SOFTWARE REQUIREMENT SPECIFICATION

SmartClass: AI-Integrated Learning and Classroom Control

By : MUHAMMAD AFIQ FAHMIE BIN AMRI

Matric No. : S67158

Supervised by: DR. MOHAMAD NOR BIN HASSAN

CSF4984 FINAL YEAR PROJECT I

FACULTY OF COMPUTER SCIENCE AND MATHEMATICS
UNIVERSITI MALAYSIA TERENGGANU

TABLE OF CONTENTS

CHAPTER 1

INTRODUCTION

The **SmartClass: AI-Integrated Learning and Classroom Control** project represents a cutting-edge approach to addressing challenges in traditional classroom environments. Leveraging the capabilities of Artificial Intelligence (AI) APIs and Internet of Things (IoT) technologies, SmartClass is designed to enhance the classroom experience for both educators and students. Developed for *Pejabat Pembangunan dan Harta (PPH)* at Universiti Malaysia Terengganu (UMT), the system aims to tackle persistent issues such as inefficient note-taking Salame et al. (2024), utility management, and attendance tracking.

Traditional classrooms often face limitations that reduces the teaching and learning efficiency. For instance, manual note-taking can be challenging for students, while managing classroom utilities and ensuring efficient attendance tracking can be time-consuming for lecturers. The SmartClass system addresses these issues by integrating solutions follows to modern classroom needs.

The system incorporates multiple functionalities to transform classroom management:

- **AI-Powered Lecture Summarization:** The system automates note-taking by transcribing and summarizing the lecture into more simpler form . Students can access organized summaries conveniently through a mobile application, ensuring they focus on learning rather than manually recording information traditionally.
- **IoT-Driven Utility Management:** Lecturers can use voice-activated controls to seamlessly adjust classroom utilities, such as lighting, to suit different teaching scenarios, including “Lecture Mode” and “Video Mode.” This simplifies the management of the learning environment.
- **RFID Attendance with Motion Detection:** Attendance tracking is automated through RFID card scanning, enhanced by motion detection to verify that only one individual is present during the scan. This feature ensures the accuracy of attendance records and prevents miss use.

- **Remote Utility Monitoring and Control:** PPH technicians can remotely manage classroom utilities, enabling them to monitor the status of devices and turn off unused equipment. This promotes energy efficiency and reduces operational costs.

This SRS document serves as a roadmap for the design and implementation of the SmartClass system. It outlines the functional and non-functional requirements, ensuring clarity and alignment among stakeholders, and academic evaluators. Additionally, it provides a framework for understanding the system's capabilities, limitations, and potential impact on educational practices.

The following subsections expand on the purpose, scope, and definitions critical to the development of SmartClass, ensuring that all stakeholders have a comprehensive understanding of the project's goals and features.

1.1 Purpose

The purpose of this Software Requirements Specification (SRS) is to define the requirements for the **SmartClass: AI-Integrated Learning and Classroom Control** system. This document provides a detailed description of the system's functionalities, performance criteria, and design constraints to ensure clarity and alignment among all stakeholders. By serving as a guiding reference throughout the development process, the SRS ensures that the final system meets its intended objectives and delivers value to its users.

The SmartClass system is being developed for *Pejabat Pembangunan dan Harta (PPH)*. It aims to address the challenges of traditional classroom environments, such as manual note-taking, inefficient utility management, and traditional way of attendance tracking. By leveraging Artificial Intelligence (AI) APIs and Internet of Things (IoT) technologies, SmartClass seeks to enhance classroom management and improve the learning experience for both educators and students.

This document is intended for the following audience:

- **System Developers:** To use the requirements outlined in this SRS as a blueprint for system design, development, and testing.
- **Project Stakeholders:** To validate that the system aligns with the objectives and expectations of UMT and *Pejabat Pembangunan dan Harta (PPH)*.
- **Academic Evaluators:** To assess the feasibility, scope, and deliverables of the project based on the defined requirements.

In summary, the SRS defines what the SmartClass system must achieve, ensuring that the development process remains focused and aligned with the project's objectives. By providing a clear understanding of the system's purpose, this document minimizes misunderstandings and supports successful implementation.

1.2 Scope

The **SmartClass: AI-Integrated Learning and Classroom Control** system is a comprehensive solution developed to improve classroom management and enhance the overall learning experience. By utilizing Artificial Intelligence (AI) APIs and Internet of Things (IoT) technologies, this system addresses the key challenges faced in traditional classroom settings, such as inefficient note-taking, resource management, and attendance tracking. The primary target audience for this project is *Pejabat Pembangunan dan Harta (PPH)* at Universiti Malaysia Terengganu (UMT).

1.2.1 System Overview

The SmartClass system provides the following core functionalities:

1. **AI-Powered Lecture Summarization:** Generate lecture summarization by transcribing and also summarizing lecture content into more simpler form. Summaries are accessible to students through a dedicated mobile application, allowing them to focus on learning instead of manual recording.

2. **IoT-Based Utility Control:** Enables voice-activated control of classroom utilities, such as lighting and multimedia equipment. This feature supports various teaching modes, including "Lecture Mode" and "Video Mode," for a seamless classroom experience.

3. **RFID Attendance Tracking with Motion Detection:** Provides automated attendance tracking through RFID card scanning. The system ensures accuracy by using motion detection to verify that only one person is present during the scan, reducing fraudulent practices.

4. **Remote Utility Monitoring and Control:** Offers PPH technicians the ability to monitor and control classroom utilities remotely. This feature promotes energy efficiency by allowing unused devices to be turned off or monitored from a distance.

1.2.2 Project Objectives

The primary objectives of the SmartClass system are as follows:

- To enhance classroom efficiency by automating routine administrative tasks.
- To improve the learning experience for students through AI-powered note-taking and utility controls.
- To support sustainable energy usage by providing remote control capabilities for utilities.
- To simplify classroom management for both lecturers and technicians.

1.2.3 Application and Benefits

The SmartClass system is designed as a prototype for Pejabat Pembangunan dan Harta (PPH), with potential for scalability to the real classes in Universiti Malaysia Terengganu in the future. Key benefits of the system include:

- Reducing the workload of lecturers and students through automation.
- Increasing energy efficiency by optimizing the usage of classroom utilities.
- Offering an user-friendly platform for classroom management.

1.2.4 Limitations and Exclusions

While the SmartClass system introduces several advanced features, its scope is limited to specific functionalities related to classroom management:

- The system does not replace the role of lecturers in teaching and decision-making.
- Administrative tasks unrelated to classroom activities, such as payroll or maintenance, are excluded.
- Full implementation beyond the prototype phase will require additional scaling, budget and development efforts.

1.2.5 Future Scalability

Although the current focus is on implementing a functional prototype for UMT, the SmartClass system is designed with scalability in mind. The modular architecture allows for future enhancements and adaptations to meet the needs of UMT's in the future.

1.3 Definitions, Acronyms, and Abbreviations

This section provides a comprehensive list of definitions, acronyms, and abbreviations used throughout this Software Requirements Specification (SRS) document. By clearly defining these terms, this section ensures a shared understanding of technical concepts, system components, and organizational entities mentioned in the project. It serves as a reference for all readers, especially those unfamiliar with specific terminologies or abbreviations related to Artificial Intelligence (AI), Internet of Things (IoT), or project management practices.

The table below presents the key terms and their meanings, offering clarity and promoting consistency throughout the document. This ensures that stakeholders, developers, and evaluators can interpret the document accurately without ambiguity.

Acronym/Term	**Definition**
AI	Artificial Intelligence Stryker and Kavlakoglu (2024) : Technology simulating human intelligence for tasks such as decision-making and speech recognition .
IoT	Internet of Things khan and Rashid (2024): Network of interconnected devices that communicate and exchange data without human intervention.
RFID	Radio Frequency Identification Wikipedia Contributors (2019): A technology for automated identification and tracking using electromagnetic fields.
PPH	Pusat Pembangunan dan Harta: Development and Property Office at Universiti Malaysia Terengganu (UMT).
SRS	Software Requirements Specification: A document that specifies the system's functional and non-functional requirements.
UMT	Universiti Malaysia Terengganu: An academic institution where the SmartClass project is being implemented.
ESP32	A microcontroller widely used in IoT applications with built-in Wi-Fi and Bluetooth capabilities Hercog et al. (2023).

Table 1.1: Definitions, Acronyms, and Abbreviations

1.4 Overview

This Software Requirements Specification (SRS) document is designed to provide a comprehensive understanding of the SmartClass system. It includes details on the purpose, scope, and functionalities of the system, ensuring alignment among all stakeholders. The document serves as a blueprint for developers, stakeholders, and evaluators to guide the development and assessment process.

The SRS is structured into the following chapters:

- **Chapter 1 - Introduction:** Provides an overview of the project, including its purpose, scope, objectives, system overview, and scalability. This chapter also introduces definitions, acronyms, and abbreviations used throughout the document.
- **Chapter 2 - Requirement Elicitation Techniques:** Discusses the sources and techniques used to gather requirements, including stakeholder input, document analysis, and observational methods.
- **Chapter 3 - System Requirements:** Details the functional and non-functional requirements of the system, organized by user roles such as students, coordinators, and supervisors.
- **Chapter 4 - Requirement Analysis:** Includes diagrams and models that describe the system architecture and behavior, such as use case diagrams, activity diagrams, class diagrams, and a CRUD matrix.
- **Chapter 5 - Summary:** Concludes the document by summarizing the key points and providing an overall perspective of the system's design and implementation.

This organization ensures a logical flow of information, making it easy for readers to navigate the document and understand the project's goals and requirements. Each chapter builds upon the previous one, providing a structured and detailed view of the SmartClass system.

CHAPTER 2

REQUIREMENT ELICITATION TECHNIQUES

Requirement elicitation is the process of gathering and understanding the necessary information to define the functionalities and constraints of a system. It plays a critical role in ensuring the success of a project by establishing a clear understanding of what the system must achieve and the conditions under which it will operate. This process typically involves interacting with stakeholders, analyzing existing systems, and reviewing relevant documentation to capture the project's needs comprehensively.

For the SmartClass project, requirement elicitation is especially important as the system aims to address specific challenges in traditional classroom settings. These challenges include inefficient note-taking Salame et al. (2024) , manual utility management, and inaccurate attendance tracking. By integrating Artificial Intelligence (AI) and Internet of Things (IoT) technologies, SmartClass provides innovative solutions such as automated lecture summarization, voice-activated utility control, and RFID-based attendance tracking. However, to design a system that effectively meets the needs of its users, it is crucial to thoroughly understand the perspectives and requirements of all involved parties, including students, lecturers, and technicians at Pejabat Pembangunan dan Harta (PPH) at UMT.

The goal of the requirement elicitation phase for SmartClass is to identify the specific business needs, potential risks, and assumptions associated with the project. This ensures that the final system not only addresses the current limitations of classroom environments but also aligns with the technological capabilities and goals of its stakeholders. By carefully analyzing these requirements, the development team can create a system that enhances the learning experience, promotes energy efficiency, and simplifies classroom management tasks.

2.1 Requirement Sources

Understanding the sources of requirements is a critical step in the requirement elicitation process. These sources ensure that all necessary inputs are considered, reducing the risk of missing important details. The requirements for the SmartClass system are derived from three primary sources: stakeholders, documents, and existing systems. Each source provides unique insights that guide the system's design and ensure it meets its intended purpose.

2.1.1 Stakeholders

Stakeholders are individuals or groups who have an interest in the system or are directly affected by its functionality. Engaging with stakeholders is vital as they provide firsthand insights into their needs, expectations, and pain points. For the SmartClass system, stakeholders play different roles based on their involvement with the project:

- **PPH Staff:** These stakeholders are responsible for maintaining and managing classroom utilities. Their feedback ensures that the IoT-based utility control features align with operational requirements, such as monitoring class utilities remotely or turning off unused equipment to save energy.

- **Lecturers:** Lecturers are also the users of the SmartClass system's voice-activated utility controls and lecture summarization features. Their feedback helps define how the system can simplify classroom management during teaching sessions. For example, lecturers may require seamless transitions between ”Lecture Mode” and ”Video Mode” without interrupting the class flow. Understanding their preferences ensures the system enhances teaching efficiency.

- **Students:** Students are the end-users of the lecture summarization and attendance tracking features. Their input focuses on improving the learning experience by ensuring access to accurate. Additionally, they may highlight usability concerns, such as mobile app accessibility, which are crucial for enhancing their interaction with the system.

Table 2.1: List of Stakeholders for the SmartClass System

Name	**Role**	**Responsibilities**	**Feedback Areas**
Mr. Mohd Riezuan Razali	PPH Staff	Maintain and manage classroom utilities.	Monitoring utility status, optimizing remote controls.
Dr. Nurul Shahida	Lecturer	Utilize voice-activated controls and teaching features.	Smooth transition between classroom modes.
Muhammad Farid Izzuddin Bakhori	Student	Access summaries, Clock in attendance; report app issues.	Usability, accessibility, learning enhancements.

Engaging stakeholders in interviews directly with them ensures their perspectives are captured comprehensively, contributing to a well-rounded system design.

2.1.2 Documents

Documents are a valuable source of information during the requirement elicitation phase. They provide technical insights, reference existing standards, and help ensure the system is designed with a strong foundation. Details of the reviewed documents are provided in Appendix A Section. For the SmartClass project, the following documents were utilized to shape the requirements:

- **Technical Manuals:** Documentation for IoT hardware components, such as the ESP32 microcontroller and RFID systems, was reviewed to understand their capabilities and limitations. This helped determine how these components could be integrated into the system effectively.
- **Research or Journal Papers:** Studies related to RFID attendance systems were analyzed. These papers offered insights into best practices and innovative solutions for addressing challenges in traditional classroom environments.
- **Software Documentation:** Documentation for APIs and tools, such as Google's Speech-to-Text API and Gemini AI API, provided details on integrating AI functionalities like lecture transcription and summarization into the system.

These documents ensured that the system's requirements were grounded in proven methodologies and supported by reliable technical resources. By leveraging these materials, the SmartClass project was able to build a solid technical framework and align its features in AI and IoT.

2.1.3 Existing Systems

Studying existing systems provides inspiration and benchmarks for developing the SmartClass system. By understanding the strengths and weaknesses of similar systems, the project can identify areas for improvement and innovation.Details of the reviewed existing system are provided in Appendix A Section. The key system analyzed is:

- **RFID Attendance Systems:** Attendance solutions implemented at other universities, such as Universiti Teknologi Malaysia (UTM) Chan et al. (2017), provided valuable insights into how RFID technology streamlines attendance tracking. These systems demonstrate how RFID cards can ensure accurate, contactless attendance recording, reducing manual errors and time consumption. Analyzing these systems revealed opportunities for enhancement, such as integrating motion detection or pairing RFID systems with mobile apps to provide real-time attendance feedback to students and lecturers.

