A Location-Aware Facility Helpdesk Mobile App with Real-Time Notifications and AI-Driven Analytics

Melvin Yii-Syn Kwan3, a), Lee-Ying Chong1, 3, b), Siew-Chin Chong1, 3, c),   
Check-Yee Law2, 3, d)

1Centre for Advanced Analytics, COE of Artificial Intelligence, Multimedia University, Jalan Ayer Keroh Lama,   
75450 Bukit Beruang, Melaka, Malaysia

2Centre for Intelligent Cloud Computing, COE of Advanced Cloud, Multimedia University, Jalan Ayer Keroh Lama,   
75450 Bukit Beruang, Melaka, Malaysia

3Faculty of Information Science and Technology, Multimedia University, Jalan Ayer Keroh Lama,   
75450 Bukit Beruang, Melaka, Malaysia

*b) Corresponding author: lychong@mmu.edu.my*

*a) 1211104379@student.mmu.edu.my*

*c) chong.siew.chin@mmu.edu.my*

*d) cylaw@mmu.edu.my*

**Abstract.** In today's rapidly evolving educational landscape, efficient facilities management is paramount for sustaining a seamless learning environment. However, many Malaysian universities still rely on outdated web portals for addressing facility-related reports, leading to inefficiencies and suboptimal campus experiences. This paper addresses this gap by proposing a mobile-first facilities helpdesk system designed to modernize issue reporting, tracking, and resolution. By offering an intuitive mobile application, the system aims to empower students and staff to report issues conveniently, fostering greater engagement in campus upkeep. A well-designed real-time notification helps users to understand the progress of the report better by notifying them of each step in handling their report. Additionally, an integrated analytics system makes it easier to spot where and when the most problems happen, thus maintenance teams can perform better data-driven decisions and optimize the campus resources. The project, initially meant for the Faculty of Information Science and Technology building at Multimedia University, aims to bring about positive changes for the campus, letting people react more efficiently and achieve a safer and more effective atmosphere for everyone, making things better for both students and staff.

# INTRODUCTION

Efficient management of facilities is necessary today to maintain a supportive learning place in Malaysian universities. Yet, since reporting issues at the university often involve traditional web services, it slows things down, causes confusion, and results in an unsatisfactory experience for students. To address these issues, a mobile-first support system for facilities is proposed in this paper. It creates a better way for reporting issues with useful features such as push notifications, attaching various types of files, and using geo-location. In using mobile technology, Malaysian universities can bring their facilities management up to date, which is now expected by modern students from other parts of the world. As a result, students and staff can enjoy better life at campus, faster help from authorities, greater community sense, and better operating conditions.

The present methods of reporting any problems found in university facilities in Malaysia are often seen as ineffective and complicated. The difficulty of using application forms and poor design discourages students from speaking up promptly, which may result in small problems getting bigger. In addition, without reliable notifications, students find it difficult to keep up with their report updates, which can make them doubt the facilities management. As a result, people start to use direct emails more, which creates chaos and leads to duplicating work. Not having a centralized data system makes it challenging for facilities management to notice regular problems, see patterns, and improve maintenance strategies. Therefore, better reporting is required to make operations and maintenance more efficient, encourage early reporting, and inform decisions based on data for resources.

There are three major objectives to this project. Firstly, the app provides a user-friendly device for reporting problems with campus facilities, so that the university community can play a bigger role in campus upkeep. Second, adding a solid notification system helps ensure everyone is aware of how their issue is being handled at all times. Also, the project implements an analytics dashboard to show important details about issues and helps the facilities management team use resources more wisely.

The suggested system is designed to transform facilities management by making it easier to find locations and report issues via mobile, leveraging the map component. The system keeps users informed by sending them real-time updates on all the submitted reports. In addition, an AI-based analytics dashboard will assist administrators in understanding the data and acting promptly, assigning resources efficiently, and making choices based on data for the benefit of the campus.

# Literature Review

## Background Study

The helpdesk systems evolution mirrors technological advancement and customer service transformation. Starting with face-to-face interactions, support evolved through telephone assistance in the late 19th century to call centers with Automatic Call Distributors in the 1960s. The 1980s formalized "helpdesk" concepts with basic ticketing systems, while the 1990s introduced Information Technology Infrastructure Library (ITIL) framework and remote support capabilities. Cloud computing in the 2000s enabled scalable solutions and self-service options, with recent AI integration transforming operations through chatbots and omnichannel support. Helpdesk systems now extend beyond IT to facilities management, scheduling staff and assets for maintenance requests.

