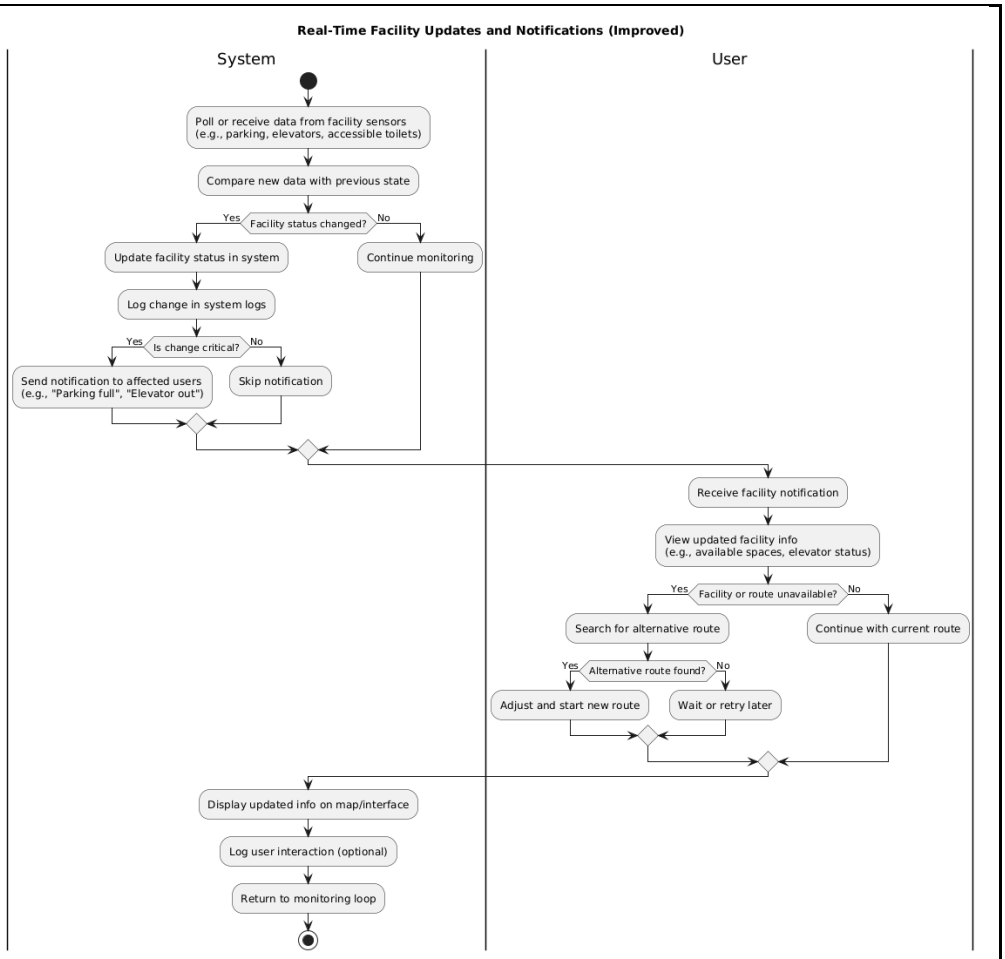


Diagram 3.10 : Real time update and notification activity diagram

User Exiting the Navigation System

This diagram describes what happens when the user exits the navigation system.