By evaluating this system, the SmartClass project ensures it delivers an innovative solution that builds on existing technologies while addressing gaps in current approaches.

2.1.4 Summary of Requirement Sources

The requirement sources for the SmartClass system play a vital role in ensuring the project is well-informed, comprehensive, and aligned with its objectives. Each source contributes unique insights that collectively form the foundation for designing a robust and user-centric system.

Stakeholders are a primary source of requirements, providing direct input based on their roles and interactions with the system. Their feedback highlights specific needs and challenges related to attendance tracking, such as the desire for an accurate, automated system that minimizes effort and prevents misuse. Involving stakeholders such as lecturers, students, and PPH technicians ensures the SmartClass project addresses these real-world problems while improving user experience.

Documents serve as a secondary but equally essential source of information. They provide the technical and theoretical background needed to design and implement the system. For example, technical manuals for RFID readers and tags offer detailed specifications guiding hardware integration. Research papers on RFID implementation in education inform the development of features like seamless attendance logging and enhanced accuracy using motion sensors or additional validation methods. Software documentation related to potential mobile app integrations ensures smooth real-time functionality.

Existing Systems offer practical insights by showcasing how similar solutions have been implemented in other contexts. These systems act as benchmarks, allowing the SmartClass project to identify best practices and potential pitfalls. For instance, studying RFID attendance systems implemented at other universities reveals the importance of usability, such as rapid scanning during entry, and the need to mitigate risks, such as unauthorized attendance logging.

In summary, the combination of stakeholder input, document analysis, and lessons from existing systems creates a well-rounded understanding of the requirements. Stakeholders provide context and practical needs, documents supply technical knowledge and theoretical insights, and existing systems demonstrate proven applications. Together, these sources ensure that the SmartClass system is efficient, user-friendly, and capable of addressing challenges associated with attendance tracking in traditional classroom environments. This comprehensive approach to requirement elicitation lays a strong foundation for the system's development, minimizing the risk of missing critical features or overlooking user expectations.

2.2 Requirement Techniques

For the SmartClass project, requirement gathering was a critical phase aimed at understanding the needs and expectations of all stakeholders involved. This process focused on techniques that emphasized direct communication and collaboration, ensuring that the requirements were not only well-defined but also aligned with the practical challenges and goals of the system. By engaging stakeholders such as lecturers, students, and PPH technicians, the project team was able to collect valuable insights into the desired functionalities, system constraints, and overall user experience.

The primary methods used for requirement gathering were conversational techniques, which allowed for meaningful interactions with stakeholders to identify and clarify their needs. These methods ensured that the system addressed real-world challenges, such as improving classroom utility management, automating attendance tracking, and providing efficient lecture summarization. In addition, support techniques were employed to refine the system through iterative feedback and collaborative sessions. These techniques focused on using visual aids like prototypes and follow-up discussions to ensure that the stakeholders' expectations were fully understood and implemented.

By combining conversational and support techniques, the SmartClass project ensured a well-rounded and practical approach to requirement gathering. This strategy not only strengthened communication between the development team and stakeholders but also facilitated the creation of a system that effectively meets the needs of its users while maintaining flexibility for future enhancements.

2.2.1 Conversational Techniques

Conversational techniques were the main approach for gathering requirements, as they allowed direct interaction with key stakeholders. These methods provided valuable insights into the specific needs, expectations, and constraints of the system. The following conversational techniques were applied:

- **Interviews:** Meetings were conducted with lecturers and PPH technicians to gather detailed requirements. Lecturers shared their needs for lecture summarization, utility management, and attendance tracking features. PPH technicians provided input on remote utility control and energy efficiency requirements. These interviews helped clarify functional and non-functional requirements while highlighting constraints. .

- **Feedback Sessions:** After creating initial user interface (UI) prototypes, feedback sessions were held with lecturers and students. These sessions allowed stakeholders to evaluate the design and provide suggestions for improvement. For example, lecturers requested simplified controls for transitioning between ”Lecture Mode” and ”Video Mode,” while students provided feedback on the accessibility of summarized lecture notes. Details of the reviewed existing system are provided in Appendix B and Appendix C Section.

By using conversational techniques, the project team was able to gain a deeper understanding of the system’s goals and ensure the requirements addressed real-world challenges.

2.2.2 Support Techniques

Support techniques were used to facilitate collaboration and ensure that the gathered requirements were properly understood and implemented. These techniques emphasized active stakeholder involvement throughout the development process:

- **Prototypes and Mockups:** The team developed prototypes of the SmartClass user interface and shared them with stakeholders. These mockups provided a visual representation of the system, making it easier for stakeholders to understand its functionality. Feedback gathered during these sessions was incorporated into subsequent iterations of the design.

- **Follow-Up Sessions:** Regular follow-up meetings were conducted with stakeholders to clarify any ambiguities in the requirements and validate that the evolving design met their expectations. This iterative process ensured continuous alignment with stakeholder needs.

These support techniques helped bridge the gap between stakeholders and the development team, fostering collaboration and minimizing misunderstandings during the requirement gathering phase.

2.2.3 Summary of Requirement Techniques

The requirement gathering process for the SmartClass project focused on methods that prioritized direct engagement with stakeholders and iterative feedback. Conversational techniques, such as interviews and feedback sessions, allowed the development team to gain valuable insights into the needs and expectations of the primary users—lecturers, students, and PPH technicians. These interactions provided a clear understanding of the challenges faced in traditional classrooms, such as inefficient note-taking, manual utility management, and inaccurate attendance tracking. Stakeholder input was instrumental in defining the system's core functionalities and aligning them with practical use cases.

Support techniques further strengthened the requirement elicitation process by creating opportunities for collaboration and visualization. The use of prototypes and mockups enabled stakeholders to see and interact with preliminary designs, offering constructive feedback that shaped the final system. Follow-up sessions ensured that ambiguities in the requirements were resolved promptly and that the evolving system continued to meet user expectations. This iterative process also allowed stakeholders to feel actively involved in the development, fostering a sense of ownership and confidence in the project.

The combination of these techniques ensured that the SmartClass system was designed to address real-world problems while maintaining a user-centric approach. By focusing on practical and collaborative methods, the project team was able to create a robust framework for gathering, validating, and refining requirements. This comprehensive approach minimized the risk of overlooked features or misaligned objectives, resulting in a system that effectively enhances classroom management, learning efficiency, and utility control. The requirement elicitation techniques used in this project not only ensured the system's feasibility but also laid the groundwork for its successful implementation and future scalability.

CHAPTER 3

SYSTEM REQUIREMENTS

3.1 Functional Requirement

Functional requirements are essential features that the SmartClass system must implement to help users achieve their goals. These requirements define the specific behaviors and capabilities of the system under various conditions, ensuring that it delivers value to its intended users. By addressing the functional needs of different user roles, these requirements serve as the foundation for system design and development, guiding the implementation of features that meet both business objectives and user expectations.

For the SmartClass project, functional requirements are organized based on user roles such as students, lecturers, and PPH technicians. Each role has unique needs that the system must address to enhance classroom operations and improve the overall learning experience. Students require seamless attendance tracking and access to summarized lecture notes, lecturers need tools to manage classroom utilities and create engaging sessions, and PPH technicians rely on features that enable efficient utility monitoring and maintenance.

Each requirement is crafted to be necessary, concise, attainable, and verifiable, ensuring that it can be realistically implemented within the project's scope. This structured approach minimizes ambiguity, supports consistency, and ensures that the development process remains focused on delivering practical, user-centric solutions. Below are the detailed functional requirements for each user role.

3.1.1 Student

Students are one of the primary users of the SmartClass system, and their interactions with the system focus on features that improve their classroom experience and academic management. These include attendance tracking, accessing attendance history, and receiving notifications related to their attendance status. Additionally, students play a crucial role in reporting issues with classroom utilities, ensuring the system operates efficiently.

By addressing these needs, the SmartClass system aims to reduce the administrative burden on students, allowing them to focus more on learning while ensuring they can access relevant information conveniently. The requirements for students are designed to be simple, intuitive, and easily accessible, reflecting their frequent and essential interactions with the system.The following requirements address their needs:

- **R1:** The system shall allow students to log their attendance using an RFID card by tapping it on a reader at the classroom entrance.
- **R2:** The system shall enable students to view their attendance records through a mobile apps.
- **R3:** The system shall notify students of their attendance status (e.g., ”Present” or ”Absent”) at the end of each class session.
- **R4:** The system shall allow students to report malfunctioning utilities through the app to alert PPH technicians.

These requirements ensure that students can easily manage their attendance and contribute to utility maintenance by reporting issues.

3.1.2 Lecturer

Lecturers are central to classroom operations and require tools that simplify their teaching activities and allow them to manage resources efficiently. The SmartClass system provides functionalities such as reviewing attendance, managing utilities via voice commands, and leveraging AI-powered tools for lecture transcription and summarization.

These features aim to improve the overall classroom experience for both lecturers and students. For example, voice-activated controls reduce interruptions during lectures, and transcription services ensure that critical lecture content is recorded and summarized effectively. The system’s requirements for lecturers are tailored to address their need for convenience, efficiency, and enhanced teaching tools. The following requirements cater to their role:

- **R5:** The system shall enable lecturers to review and verify student attendance records for each class session.
- **R6:** The system shall allow lecturers to control classroom utilities, such as turning lights on or off, using predefined voice commands.
- **R7:** The system shall provide transcription of lectures, converting spoken words into text for documentation.

- **R8:** The system shall automatically generate summaries of lecture content in both English and Malay. Lecturers must review and approve these summaries before they are accessible to students.

These requirements ensure that lecturers can efficiently manage the classroom environment and provide students with useful lecture materials.

3.1.3 PPH Staff

PPH staff are responsible for the maintenance and management of classroom utilities. Their role involves monitoring utility usage, responding to malfunction reports, and ensuring that classroom resources operate efficiently. The SmartClass system supports these tasks by providing a utility monitoring dashboard, alert notifications for malfunctions, and detailed reports on utility usage.

These features are designed to optimize resource management and enable proactive maintenance, ensuring the sustainability and operational efficiency of classroom infrastructure. The requirements for PPH staff reflect their critical role in the system's overall functionality and aim to reduce their workload while improving response times for utility issues. The following requirements support their tasks:

- **R9:** The system shall provide a dashboard for PPH staff to monitor and control classroom utilities remotely.
- **R10:** The system shall alert PPH technicians whenever a malfunctioning utility is reported by a student or lecturer.

These requirements ensure that PPH technicians can manage utilities efficiently, respond promptly to issues, and optimize energy usage.

3.1.4 Summary of Functional Requirements

The functional requirements outlined above address the needs of SmartClass users by providing features tailored to their roles. Students can easily manage their attendance and report utility issues, lecturers can streamline classroom management and deliver enhanced learning experiences, and PPH technicians can maintain operational efficiency with improved utility management tools. Each requirement is designed to support the system's overall goals of enhancing classroom management, learning efficiency, and resource optimization.

3.2 Non-functional Requirements

Non-functional requirements (NFRs) describe how the system operates and defines its overall quality. While functional requirements focus on what the system should do, NFRs focus on attributes like speed, reliability, and usability that ensure the system works well under different conditions. These requirements are important because they improve the system's overall user experience and determine how efficiently it performs in real-world scenarios.

For the SmartClass project, NFRs are carefully designed to meet the needs of students, lecturers, and PPH technicians while ensuring long-term scalability and reliability. These requirements address key areas like performance, compatibility, usability, and security, which are critical for the success of the system. By implementing these NFRs, the SmartClass system is set to deliver an optimized, secure, and user-friendly experience that meets stakeholder expectations effectively.

3.2.1 Performance and Scalability

- **NFR1:** The system should process attendance within a few seconds after a student taps their RFID card.
- **NFR2:** The system should handle up to 50 users at the same time without slowing down.
- **NFR3:** The system should be able to expand to manage more classrooms and users in the future.

3.2.2 Portability and Compatibility

- **NFR4:** The mobile app should work on Android devices.

3.2.3 Reliability and Availability

- **NFR5:** The system should work almost all the time, with very little downtime.
- **NFR6:** Any serious problems, like a system crash, should be fixed within a short time to avoid disruptions.
- **NFR7:** Data like attendance and utility reports should be saved every day to avoid losing information.

3.2.4 Security

- **NFR8:** The system should keep user information safe while sending or storing it.

- **NFR9:** The system should track attempts to access it and notify an administrator if someone without permission tries to log in.

3.2.5 Usability and Localization

- **NFR10:** The system's design should be simple and easy to use, letting users complete tasks quickly.
- **NFR11:** The system should support both English and Malay to make it usable for everyone at UMT.

3.2.6 Maintainability

- **NFR12:** The system should be easy to update or add new features without affecting existing parts.

Non-functional requirements ensure that the SmartClass system operates effectively and meets quality standards that go beyond basic features. These requirements are designed to make the system scalable, secure, and accessible while maintaining user satisfaction. The focus on performance ensures tasks like attendance processing happen quickly, while reliability and availability guarantee minimal interruptions.

Security requirements protect sensitive data, creating a safe platform for students, lecturers, and PPH technicians. Usability and localization aim to make the system easy for everyone to use and support diverse languages. Lastly, maintainability ensures the system can adapt to future needs, reducing complexity in updating or improving features. Together, these NFRs ensure that the SmartClass system remains robust, dependable, and user-friendly for years to come.

CHAPTER 4

REQUIREMENT ANALYSIS

4.1 Introduction

Requirement analysis is a crucial phase in the development process, where the needs and expectations gathered during requirement elicitation are transformed into structured models and diagrams. This phase ensures a clear and detailed understanding of how the system will function and interact with its users. For the SmartClass system, requirement analysis serves as a foundation for verifying and validating that the design aligns with the stakeholders' requirements while ensuring feasibility and clarity for developers.

A key component of this phase involves the use of visual tools such as use case diagrams and descriptions. These tools provide a structured way to represent user interactions with the system, capturing the core functionalities and workflows. Use case diagrams help outline the system's behavior at a high level, while use case descriptions provide a deeper exploration of each scenario, detailing the interactions between users and the system in specific situations. By combining these techniques, the project gains a comprehensive view of how the SmartClass system meets user needs.

This chapter focuses on the analysis methods used for the SmartClass system, including use case diagrams and descriptions. These artifacts are vital for bridging the gap between stakeholder expectations and technical implementation. They not only aid in documenting system behavior but also provide a basis for communicating with stakeholders and validating that all critical functionalities are covered. Through this analysis, the SmartClass system is refined into a well-structured, user-centered solution ready for implementation.