Ariffin and Daud [1] found 80% of Malaysian public universities adopted centralized customer service for IT and facilities management. Infrastructure and facilities in an educational institute serves as the supporting tools to achieve learning objective [2]. A quantity surveying by Yap et al. [3] revealed that the quality of academic services and facilities was the second largest factor that impact on student experience. Modern university complexes demand increasingly sophisticated helpdesk systems, as stated by Rezaeian et al. [4] through comparative 30-year study. Despite this, most Malaysian universities still rely on functional but limited web portals for helpdesk interfaces, lacking advanced reporting and issue tracking capabilities. These outdated systems fail to provide modern user experiences with real-time updates and intuitive interfaces. Rezaeian et al. [4] note that gaps between user expectations and system capabilities reduce satisfaction and efficiency.

System performance and user satisfaction present opportunities for mobile platform transformation. Heflin et al. [5] suggested mobile technology can reimagine traditional processes with more flexible models. This shift moves from authority-based to community-based structures that increases user engagement [6]. Mobile platforms offer immediate information access with enhanced interaction, provided interfaces are well-designed. Recent studies indicate that mobile apps are becoming an essential part in designing smart campus environments that provide end-users with digital services, augmenting physical resource with smart through digital functionality [7]. Mobile applications make campus services more accessible and convenient for students and staff to engage any time and available anywhere. This shift would not only modernize the helpdesk experience but also align with the increasing preference for mobile-first digital interactions, enhancing both the efficiency of campus facilities management and the overall student experience.

## Existing Systems

The MyJalan KKR mobile application [8], launched by the Ministry of Works Malaysia, serves as a compelling case study in leveraging technology for public service. This initiative provides a user-friendly platform for citizens to report road-related issues directly through their smartphones. The app streamlines the reporting process with intuitive location pinning via GPS, and a simple form for fault category and description. However, it currently lacks push notifications and offers somewhat imprecise location selection.

The FixIT portal [9], developed by Telekom Malaysia, offers a web-based platform for reporting faulty facilities, particularly within educational and corporate settings like Multimedia University. Users can easily report any problems by using QR code or by accessing the app. On the portal, there is a dashboard to check report status, a menu to categorize fault locations, and a list where user can follow submitted reports with their details and the actions taken. Its main advantage is that it is simple to use, and the fast QR code system makes finding locations accurate. Yet, FixIT doesn’t send real-time notifications and responds quite slowly.

The e-Aduan portal at Universiti Teknologi MARA lets students and staff report any service, facility, or IT concerns in an easy-to-access online system [10]. The Facility Management System allows users to formally submit complaints by giving important details like building, block, level, and the location’s area. Fault reporting also requires describing the category, the element, the problem, and writing about it in detail. While the portal is straightforward and offers detailed fault categorization, it lacks real-time notifications and the location selection process can be tedious due to its segmented nature. Additionally, the system does not offer the functionality to upload images for better issue representation.

The UNITAR One Stop Centre Portal [11] is a centralized web platform designed to streamline student interaction with UNITAR International University's services. It offers a straightforward interface with three primary functions: creating a ticket, checking ticket status, and accessing FAQs Students create tickets by choosing a help topic and subtopic, describing their issue like writing an email, and optionally adding attachments. They can track submitted tickets and view updates and replies in an email-style format. While the portal is user-friendly and well-organized, it lacks real-time notifications and does not offer specific help topics for reporting facilities-related issues on campus.

# The proposed solution

The overall structure of the proposed application is shown in Figure 1. The parties involved includes administrators, maintenance workers, students, and staff. Students and staff mainly use the app to report facility issues and check the status of their reports. They can report issues by selecting the location on the map shown in the app. Maintenance workers use the mobile app to update the progress of issues assigned to them. Administrators use the application to manage buildings and facilities, and assign tasks to workers if the system does not auto-assign them. In addition, administrators have access to the analytics dashboard and the AI summary, which provides insights into reported issues within a specific date range. All parties are notified of any updates regarding the issues they reported or were assigned.

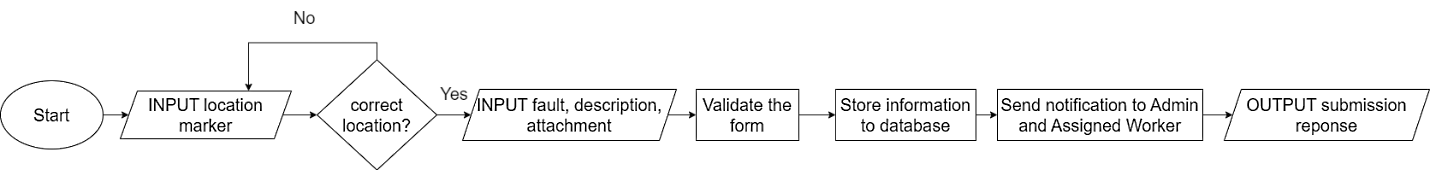
A diagram of a company

AI-generated content may be incorrect.