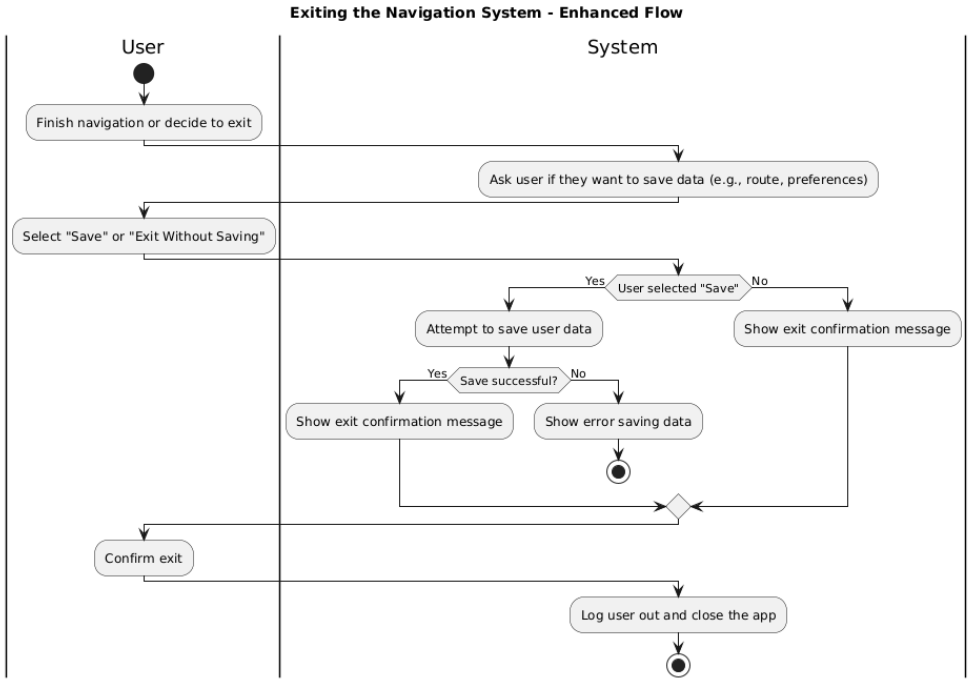
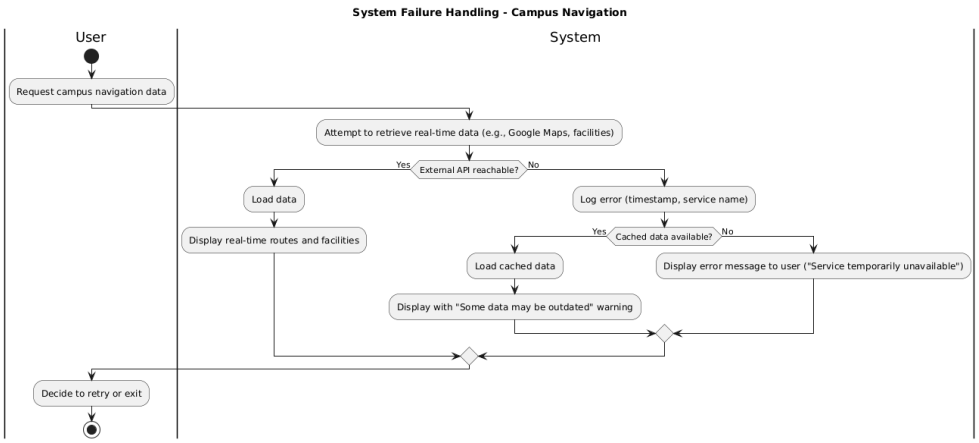


Diagram 3.11 : User exiting the navigation system activity diagram

System Failure/Error Handling



This diagram outlines the process when the system encounters an error, such as failing to load campus data or calculate a route.

Diagram 3.12 : System failure activity diagram

4.Verification

4.1 Stakeholders and Verification Summary

Stakeholders	Verification Method	Verification Summary
Students	User acceptance testing (UAT), surveys	Verified the navigation system improves accessibility and ease of locating campus facilities. Students found the map integration and notification features helpful and intuitive.
MMU staff/admins	Functional testing, interviews	Confirmed system supports effective management of events and facilities. Real-time synchronization with campus databases and notification delivery was accurate and timely.
Software Developers (Project Team)	Unit testing, integration testing, performance testing, security testing	Ensured all modules function correctly and cohesively. Performance testing met the response time (<3s) and availability (≥99.5%) requirements under peak loads. API integrations were stable. Security testing verified encrypted data transmission and compliance with PDPA and GDPR.

4.1.1 Students Verification

Verification Objective: Confirm the system meets usability, accessibility, and functional needs for student users.