4.2 Use Case Diagram

The Use Case Diagram provides a high-level visual representation of the SmartClass system's key functionalities and how they interact with the system's users. It illustrates the relationships between actors (such as students, lecturers, and PPH technicians) and the system's major processes or use cases. This diagram is a valuable tool for understanding the system's structure, and it helps clarify the scope of the system and how it supports the goals of each user.

The diagram offers a simplified view of the system, allowing stakeholders to quickly understand the main operations of the system. By showing which actors are involved in specific processes, we can confirm that all the necessary functionalities are included in the design. The use case diagram for the SmartClass system is shown below. It includes key actors and major use cases that define the business processes for students, lecturers, and PPH staff.

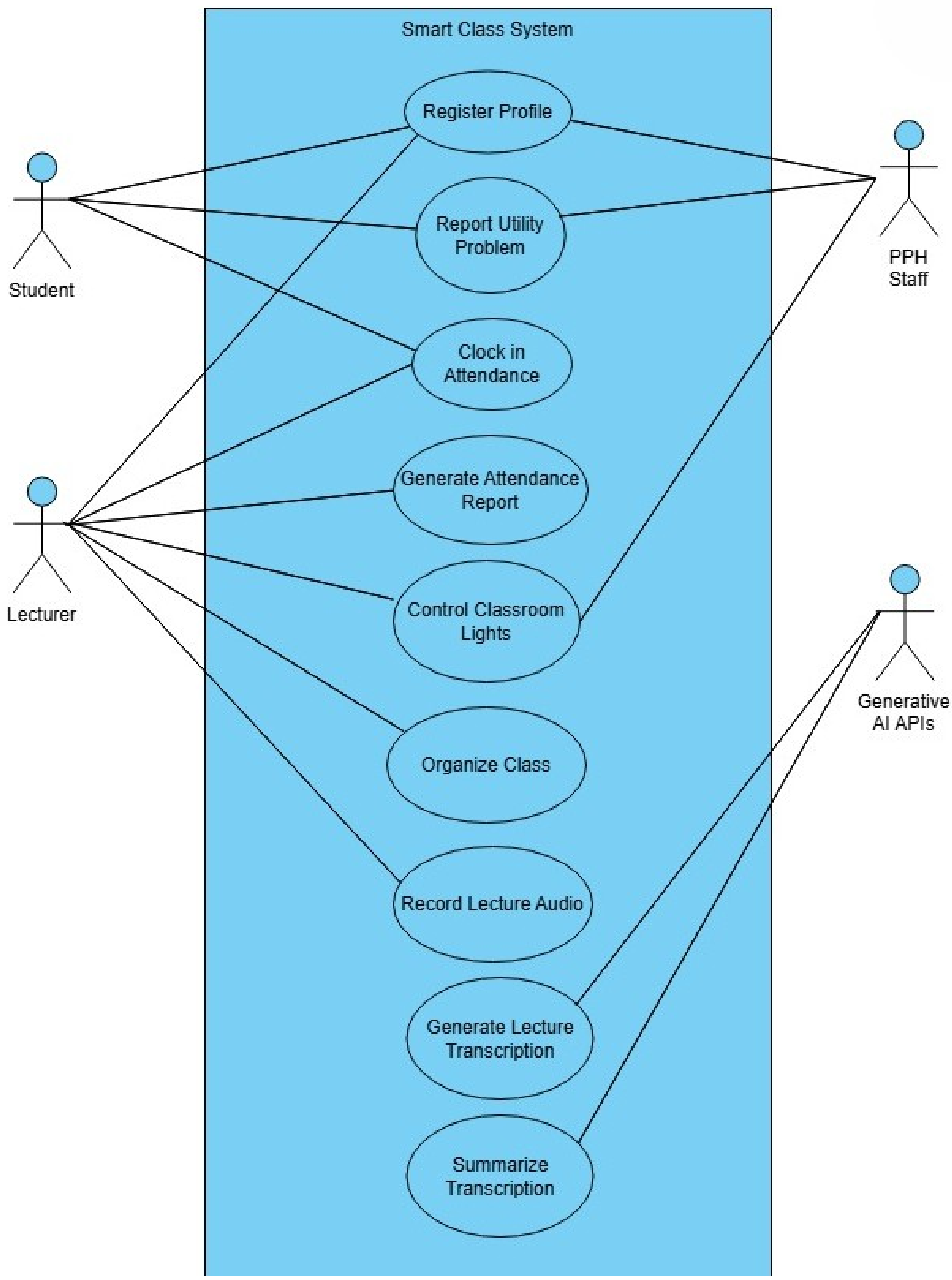


Figure 4.1: Use Case Diagram for SmartClass System

4.3 Use Case Description

Use-case descriptions provide a means to more fully document the different aspects of each individual use case. The use-case descriptions are based on the identified requirements, use-case diagram, and the activity diagram descriptions of the business processes. Use-case descriptions contain all the information needed to document the functionality of the business processes.

Table 4.1: Use Case 1: Register Profile

Use Case Name: Register Profile	**ID:** UC1	**Importance Level:** High
Primary Actor: Student, Lecturer, PPH Staff	**Use Case Type:** Primary, Essential	

Stakeholder and Interests:
Student - Wants to create or update their account for access to the system.
Lecturer - Needs to create or update their account to control classroom utilities and access to the system.
PPH Staff - Requires an account to manage and control utilities in the classroom.
Brief Description: This use case describes how a user can register a new account or update an existing one within the system.
Trigger: A user wants to create an account or update their existing profile.
Type: External
Normal Flow of Events:

1. The user accesses the registration page.
2. The user fills in their personal details (e.g., name, email, user role and etc).
3. The user submits the registration form.
4. The system validates the entered data.
5. If valid, the system creates a new account or updates an existing one.
6. The user is notified of the successful account creation/update.

Alternate/Exceptional Flows:
1a. Invalid data entered - The system prompts the user to correct the errors.
2a. Email already exists - The system notifies the user to choose another one.

Table 4.2: Use Case 2: Report Utility Problem

Use Case Name: Report Utility Problem	**ID:** UC2	**Importance Level:** High
Primary Actor: Student, Lecturer, PPH Staff	**Use Case Type:** Primary, Essential	

Stakeholder and Interests:
Student - Wants to ensure any malfunction in classroom utilities is reported promptly to avoid disruptions in lectures.
Lecturer - Needs reported problems to be addressed promptly to conduct classes efficiently.
PPH Technician - Requires accurate and timely problem reports to perform maintenance and repair tasks effectively.
Brief Description: This use case describes the process of reporting issues with classroom utilities (e.g., malfunctioning lights or equipment) and notifying the PPH technician for resolution.
Trigger: A classroom utility malfunctions and is noticed by the student or lecturer.
Type: External
Normal Flow of Events:

1. The user (student or lecturer) identifies an issue with the classroom utilities.
2. The user logs into the Smart Class app.
3. The user accesses the "Report Problem" feature and describes the issue.
4. The system logs the report and assigns a reference ID.
5. The system notifies the PPH technician about the issue.
6. The technician reviews the report and takes action to resolve the problem.
7. The system updates the issue status to "Resolved" and notifies the user.

Alternate/Exceptional Flows:
1a. The user is unable to report the problem due to network issues - The issue is reported via manual communication to the PPH technician.
6a. The technician cannot resolve the issue remotely - Onsite maintenance is scheduled.

Table 4.3: Use Case 3: Clock in Attendance

Use Case Name: Clock in Attendance	**ID:** UC3	**Importance Level:** High
Primary Actor: Student, Lecturer	**Use Case Type:** Primary, Essential	

Stakeholder and Interests:
Student - Wants to mark their own attendance for the class.
Lecturer - Needs to confirm the attendance of students in the class and also manually clock in the student's attendance if the system is problem.
Brief Description: This use case describes how a student clock in their attendance, and how the lecturer confirms it.
Trigger: The student arrives for class and wants their attendance recorded.
Type: External
Normal Flow of Events:

1. The student taps the RFID reader to clock in their attendance.
2. The student enter the class and the motion detector detect the person come in to the class.
3. The student's attendance will clock in only when one person is detected .
4. The lecturer can accesses the system and verifies the student's attendance.
5. The system confirms attendance has been successfully recorded.

Alternate/Exceptional Flows:
1a. Student fails to mark attendance by a certain time - Lecturer confirms the student's presence manually.

Table 4.4: Use Case 4: Generate Attendance Report

Use Case Name: Generate Attendance Report	**ID:** UC4	**Importance Level:** Medium

Primary Actor: Lecturer **Use Case Type:** Primary, Essential

Stakeholder and Interests:

Lecturer - Needs to generate reports for reviewing attendance patterns of students.

System - Produces timely and accurate attendance data to aid lecturers.

Brief Description: This use case describes how lecturers generate and view attendance reports for their classes.

Trigger: The lecturer requires an attendance report for class analysis.

Type: External

Normal Flow of Events:

1. The lecturer logs into the system.
2. The lecturer navigates to the *Generate Attendance Report* section.
3. The lecturer selects the class.
4. The system retrieves the necessary data and generates the report.
5. The system displays the generated attendance report.

Alternate/Exceptional Flows:

4a. No data found - The system notifies the lecturer with an error message.

Table 4.5: Use Case 5: Control Classroom Lights

Use Case Name: Control Classroom Lights	**ID:** UC5	**Importance Level:** High

Primary Actor: Lecturer, PPH Staff **Use Case Type:** Primary, Essential
Stakeholder and Interests:
Lecturer - Needs to control the lights based on different teaching modes (e.g., presentation, lecture).
PPH Staff - Manages the classroom environment by ensuring lights are set as required for different activities.
Brief Description: This use case describes how the lecturer or PPH staff controls the classroom lights for different classroom settings.
Trigger: The lecturer starts a class and needs to control the classroom lights.
Type: External
Normal Flow of Events:

1. The user (Lecturer or PPH staff) accesses the utility control system.
2. The user selects the appropriate light control option.
3. The system activates or deactivates the lights as per the selected setting.
4. The user confirms the changes.

Alternate/Exceptional Flows:
4a. Light system malfunction - The system notifies the user and provides troubleshooting options.

Table 4.6: Use Case 6: Organize Class

Use Case Name: Organize Class	**ID:** UC6 **Importance Level:** High
Primary Actor: Lecturer	**Use Case Type:** Primary, Essential

Stakeholder and Interests:
Lecturer - Needs to manage class information, including creating, updating, editing, and deleting class records, to organize academic sessions effectively.
System - Provides a user-friendly interface for managing class data and ensures data integrity.
Brief Description: This use case describes how lecturers can create, update, edit, and delete class information using the system.
Trigger: Lecturer accesses the system to organize class information.
Type: External
Normal Flow of Events:

1. The lecturer logs into the system.
2. The lecturer navigates to the ”Organize Class” section.
3. The lecturer selects an action (create, update, edit, or delete).
4. The system displays the appropriate interface for the selected action.
5. The lecturer inputs or modifies class information.
6. The system validates the data and saves the changes.
7. The system confirms the action with a success notification.

Alternate/Exceptional Flows:
2a. Lecturer enters invalid credentials - The system denies access and prompts the lecturer to retry.
6a. Data validation fails - The system displays an error message and prompts the lecturer to correct the input.
6b. System error occurs - The system notifies the lecturer and logs the error for administrative review.

Table 4.7: Use Case 7: Record Lecture Audio

Use Case Name: Record Lecture Audio	**ID:** UC7	**Importance Level:** High

Primary Actor: Lecturer | **Use Case Type:** Primary, Essential

Stakeholder and Interests:
Lecturer - Needs to record audio of lectures for future transcription and student review.
System - Captures the audio and ensures it is stored correctly for future use.
Brief Description: This use case describes how the lecturer records lecture audio for transcribing purposes.
Trigger: Lecturer starts the lecture and enables the recording feature.
Type: External
Normal Flow of Events:

1. The lecturer accesses the audio recording system.
2. The lecturer starts the recording using the apps.
3. The system records the lecture audio.
4. The recording is saved to the designated storage location (e.g., SD card).

Alternate/Exceptional Flows:
4a. System failure - The system notifies the lecturer and stops the recording.

Table 4.8: Use Case 8: Generate Lecture Transcription

Use Case Name: Generate Lecture Transcription	**ID:** UC8	**Importance Level:** High
Primary Actor: System	**Use Case Type:** Primary, Essential	

Stakeholder and Interests:
Student - Needs the transcription of lecture audio for study or revision.
Lecturer - Wants the lecture transcription for creating notes or summaries.
Brief Description: This use case describes how the system generates transcriptions from recorded lectures.
Trigger: Lecture audio recording has been completed, and transcription is required.
Type: External
Normal Flow of Events:

1. The system retrieves the recorded lecture audio.
2. The system sends the audio to a transcription service or API.
3. The transcription service processes the audio and returns the transcription.
4. The system saves the completed transcription.

Alternate/Exceptional Flows:
4a. Audio quality is poor - The system notifies the user of transcription failure or issues.

Table 4.9: Use Case 9: Generate Summarize Transcription

Use Case Name: Generate Summarize Transcription	**ID:** UC9	**Importance Level:** High

Primary Actor: System **Use Case Type:** Primary, Essential

Stakeholder and Interests:
Student - Needs the transcription or summarization of lecture audio for study or revision.
Lecturer - Wants the lecture transcription and summarization for creating notes or summaries.

Brief Description: This use case describes how the system generates summarization from transcription of the lectures.

Trigger: Lecture transcription has been completed, and summarization is required.

Type: External

Normal Flow of Events:

1. The system retrieves the transcription text of lecture.
2. The system sends the transcription to a summarization service or API.
3. The summarization service processes the transcription and returns the summarization texts.
4. The system saves the completed summarization and display to the lecturer.

Alternate/Exceptional Flows:
4a. Transcription is not available - The system notifies the user of transcription availability or issues.

The above use cases represent the primary functionalities of the Smart Class System. Key interactions include creating user accounts, marking attendance, and generating attendance reports. These processes ensure streamlined classroom operations, while the system provides accurate and timely data for lecturers and administrators. Alternate flows highlight potential issues and system responses, ensuring robust performance.

4.4 Activity Diagrams

Activity diagrams provide a visual representation of workflows within a system, showcasing how tasks and processes progress and interact. These diagrams clarify the logical sequence of activities, decisions, and interactions among system components or actors. The following diagrams illustrate key workflows in the Smart Class system, capturing user interactions, data flows, and decision points.