**Figure 1.** Overview of the proposed solution

## Location-Aware through Map View

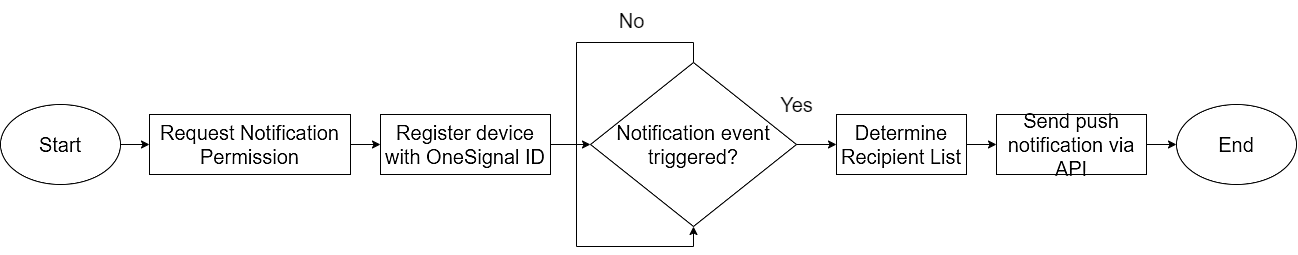
The map functionality in this project is implemented using the “react-native-maps” library [12]. The map provider used is Google Maps, and the initial region is set to the campus’s latitude and longitude. This map component plays a role in two key user workflows: during the submission of a new report as displayed in Figure 2 and when an administrator adds a new building to the system. In both scenarios, the component actively listens for Point of Interest (POI) click events. Upon such an event, it captures and records the precise latitude and longitude of the selected location using React’s “useState” hook before submitting the data to Firestore. Furthermore, the map dynamically renders markers based on building location data retrieved from the application's database. This visual representation empowers users to easily and accurately select the relevant location when submitting a report.



**Figure 2.** Flowchart of submitting a new report

## Real Time Notification

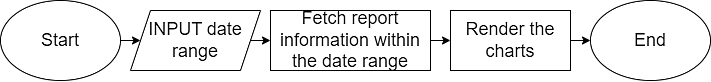
The real-time notification feature uses the “react-native-onesignal” library [13]. A prerequisite for this integration involves linking the project repository (on Firebase) to the OneSignal platform, which generates a OneSignal App ID required for configuration. Figure 3 describes the flow of the working mechanism for real time notification. Within the application’s entry point (App.tsx), a listener is initialized to manage both user permissions for notifications and the handling of incoming push notifications. Upon granting notification permission, a user’s device is registered with the OneSignal platform and receives a unique identifier. The primary API endpoint used for sending notifications is [*https://api.onesignal.com/notifications*](https://api.onesignal.com/notifications). In the proposed application, administrative users are distinguished by the role "admin," while workers and general users are identified through their unique OneSignal registration IDs. When a push notification event is triggered, the application dynamically determines the recipients. For example, when a worker updates the report progress, the application identifies the user who submitted the report and any users tracking its status. This list is then included in the request body when invoking the OneSignal API endpoint.



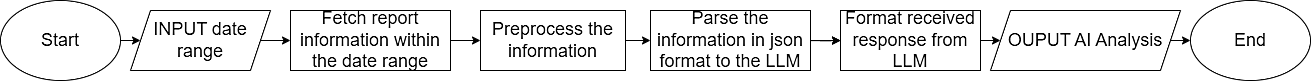
**Figure 3.** Flowchart of push notification event

## AI Driven Analytical Dashboard

The analytical dashboard features charts and AI analysis tabs. The charts tab uses ‘react-native-chart-kit’ to visualize KPIs through line and pie charts, showing trends over time and distributions of report statuses, fault types, and facilities. Figure 4 demonstrates the flow of the working mechanism for analytic dashboard. As for the AI analysis tab, it employs the llama-4-scout-17b-16e-instruct Large Language Model (LLM) via Groq API [14], chosen for its 10M token context window and processing speed. Figure 5 demonstrates the working mechanism for the AI analysis feature. When an administrator selects a date range, the application compiles report data, chart information, and aggregated metrics (total reports, completion rates, average times, common faults/facilities) into a JSON structure. This payload is sent to the LLM with instructions to generate an overview, insights, trends, and recommendations, which are then returned to the application.