Test Case ID	Description	Method	Expected Result	Status
ST-UAT-01	Query real-time facility status	User acceptance testing	Students can view up-to-date availability of campus facilities	Pass/Fail
ST-UAT-02	Navigate to campus locations via map	User acceptance testing	Clicking a location opens correct Google Maps directions	Pass/Fail
ST-UAT-03	Access personal class schedules	User acceptance testing	Students can view their individual schedules correctly	Pass/Fail
ST-UAT-04	Customize and receive push notifications	Survey and Practical Test	Notifications are received accurately and customizable by users	Pass/Fail
ST-UAT-05	Use of accessibility features	Accessibility testing by sample students	UI is accessible (high contrast/screen reader compatibility works)	Pass/Fail

4.1.2 MMU Staff / Admin Verification

Verification Objective: Ensure that administrative staff can effectively manage campus data including events, facilities, and notifications.

Test Case ID	Description	Method	Expected Result	Status
AD-FUNC-01	Manage campus events (create, update, delete)	Functional testing	Admins can successfully manage campus events	Pass/Fail
AD-FUNC-02	Update and sync facility status	Functional testing	Facility updates are reflected accurately in the user interface	Pass/Fail
AD-PERF-01	Verify real-time synchronization with campus databases	Integration testing	Campus event and facility data syncs within 1 minute	Pass/Fail
AD-NOTIF-01	Verify notification triggers for event/facility changes	Functional testing	System sends correct notifications upon updates	Pass/Fail
AD-SEC-01	Admin access control and data security	Security testing	Admin access is restricted and data remains secure	Pass/Fail

4.1.3 Software Developers (Project Team) Verification

Verification Objective: Validate the functionality, integration, performance, and security of all system components.

Test Case ID	Description	Method	Expected Result	Status
DEV-UNIT-01	Test each module independently	Unit testing	Each module passes all functional unit tests	Pass/Fail
DEV-INT-01	Test module/API integration	Integration testing	System components and APIs work together seamlessly	Pass/Fail
DEV-PERF-01	Load test system with 1000 concurrent users	Performance testing	Response times remain <3 seconds; availability ≥99.5%	Pass/Fail
DEV-SEC-01	Test security and encryption measures	Security testing	Data is encrypted and no major vulnerabilities are present	Pass/Fail
DEV-ERR-01	Test error handling and recovery mechanisms	System testing	System recovers gracefully and shows appropriate user-facing messages	Pass/Fail

4.2 Verification Methods and Coverage

Functional Testing

Ensured all features (facility status query, event updates, class schedule access, notifications) operate as intended with no critical defects.

Performance Testing

Confirmed that the system supports high user loads and maintains fast response times during peak use.

Usability Testing

Conducted sessions with students, including users with disabilities, leading to UI refinements based on accessibility feedback.

Security Verification

Ensured compliance with PDPA and GDPR. Validated secure data transmission and restricted access controls.

Interface Testing

Confirmed reliable communication with external systems through RESTful APIs and integration with Google Maps.

Error Handling Testing

Simulated failure scenarios to confirm the system provides clear recovery paths and error messages.

5. Appendices

5.1 Assumptions and dependencies

5.1.1 Assumptions

- Users will have consistent and reliable internet connectivity to access real-time data and notifications.

- Campus staff and students possess basic smartphone literacy.
- Google Maps API and third-party services used for navigation will remain operational throughout the system's lifecycle.
- APIs from campus facilities and event management systems will remain available and function as expected.
- Users will provide consent for data collection and processing, as per data protection regulations (PDPA, GDPR).
- Initial database assumptions were revised after conflict analysis with stakeholders to ensure alignment with actual campus workflows (e.g., event attendance tracking for students/staff).

5.1.2 Dependencies

- The system depends on continuous synchronization with campus facilities and event management systems for up-to-date information.
- Real-time notification delivery relies on mobile platform push notification services (e.g., Android Firebase Cloud Messaging, Apple Push Notification Service).
- Compliance with applicable data privacy laws such as the Personal Data Protection Act (PDPA) and General Data Protection Regulation (GDPR) is mandatory.
- The system’s performance and availability may be affected by the campus network infrastructure and third-party API service availability.

5.2 Acronyms and abbreviations

Acronym	**Meaning**
API	Application Programming Interface
GDPR	General Data Protection Regulation
PDPA	Personal Data Protection Act
WCAG	Web Content Accessibility Guidelines
UAT	User Acceptance Testing
REST	Representational State Transfer
HTTPS	Hypertext Transfer Protocol Secure