- **Activity Diagram for Register Profile**

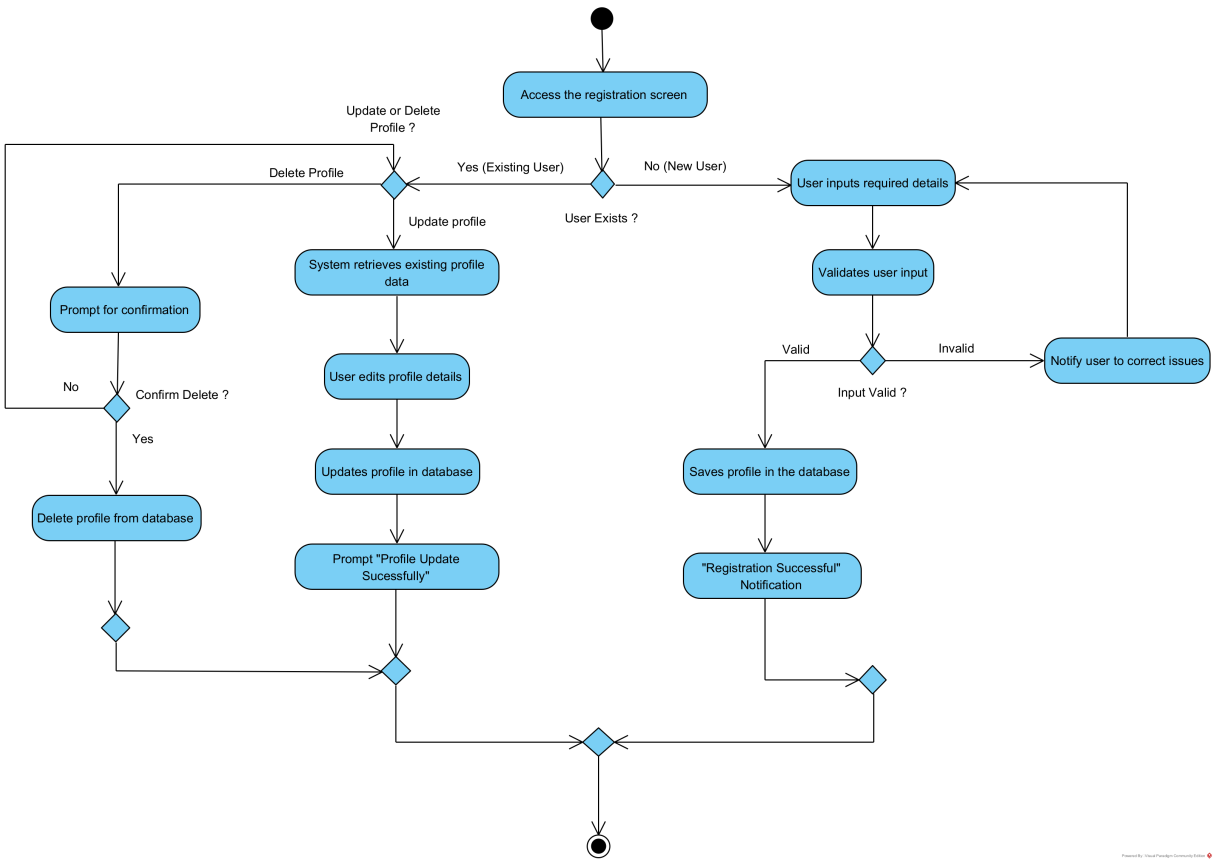


Figure 4.2: Activity Diagram for Register Profile

Activity diagram above shows how users create, change, or delete their profile. The diagram starts when a user opens the registration screen. They then can choose to make a new profile (for new user) or change or delete profile. If they're making a new profile, user must type in their information, and the system will checks the validity of the inputs. If everything is correct, the system saves the profile and prompts the user the registration information. If there is a problem with the information, the system prompt the user what to fix. If a user wants to change their profile, the system first finds their old information. They then make changes, and the system saves the updated profile. If user want to delete their profile, the system will prompt the user the confirmation if they are sure or delete. If user choose yes, the system will delete the profile.

- **Activity Diagram for Report Utility Problem**

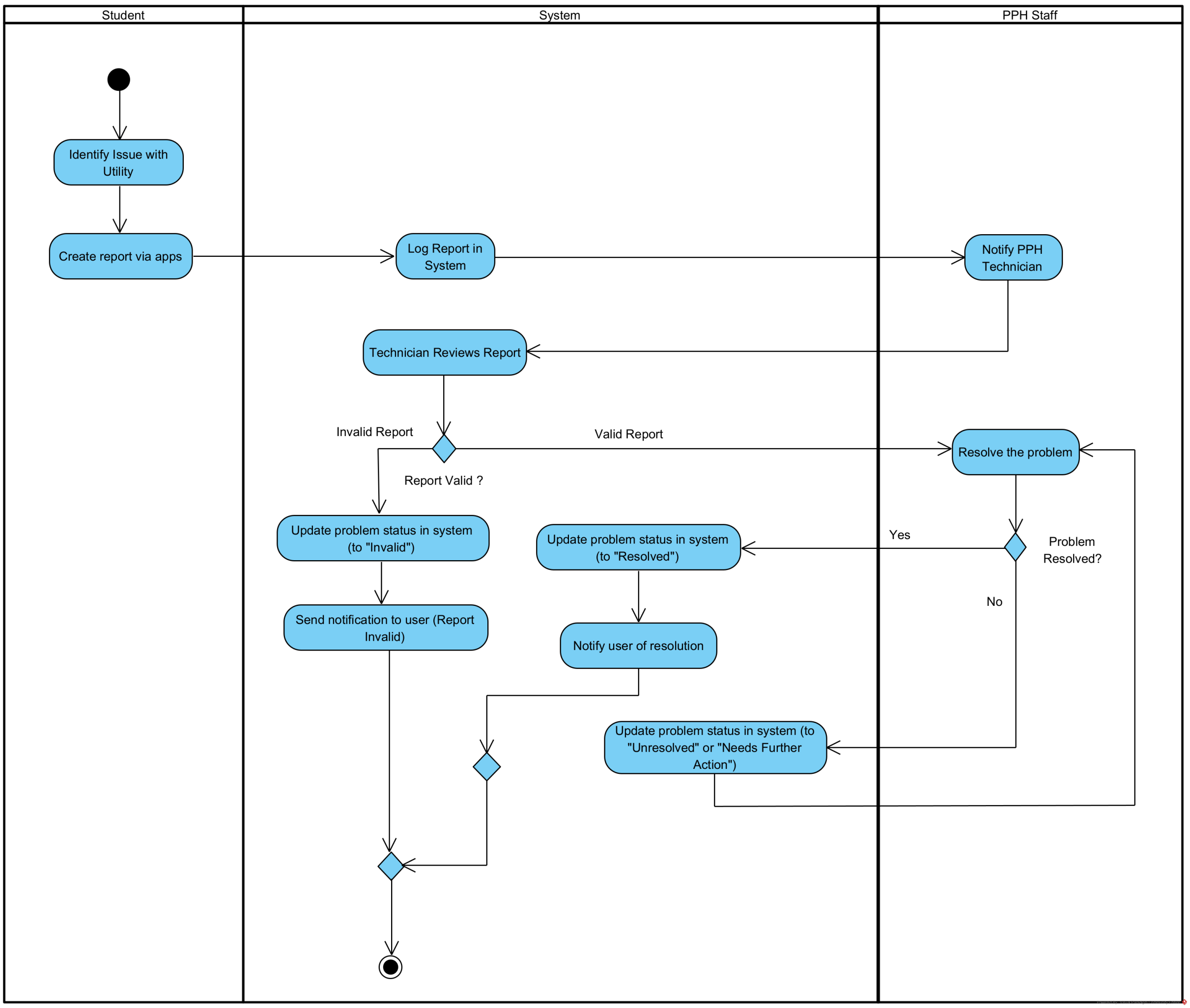


Figure 4.3: Activity Diagram for Report Utility Problem

Activity diagram above shows the process of reporting and resolving utility problems. A student first identifies a utility issue and creates a report. This report then is logged in the system, and a notification is sent to the PPH technician. The technician then reviews the report. If the report is invalid, the system updates the problem status to "Invalid" and sends a notification to the student. If the report is valid, the technician need to resolve the problem. After the problem has resolved, the technician checks if the problem is been resolved. If it is, the system updates the problem status to "Resolved" and notify the student. If the problem is not yet resolved, the system updates the problem status to either "Unresolved" or "Needs Further Action". This process ensures that all reported utility issues are tracked, reviewed, and addressed, with updates provided to the reporting student.

- **Activity Diagram for Clock in Attendance**

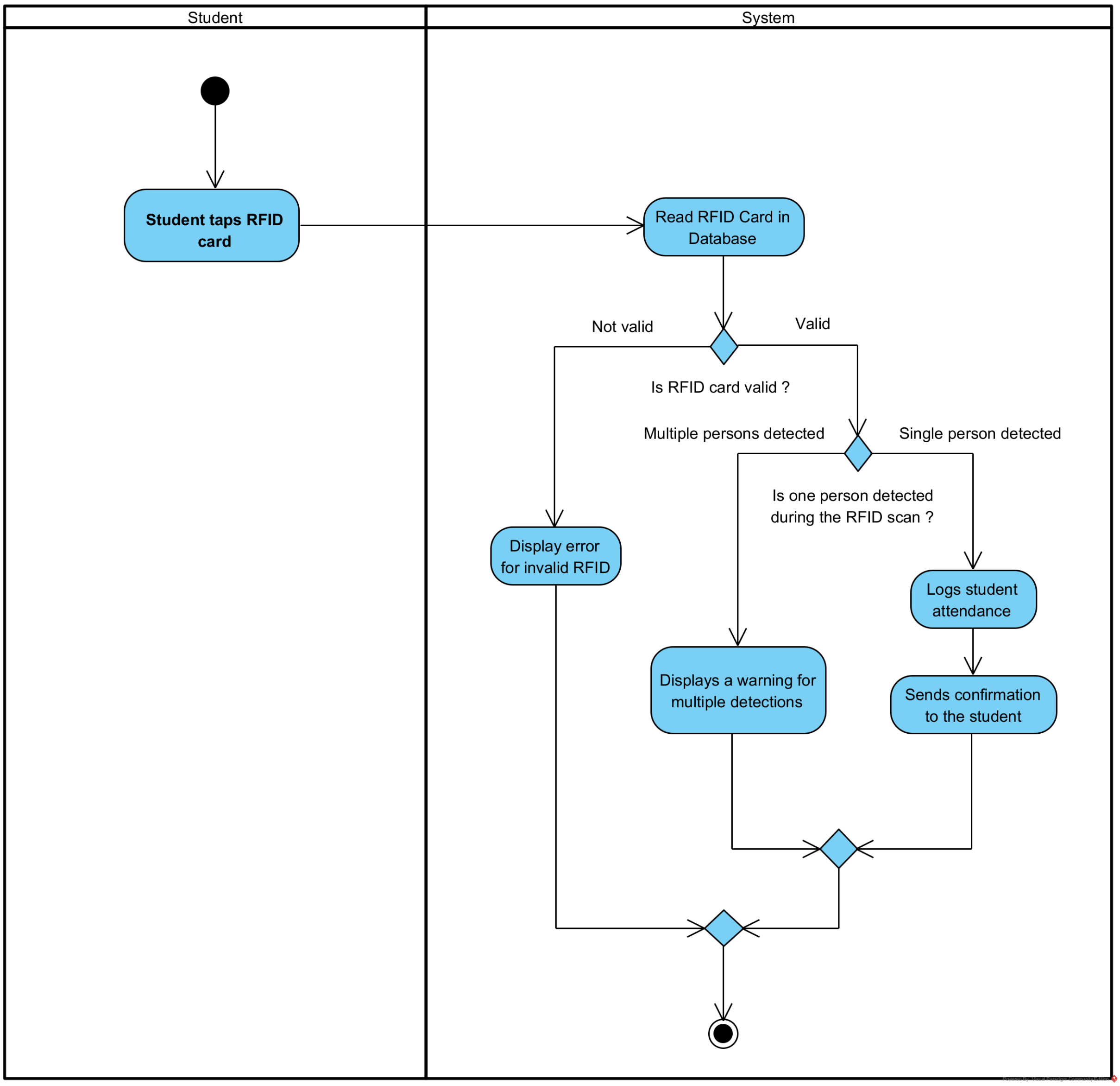


Figure 4.4: Activity Diagram for Clock in Attendance

This activity diagram above describes the student attendance marking process using RFID technology. A student initiates the process by tapping their RFID card on the reader. The system then verifies the card's validity. If the card is invalid, an error message is displayed. If valid, the system checks whether the person entered class is more than one person or not by using motion detector. If multiple person are detected, a warning sound were initiated and the attendance will not be marked. If a single valid scan is detected, the student's attendance is logged, and a confirmation is sent to the student. This process ensures accurate and efficient attendance tracking.

- **Activity Diagram for Generate Attendance Report**

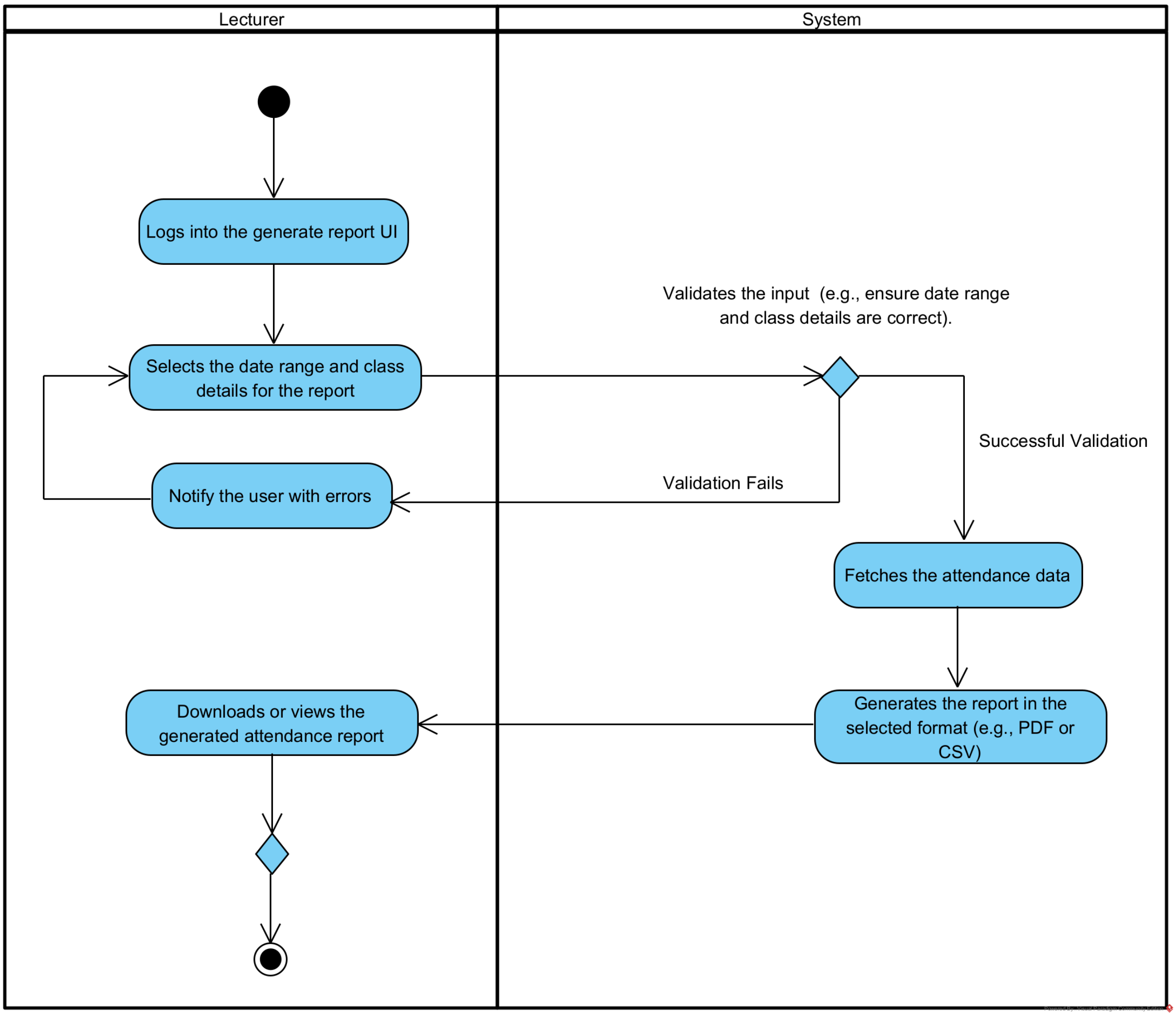


Figure 4.5: Activity Diagram for Generate Attendance Report

This activity diagram describes the process of generating attendance reports by the lecturer. The process begins with the lecturer logging into the report generation UI in the system. The lecturer then selects the desired date range and class details for the report. The system validates these inputs. If the validation fails, the lecturer is notified and must correct the inputs or shows the data in that particular range is not available or found. Upon successful validation, the system fetch the relevant attendance data from the database and generates the report in the selected format (PDF or CSV). Finally, the lecturer downloads or views the generated report.