**Figure 4.** Flowchart of rendering the analytic dashboard



**Figure 5.** Flowchart of generating AI analysis

# Implementation result

The proposed mobile application has been completed, featuring a location-aware map view, real-time notifications, and an AI analytics dashboard, as shown in Figures 6(a)–6(d). Users (students or staff) report facility issues by pressing the location marker in the map view component, as illustrated in Figure 6(a). Facility options are then displayed based on the selected building, while equipment options are rendered according to the chosen fault type. Once a user submits a report, the details are stored in the Firestore database, triggering the system to send out push notifications. Administrators receive these real-time push notifications as shown in Figure 6(b). The notification title is dynamically generated based on the reported fault type, and the content is tailored to the user’s role. Besides, users can access previous notifications through the application's notification history component.

Analytic dashboard page is an exclusive feature for administrators as shown in Figure 6(c). It displays statistics such as fault type distribution, report over time, report status distribution, and frequent reported facility. This page provides a visualized overview of the reported issues within the selected date range for the administrators. In the analytic dashboard page, quick filters as well as custom date filter are provided to quickly define a certain date range. The chart will re-render based on the date range defined by the administrator. Figure 6(d) displays the AI analysis page where reports within the defined date range are passed to the LLM to be analyzed. This page is also an exclusive feature for administrators. The initial response returned from the LLM is in markdown format and is transformed into readable format. In addition, the analyzed results are saved to the Firestore database and accessible through the history button on the top right corner.

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |
| (a) | (b) | (c) | (d) |

**Figure 6.** Dashboard of the proposed system: (a) Submit report form, (b) Real-time push notifications, (c) Analytic dashboard page, (d) AI analysis page

# Testing

Functional testing is performed to the primary functional modules after the development is completed. By testing the modules, it determines whether the system satisfies the requirements. Table 1 shows the list of test plans for each of the different functional modules [15].

**Table 1.** Test cases for the proposed system

|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **Test Plans** | **Expected Result** | **Result** |
| **Report Related Process** | | | |
| 1 | User submits the report after selecting the location and filling in required details. | Show a success alert message and notify the admin | Passed |
| 2 | A worker updated the details of the assigned report. | Admin, user who reported the issue and users who are tracking the issue should be notify. | Passed |
| 3 | User tries to track more than 5 issues. | Show an error/alert message to tell user to remove some records from current tracking list. | Passed |
| 4 | Admin assigns the task to worker if the auto assign task function did not find available worker. | The selected worker should receive a notification and well as well the user who reported the issue and user who are tracking the issue | Passed |
| 5 | Visibility of report details based on user role | User and worker should only see report that they submitted or assigned respectively. Admin should be able to review all reports submitted. | Passed |
| **Building and Facility Related Process** | | | |
| 6 | Admin insert the same building/facility | Show an alert message to inform the admin that the same building/facility is already added. | Passed |
| 7 | Admin edits the building/facility detail. | The changes should be recorded, and the page should be updated. | Passed |
| 8 | Admin did not fill in all the required fields in the create court form. | Show an error/alert message to ask for admin input again. | Passed |
| **Analytic Dashboard Related Process** | | | |
| 9 | Admin selects a date period for the analytics dashboard | The charts should be rendered based on the report information within the selected date range | Passed |
| 10 | Admin select a date period to generate the AI summary | The AI summary should only conclude the report that is reported within the selected date range | Passed |
| 11 | Admin did not select a date period | Show an alert message to inform the admin to select a data range. | Passed |
| 12 | Admin select a time frame to generate report | The report of the specific timeframe is showed to the admin. | Passed |

# USABILITY TESTING

A survey was conducted to evaluate the usability of the proposed system, receiving a total of 30 responses from students at Multimedia University. The majority of respondents were from the Melaka campus, with a smaller number from the Cyberjaya campus. The participants included 26 undergraduate students, 2 postgraduate students, 1 diploma student, and 1 foundation student. The purpose of the survey was to gather students' opinions on the proposed system and assess their willingness to adopt it if implemented on campus (see Figure 7).

|  |  |
| --- | --- |
| A screenshot of a graph  AI-generated content may be incorrect. | A screenshot of a graph  AI-generated content may be incorrect. |
| **(a)** | **(b)** |

**FIGURE 7.** Majority of students agreed: (a) The map feature simplifies form submission, and (b) The analytics dashboard and AI analysis improve FMD decision-making