- **Activity Diagram for Control Classroom Utilities (Lecturer)**

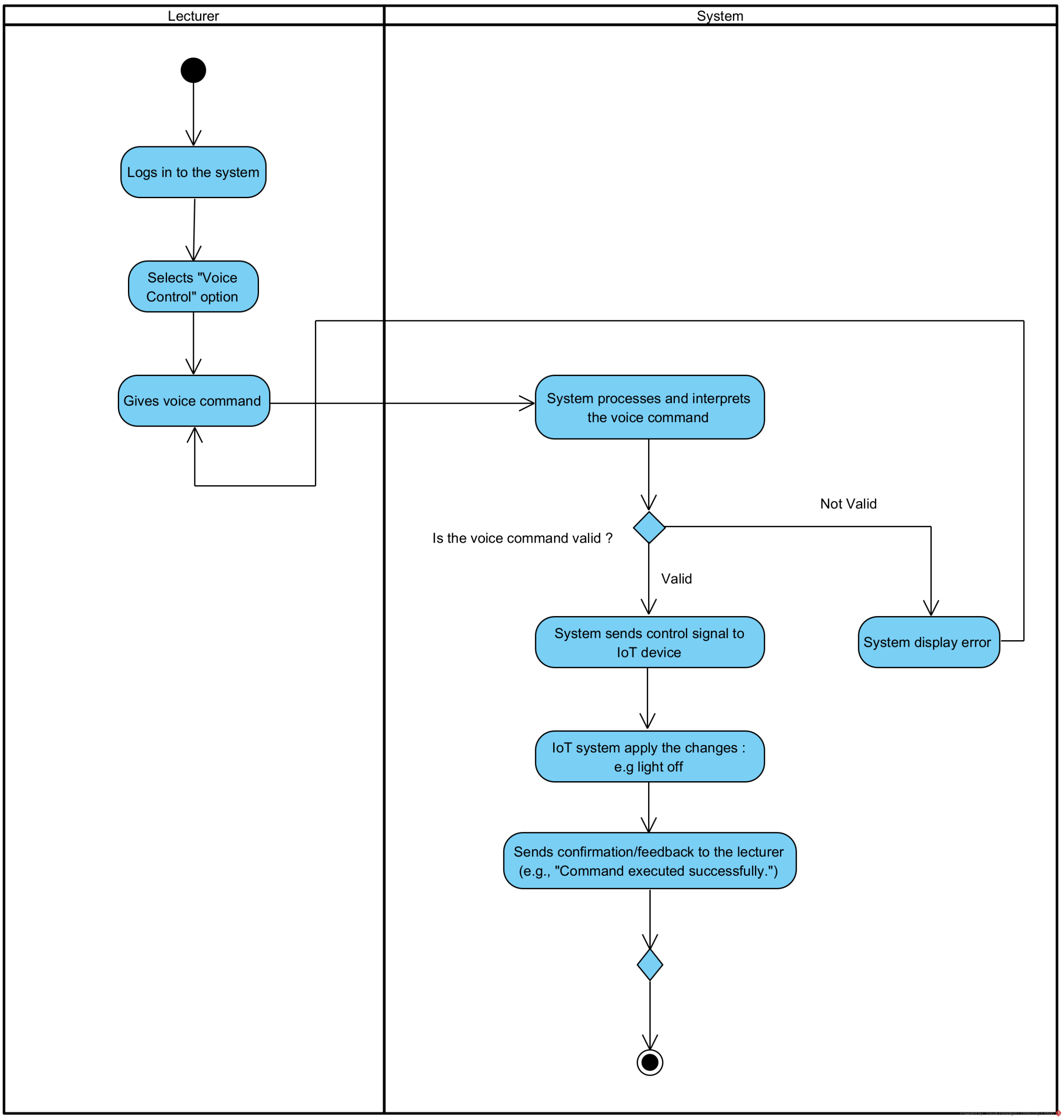


Figure 4.6: Activity Diagram for Control Classroom Utilities (Lecturer)

This diagram shows how a lecturer uses voice commands to control utilities in the classroom. First, the lecturer logs into the system and chooses the voice control option. Then, the lecturer gives command. The system validate the command and figures out what the command means, and checks if it's a valid command. If it's not a valid command, the system shows an error message. If the command is valid, the system sends a signal to the device (e.g lights and etc), and the device does according to the commands. Finally, the system sends a message back to the lecturer to confirm that the command was carried out successfully.

- **Activity Diagram for Control Classroom Utilities (PPH Staff)**

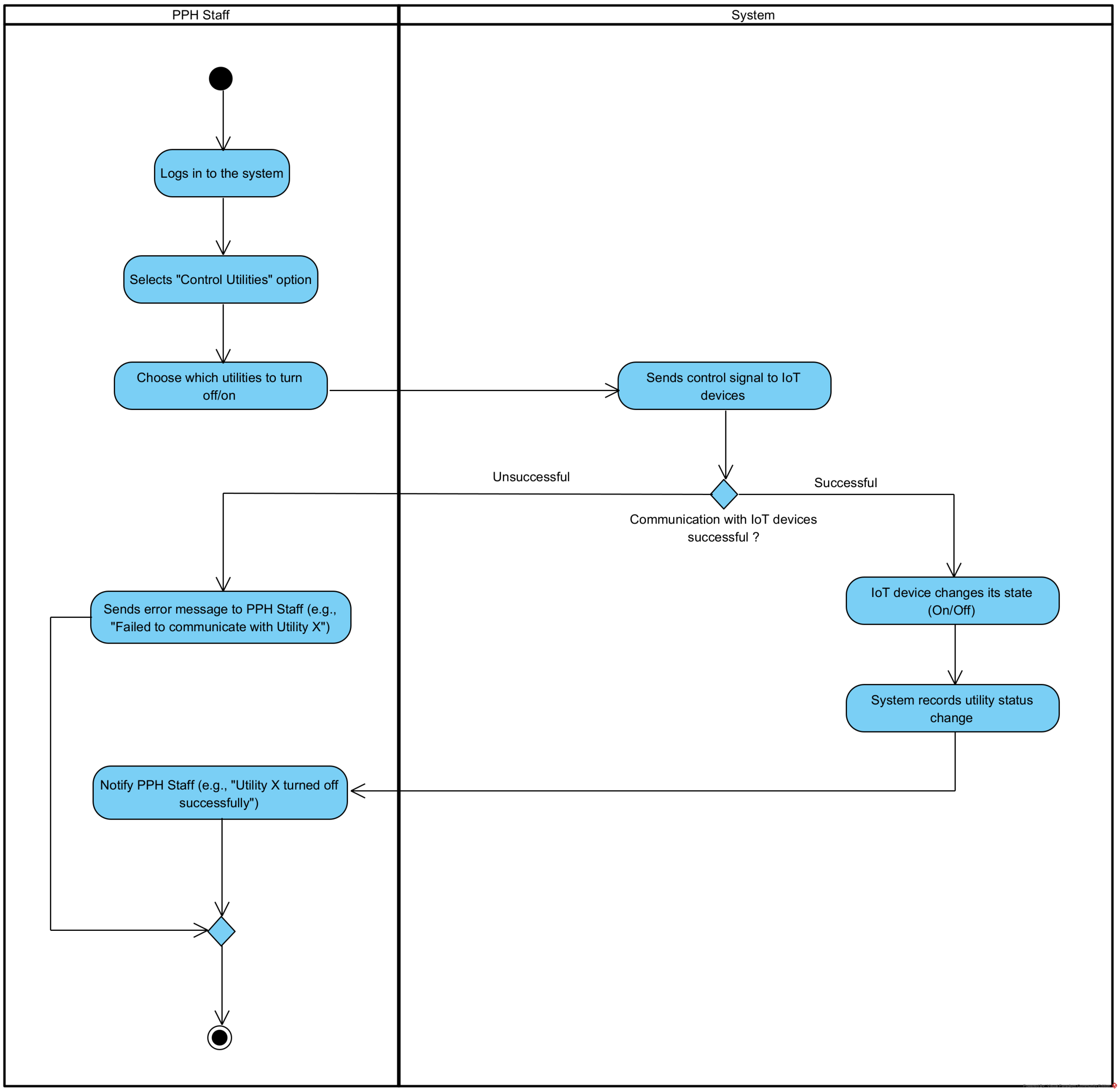


Figure 4.7: Activity Diagram for Control Classroom Utilities (PPH Staff)

Activity diagram above shows how PPH staff control classroom utilities. It starts with a PPH staff member logging into the system. They then select the "Control Utilities" option and choose which utilities they want to turn on or off. The system then tries to send a control signal to the corresponding IoT devices. If the communication with the devices is successful, the IoT devices change their state (turning on or off), and the system records this change. The system then notifies the PPH staff that the utility was successfully turned off or on. If the communication with the IoT devices is unsuccessful, the system sends an error message to the PPH staff, indicating that it failed to communicate with a specific utility.

- **Activity Diagram for Lecturer Organize Class**

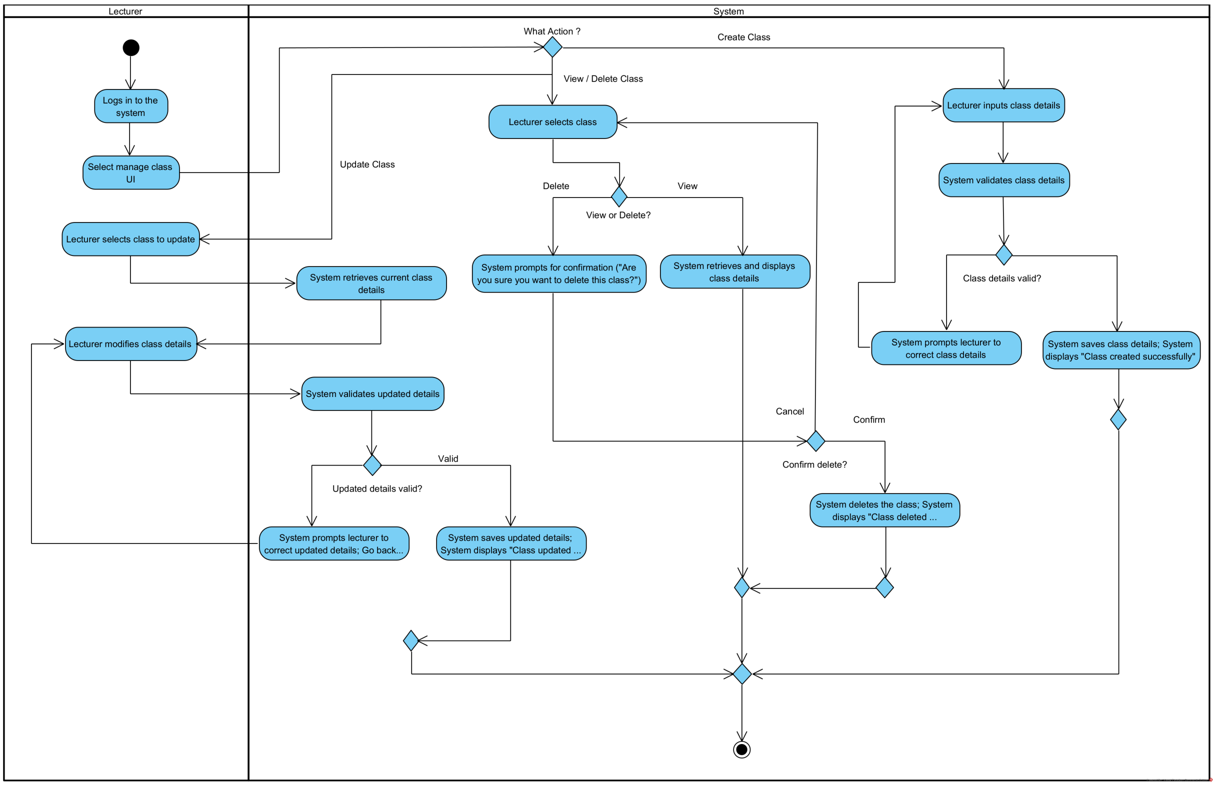


Figure 4.8: Activity Diagram for Lecturer Organize Class

Above activity diagram shows the process a lecturer follows to manage classes within the system. It begins with the lecturer logging into the system and selecting the "Manage Class" option. From there, the lecturer chooses what action they want to take: create, view/delete, or update a class.

If the lecturer chooses to create a class, lecturer must input the necessary class details. The system then validates these details. If the details are invalid, the system prompts the lecturer to correct them. If the details are valid, the system saves the class information and displays a "Class created successfully" message.

If the lecturer chooses to view or delete a class, first select the class they wish to view or delete. The system then retrieves and displays the class details. If the lecturer chooses to delete the class, the system prompts for confirmation. Upon confirmation, the system deletes the class and displays a "Class deleted" message to the lecturer. If the lecturer cancels the deletion, the process loop back to the lecturer selects class.

If the lecturer chooses to update a class, they first select the class they want to update. The system retrieves the current class details. The lecturer then modifies these details, and the system validates the updated information. If the updated details are invalid, the system prompts the lecturer to correct them. If the updated details are valid, the system saves the updated details and displays a "Class updated" message.

- **Activity Diagram for Generate Lecture Summarization**

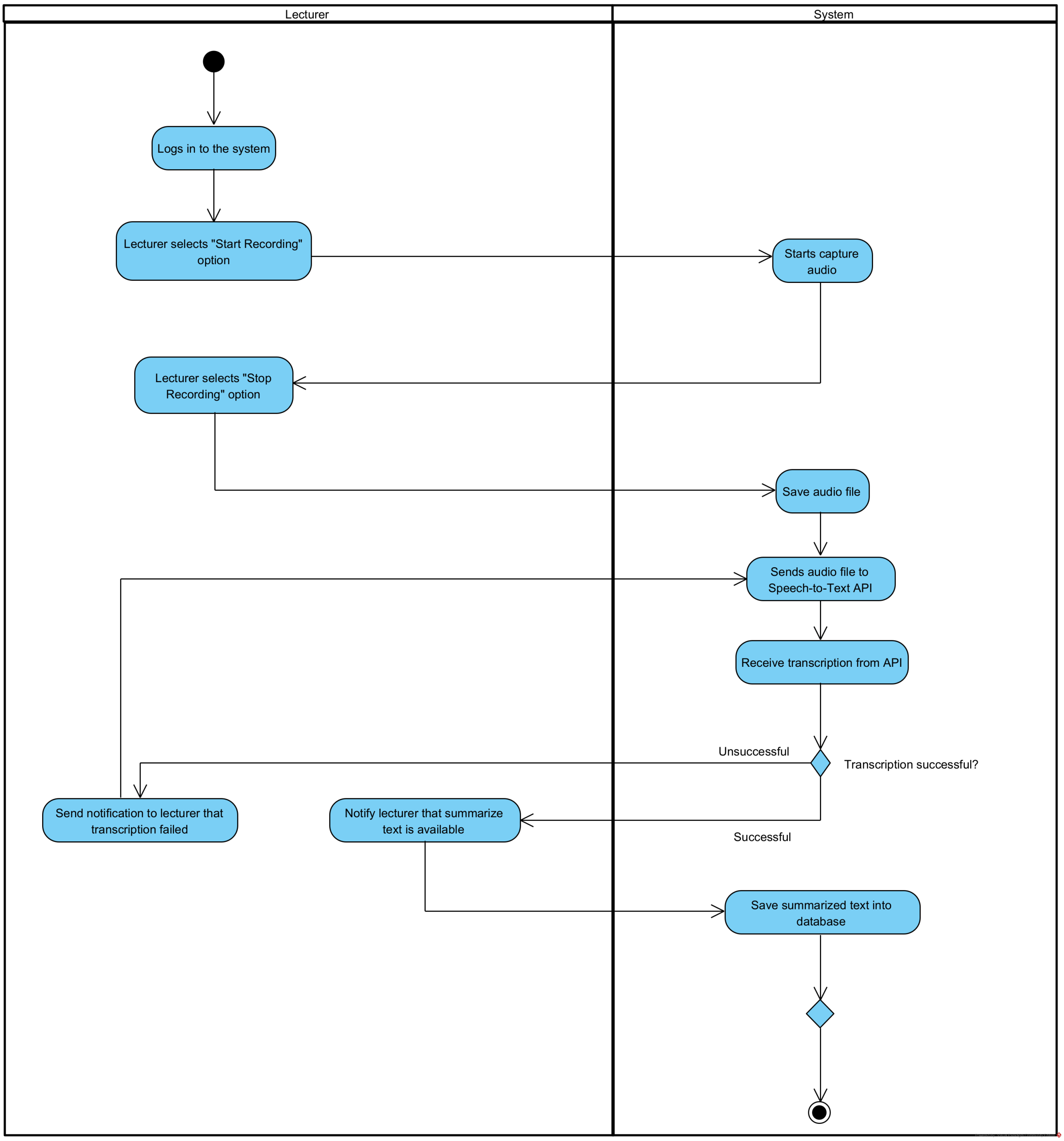


Figure 4.9: Activity Diagram for Generate Lecture Summarization

This above activity diagram illustrates the process of recording, transcribing, and summarizing lecture audio. The lecturer logs into the system and selects the 'Start Recording' option, which initiating audio capture IoT. Upon selecting 'Stop Recording,' the system saves the audio file and sends it to a Speech-to-Text API for transcription purposes. The resulting text is then saves to the system and after that will be sent to a Artificial Intelligence API for the summarization purposes. The system receives the summarized content and notifies the lecturer upon completion of the summarization.

- **Activity Diagram for Lecture Summarization Approval**

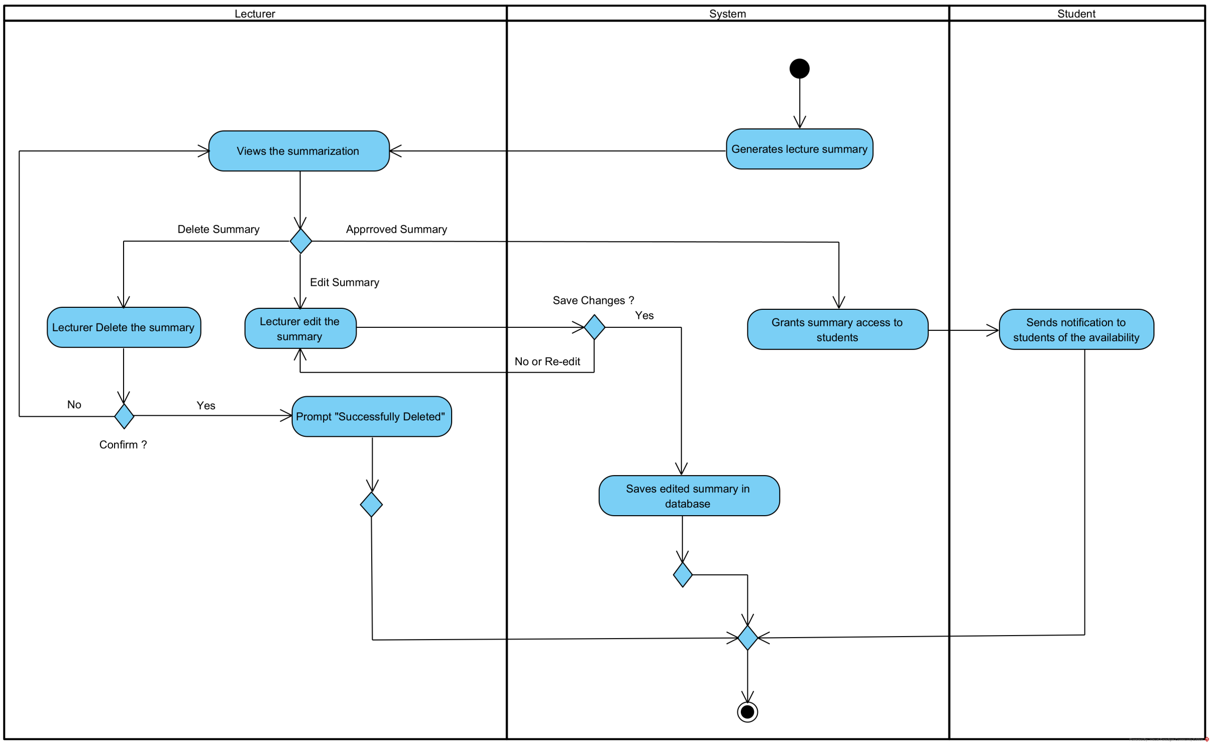


Figure 4.10: Activity Diagram for Lecture Summarization Approval

This above activity diagram shows the lecture summarization approval process by the lecturer. The system generates a lecture summary and notifies the lecturer of the completion of the availability of the summarization. The lecturer reviews the summary and decides whether to approve it or not. If the lecturer approved, the system grants student access of the summarization and sends a notification to students for the availability of the summarization. If not approved, student access is restricted, and the summary is marked as pending approval in the lecturer page.

- **Activity Diagram for Student View Summarization**

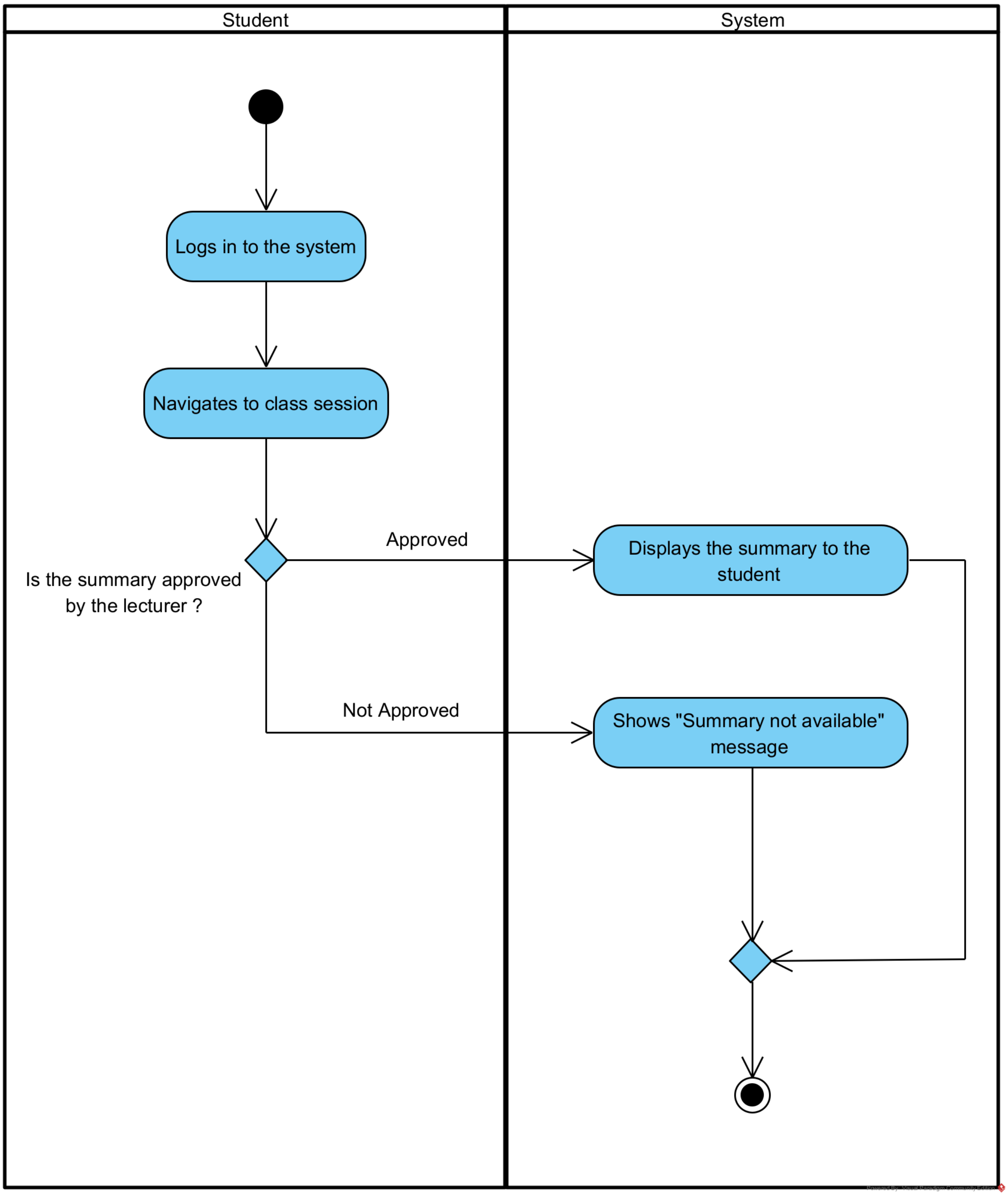


Figure 4.11: Activity Diagram for Student View Summarization

This activity diagram above illustrates the process of a student accessing a lecture summarization from the apps. The student logs into the system and navigates to the relevant class session. The system then checks if the lecturer has approved the summary or not. If approved, the summary is displayed to the student. If not approved, a 'Summary not available' message is displayed to the students.

4.5 Class Diagram

A class diagram serves as a static representation of the system's structure by illustrating the various classes, their attributes, methods, and the relationships between them. It provides an overview of how the different components are organized and how they interact with each other within the system. This diagram plays an essential role in understanding the system's design, offering insights into its foundational architecture and ensuring the proper organization of functionality.

For the Smart Class system, the class diagram highlights the key components, such as users, utilities, and databases, along with their relationships and behaviors. By examining this diagram, developers can gain a clear understanding of how data flows through the system and how the components remain connected over time. It also assists in ensuring that all requirements are met during development.