From Figure 7(a), 28 out of 30 respondents agreed that it is easier now to select the location when filling in the submission form for reporting facility issues. This is thanks to the map feature which makes pinpointing location more intuitive than the traditional form filling approach. As shown in Figure 7(b), 22 out of 30 respondents agreed that the analytic dashboard and AI analysis feature could help to provide useful insights to Facility Management Department (FMD) in improving the facility management. The other 8 respondents reflected that although the dashboard visualizes the fundamentals insights, it does not provide any patterns like the AI analysis feature. Some of them recommended that instead of just displaying the numbers, predictive features can be added to the charts. While some also recommend integrating this feature to the user portal instead of making it an exclusive feature to the administrator.

From Figure 8, majority of the respondents are willing to use the proposed system and agreed that it can help the FMD to manage facility issues better. The implemented map component provides a straightforward approach to pinpoint the location instead of going through series of input field when selecting the location. The availability of the analytics dashboard and ability to provide quick visualization and analysis helps FMD to better understand the pattern of the reported facility issues.

|  |  |
| --- | --- |
|  |  |
| **(a)** | **(b)** |

**FIGURE 8.** Majority of students: (a) Use the proposed system to report facility issues on campus, and (b) Agree that the proposed system helps the campus manage facility issues more efficiently

# Conclusion

This paper proposed a facility helpdesk mobile application which addressed previous problems on existing systems among the local universities. The system introduces several key features to streamline and enhance the efficiency of the operation, as well as encouraging data-driven decision making. Compared to the existing systems studied in this paper, the proposed system keeps the important features (e.g. map view, push notification) while offering unique features such as the dedicated analytic dashboard and AI analysis feature. This helps to facilitate proactive maintenance strategies and optimized resource allocation. Ultimately, the successful deployment of this mobile facility helpdesk system promises to cultivate a more responsive, efficient, and well-maintained campus environment, leading to increased satisfaction among students and staff and contributing to an improved overall university experience.

# References

1. N.M. Ariffin, and N.M. Daud, “Customer Centered Service in Malaysia: A Case Study in Malaysian Public Universities,” MJoC **4**(1), 214 (2019).
2. I. Ilmi, M. Erihadiana, and S. Wanayati, “Facilities and Infrastructure Management,” Edu **6**(3), (2023).
3. J.B.H. Yap, Q.L.T. Hew, and M. Skitmore, “Student Learning Experiences in Higher Education: Investigating a Quantity Surveying Programme in Malaysia,” CEB **22**(1), (2022).
4. M. Rezaeian, and M. Wynn, The Evolution of the Customer-Centric Helpdesk: Two Case Studies (2018).
5. H. Heflin, J. Shewmaker, and J. Nguyen, “Impact of mobile technology on student attitudes, engagement, and learning,” Computers & Education **107**, 91–99 (2017).
6. S. Hamm, G. Saltsman, B. Jones, S. Baldridge, and S. Perkins, “A Mobile Pedagogy Approach for Transforming Learners and Faculty,” in Handbook of Mobile Learning, (Routledge, 2014).
7. T. Lumor, M. Pulkkinen, and A. Hirvonen, “The Actual Adoption and Use of Mobile Apps: The Case of a Higher Education Context,” in American Conference on Information Systems (AMCIS) 2020 Proceedings (2021).
8. Government of Malaysia, “MyJalan KKR,” Google.com, Available at: https://play.google.com/store/apps/details?id=my.gov.onegovappstore.mypotholes, (2021).
9. Telekom Malaysia, “FixIT,” Tm.com.my, Available at: https://www.sub.tm.com.my/fixit/index.cfm?path=pwa (2025).
10. Universiti Teknologi Mara, “:: eAduan ::”, Uitm.edu.my, Available at: https://fms.uitm.edu.my/eAduan/, (2025).
11. UNITAR International University, “UNITAR One Stop Centre Portal,” Unitar.my, Available at: https://osc.unitar.my/index.php (2025).
12. npm, “react-native-maps,” Npm, Available at https://www.npmjs.com/package/react-native-maps, (2025).
13. npm, “react-native-onesignal,” Npm, Available at https://www.npmjs.com/package/react-native-onesignal, (2025).
14. groq, “GroqCloud - Build Fast,” Groq.com, Available at https://console.groq.com/docs/model/llama-4-scout-17b-16e-instruct, (2024).
15. Z.Y. Chia, and P.Y. Goh, “Real Time 3D Internal Building Directory Map,” Journal of Informatics and Web Engineering **3**(2), 37–56 (2024).