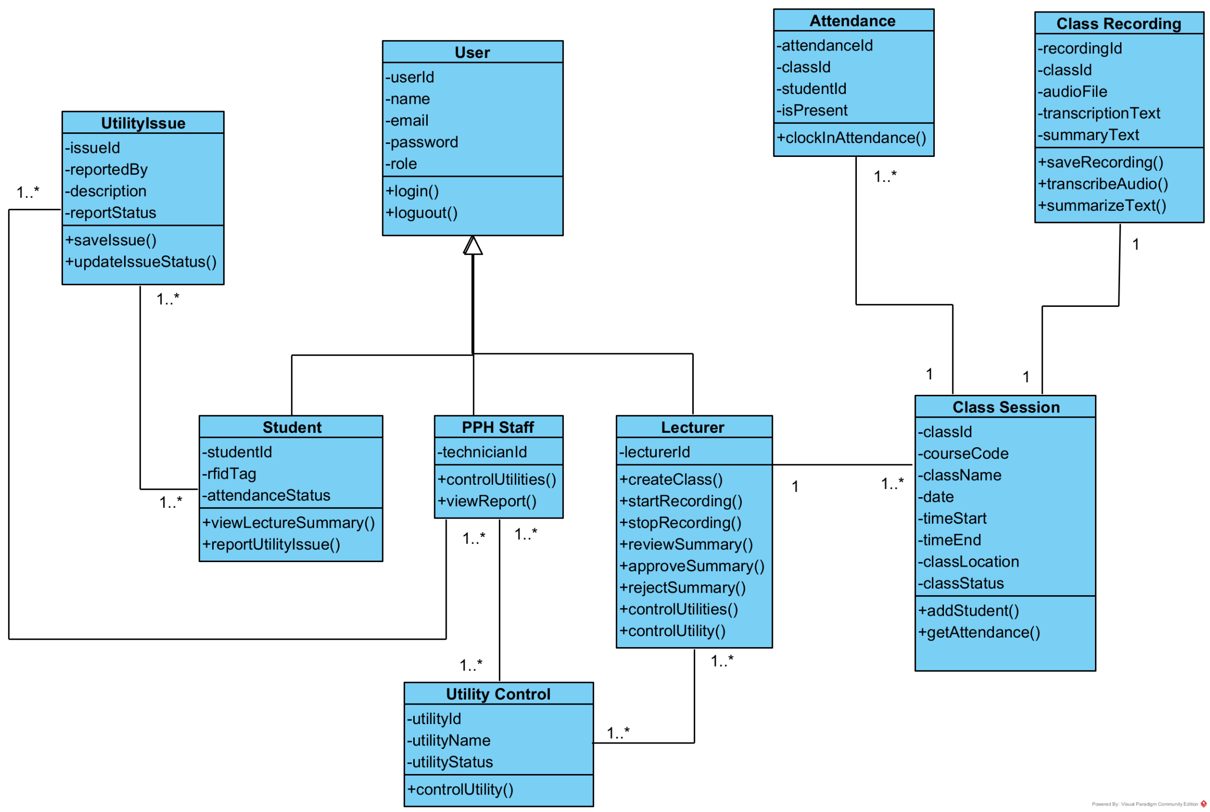


Figure 4.12: Class Diagram for SmartClass System

The class diagram represents the core components of the Smart Class system, showcasing their attributes, methods, and the relationships among them. At the center is the `User` class, which acts as a generalization for specific roles such as `Student`, `Lecturer`, and `PPH Staff`. Each subclass inherits common properties, such as `userId`, `name`, `email`, and role-specific behaviors, while also defining unique methods. For instance, the `Student` class can view lecture summaries and report utility issues, whereas the `Lecturer` class has methods for managing class sessions and approving summaries.

The `Class Session` class is directly associated with both `Attendance` and `Class Recording`, signifying that attendance tracking and audio transcriptions are integral to individual class sessions. The `Attendance` class handles the presence of students, while the `Class Recording` class focuses on managing recordings, transcription, and summarization features.

Additionally, the diagram highlights `Utility Control` and `Utility Issue` classes, which are particularly relevant for `PPH Staff`. These classes represent the IoT-based classroom utilities and issue tracking systems, respectively. The relationships in the diagram illustrate how `PPH Staff` can monitor and manage utilities and respond to utility issues as needed. Overall, this class diagram effectively demonstrates the structure and interactions required to implement the functionalities of the Smart Class system.

4.6 Sequence Diagrams

Sequence diagrams are essential tools used to visualize how objects in a system interact within the context of a particular use case. They show the details flow of messages between objects and also the actor, emphasizing the order in which these interactions occur over time. This type of diagram helps provide a clear picture of the system's behavior during specific events in details, making it an invaluable tool in software design nowadays.

In the context of the Smart Class system, sequence diagrams play a very crucial role in illustrating how different components, such as students, lecturers, and the system itself, interact to complete tasks like generating attendance reports, marking attendance, and handling utility issues. By detailing the specific flow of messages between the system's entities, these diagrams allow for a more precise understanding of the timing and dependencies involved in the system, helping to ensure the smooth operation of all processes in the system, and also make sure that the complete flow is in details. The following is the sequence diagram for the Smart Class system, highlighting the interaction between the system and its users.

- **Sequence Diagram for Register Profile**

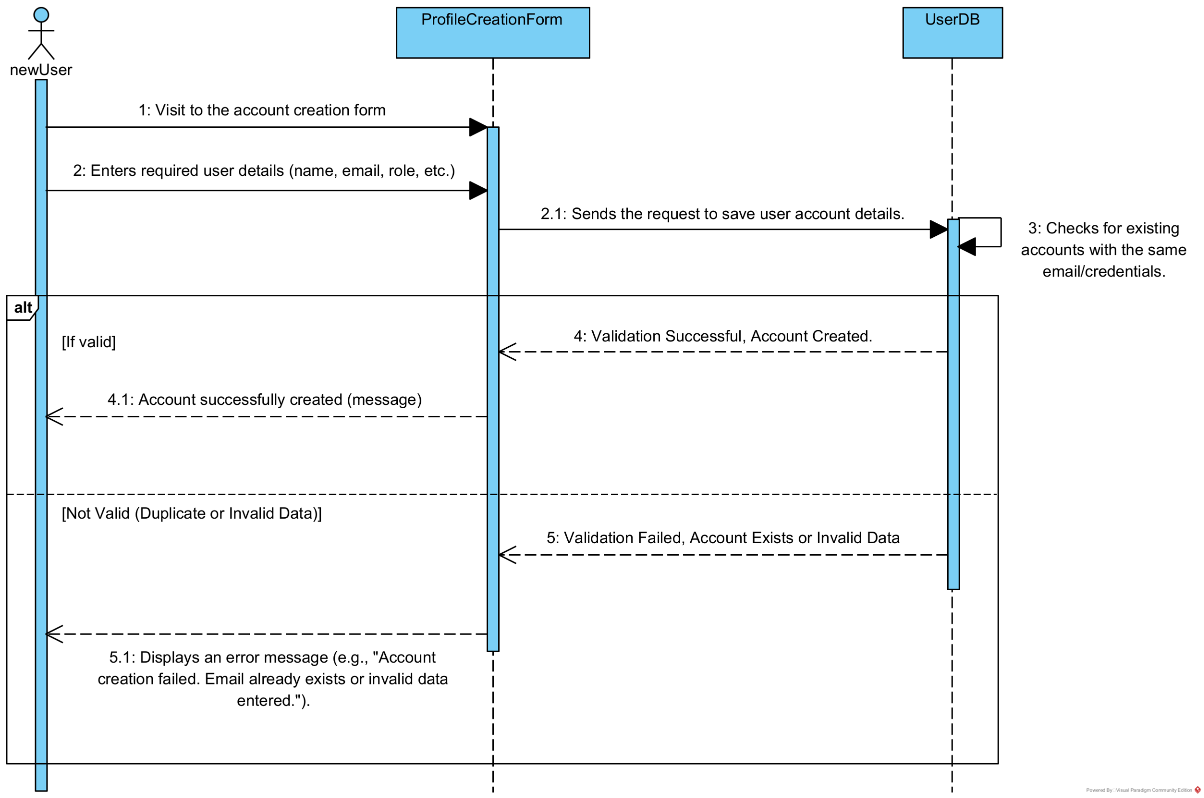


Figure 4.13: Sequence Diagram for Register Profile

This sequence diagram above shows the user registration process. A new user visits the account creation form and enters the required details. The form sends this data to the system, which checks the database for existing accounts or not. If the data is valid or no duplication in the database, the account is created, and a success message is displayed to the user. If the data is invalid (e.g., duplicate email), an appropriate error message is shown to the user of the error.

- **Sequence Diagram for Report Utility Problems**

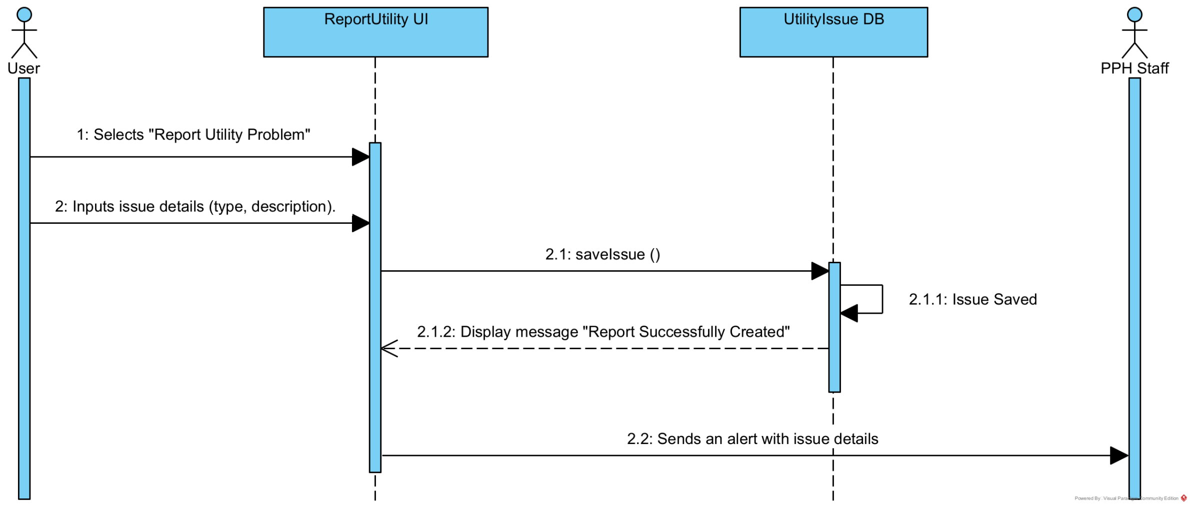


Figure 4.14: Sequence Diagram for Report Utility Problems

This sequence diagram above illustrates the process of user reporting utility problems to the PPH Staff by using the apps. The users selects the 'Report Utility Problem' option and then user should inputs the issue details such as issue type and issue description. The UI then sends a request to the database to save the issue in the database. Upon successful of saving the issue, a confirmation message is displayed or shown to the user, and an alert with the issue details is sent to the PPH staff.

- **Sequence Diagram for Clock in Attendance**

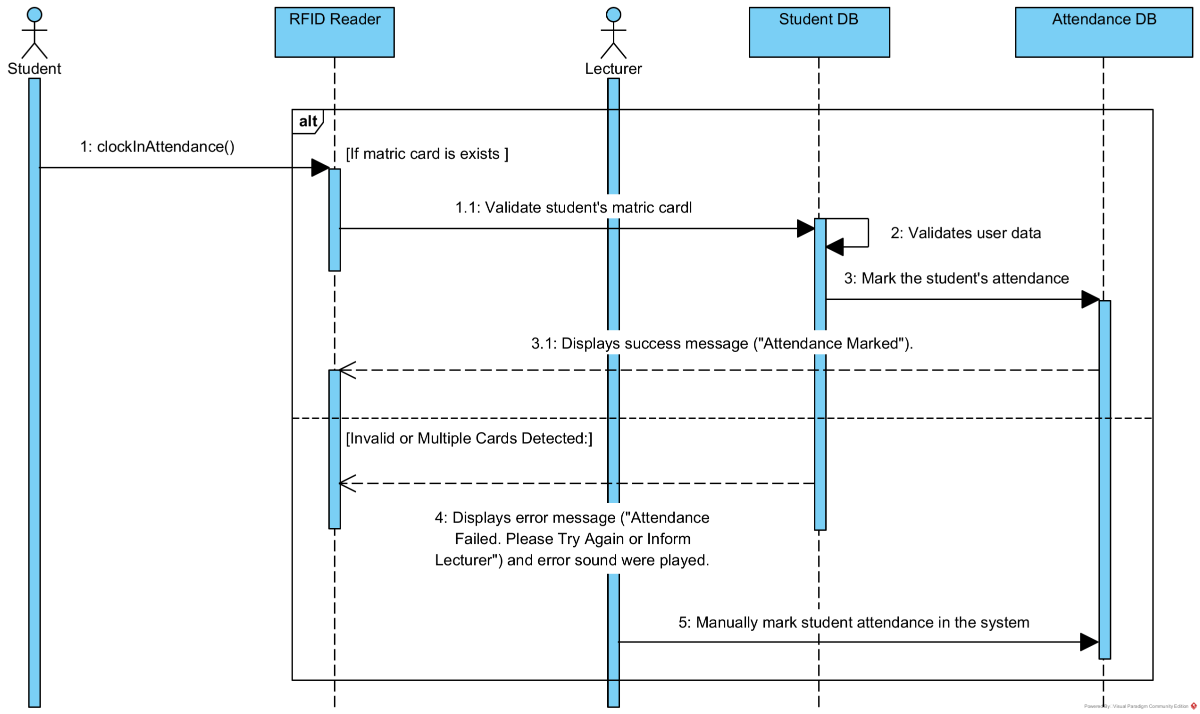


Figure 4.15: Sequence Diagram for Clock in Attendance

This sequence diagram shows the process of student attendance clock-in process using RFID. A student initiates clock-in using their matric card, triggering the RFID reader's clockinAttendance() function. The reader validates the matric card against the Student Database. Upon successful validation, the system marks the student's attendance in the Attendance Database and displays a "Attendance Marked" message to the student. If the card is invalid or multiple person are detected by the motion sensor when entering the class, an error message ("Attendance Failed...") and sound are played. In case of failure, the lecturer can manually mark attendance of the student in the system.

- **Sequence Diagram for Generate Attendance Report**

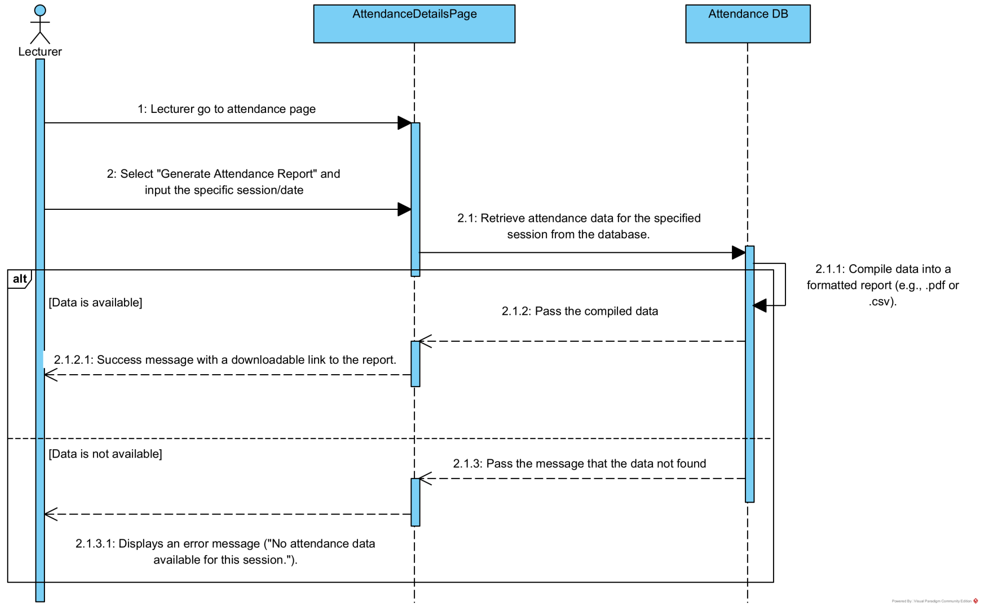


Figure 4.16: Sequence Diagram for Generate Attendance Report

This sequence diagram shows the process of generating an attendance report by lecturer. A lecturer navigates to the attendance page and selects the ”Generate Attendance Report” option, specifying the desired session or date. The AttendanceDetailsPage then requests the corresponding attendance data from the Attendance Database. If the data is available, the database compiles it into a formatted report like (e.g., PDF or CSV) and sends it back to the AttendanceDetailsPage to display to the lecturer, which then presents a success message with a download link to the lecturer. If no data is found in the database for the specified session, the database sends a ”Data not found” message to the AttendanceDetailsPage, which will displays an error message to the lecturer indicating that no attendance data is available for that session.

- **Sequence Diagram for Control Class Utility**

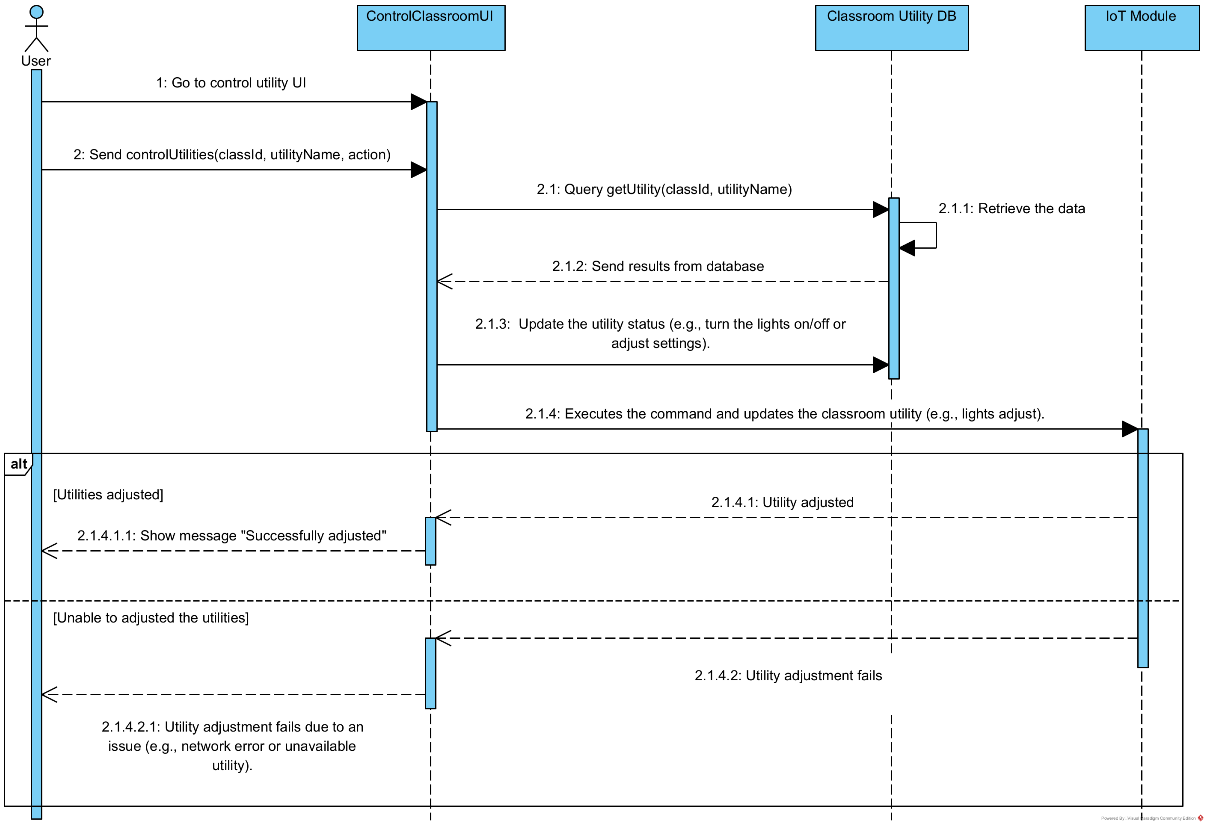


Figure 4.17: Sequence Diagram for Control Class Utility

This sequence diagram describes the process of a user (Lecturer and PPH Staff) controlling classroom utilities such as lights, and etc. The user navigates to the control utility UI and sends a control request to the system, specifying the class ID, utility name (e.g., lights, projector), and desired action (e.g., on, off, adjust). The ControlClassroomUI then queries the Classroom Utility Database to retrieve the relevant utility information. The database retrieves and sends this data back to the UI. The UI then updates the utility status in the database and sends a command to the IoT Module to execute the requested action on the physical utility in the classroom. If the utility is successfully adjusted, the IoT Module confirms this, and the UI displays a "Successfully adjusted" message to the user. However, if the utility adjustment fails due to an issue like a network error or an unavailable utility, the IoT Module reports the failure, and the UI displays a corresponding error message to the user explaining the reason for the failure.

- **Sequence Diagram for Record Lecture Audio**

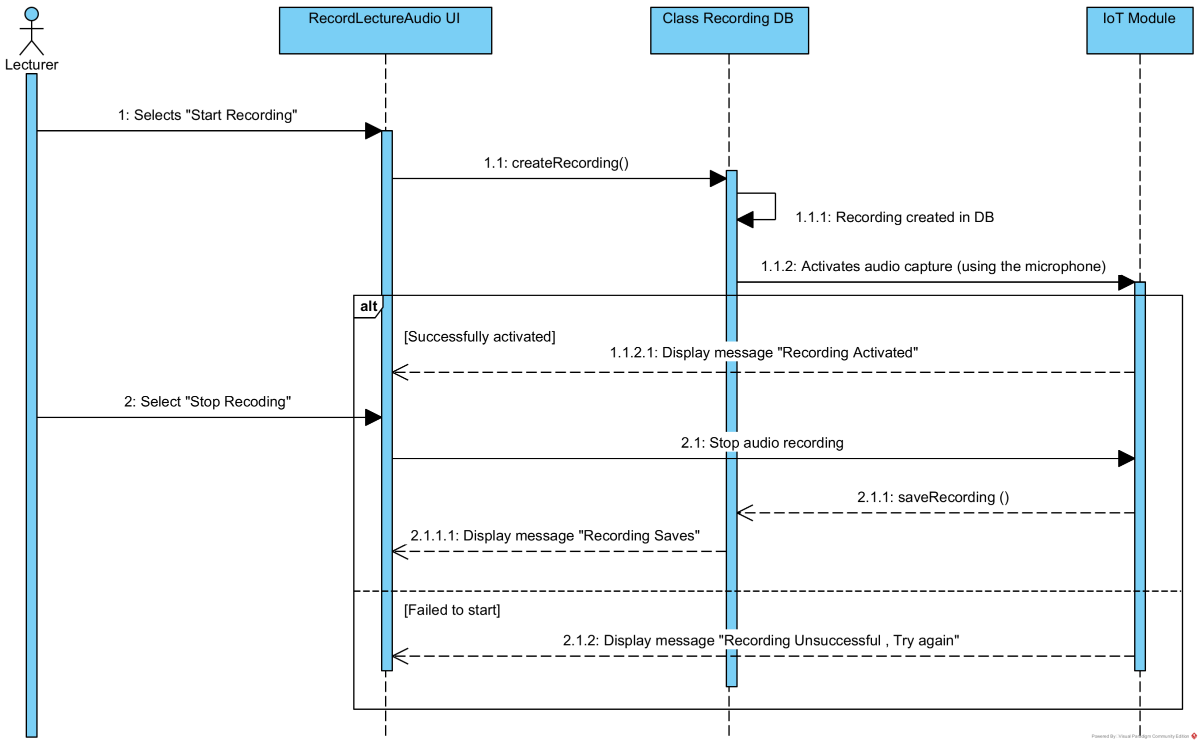


Figure 4.18: Sequence Diagram for Record Lecture Audio

This sequence diagram shows the process of recording a lecture's audio. The lecturer initiates the recording by selecting the "Start Recording" option in the RecordLectureAudio UI in the system. This triggers the creation of a new recording entry in the Class Recording Database. The database confirms the recording's creation and then activates the audio capture using the microphone via the IoT Module. If the audio capture is successfully activated, a "Recording Activated" message is displayed in the UI. When the lecturer selects "Stop Recording," the UI sends a signal to the IoT Module to stop the audio recording. The IoT Module then saves the recording to the Class Recording Database. Upon successful saving, a "Recording Saved" message is displayed in the UI. If the audio capture fails to start initially, a "Recording Unsuccessful, Try again" message is displayed in the UI to the lecturer, indicating a problem with the recording process.

- **Sequence Diagram for Generate Lecture Transcription**

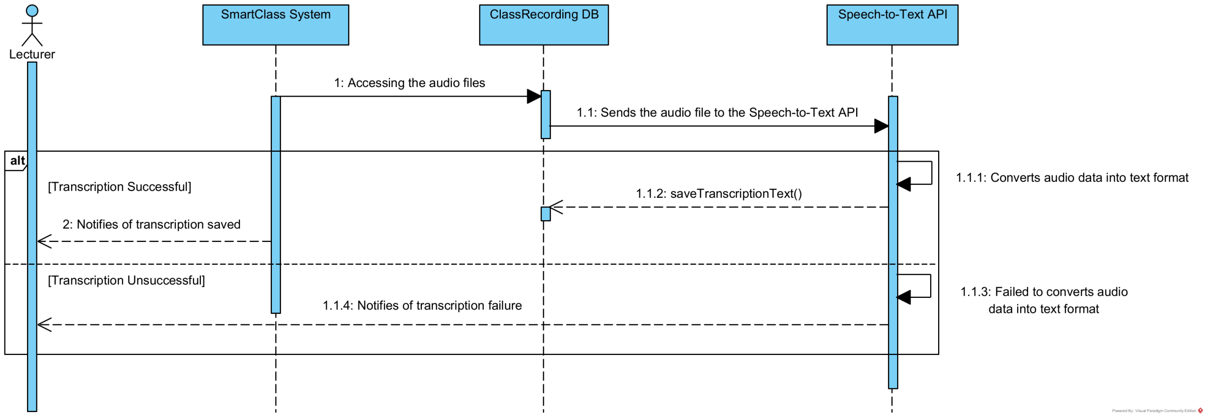


Figure 4.19: Sequence Diagram for Generate Lecture Transcription

This sequence diagram shows the process of generating a lecture transcription by using Speech-to-Text API. The lecturer initiates the process by accessing the audio files within the SmartClass System. The SmartClass System then sends the selected audio file to the Speech-to-Text API for conversion to the text. The Speech-to-Text API attempts to convert the audio data into text format. If the transcription is successful, the API sends the resulting text back to the SmartClass System, which then saves the transcription text into the ClassRecording Database and notifies the lecturer that the transcription has been successfully saved. However, if the API fails to convert the audio data into text, it notifies the SmartClass System of the transcription failure, which in turn notifies the lecturer about the unsuccessful transcription.

- **Sequence Diagram for Generate Summarization**

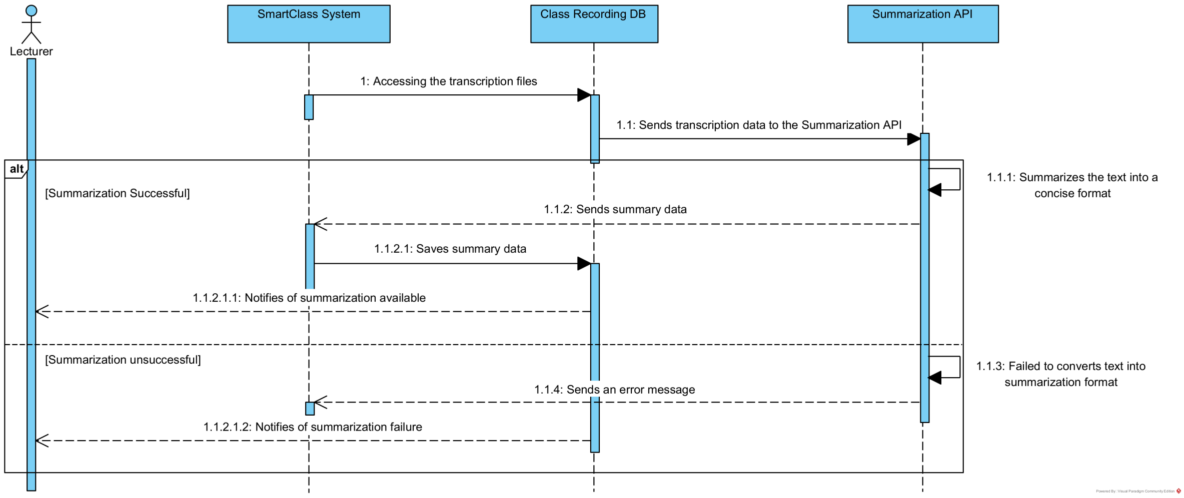


Figure 4.20: Sequence Diagram for Generate Summarization

This sequence diagram illustrates the process of generating a lecture summarization by using Generative AI API. A lecturer initiates the process by accessing the transcription files through the SmartClass System. The SmartClass System then sends the transcription data to the Summarization API. The Summarization API attempts to summarize the text into a more simpler format or maybe in bullet point form. If the summarization is successful, the API sends the summary data back to the SmartClass System, which saves the summary data to the Class Recording Database and then notifies the lecturer that the summarization is successfull and available. If the summarization process is unsuccessful (e.g., due to errors in the text or issues with the API), the Summarization API sends an error message back to the SmartClass System, which in turn notifies the lecturer of the summarization failure.

4.7 CRUD Matrix

The CRUD matrix is a method that used to analyze how actors or modules in a system interact with the underlying data entities. This technique is very valuable in breaking down each actor's or component's operations into four main categories: Create (C), Read (R), Update (U), and Delete (D). By systematically mapping these interactions, it becomes more easier to understand the responsibilities of each module, the dependencies between various components, and how data flows within the system. Such analysis ensures that every entity is appropriately managed and accounted for within the system's design.

In the context of the Smart Class system, the CRUD matrix offers insights into how key actors, such as lecturers, students, and PPH staff, collaborate with different modules and databases. It highlights operations performed on classroom utilities, attendance records, lecture notes, and user databases. This comprehensive approach not only just helps in defining system functionality but also aids in verifying that all critical operations are supported by the design. The table below provides a detailed overview of these CRUD interactions, offering a clear and structured representation of the system's workflows.

Actor/Class	**Classroom Utility**	**Attendance**	**Lecture Summarization**	**Student DB**	**Utility Report**
Lecturer	R, U	C, R, U	C, R, U, D	R	C, R, U, D
Student		C, R	R	R	C, R, U, D
PPH Staff	C, R, U, D			C, R, U, D	R, U, D
RFID Reader		C		R	

Table 4.10: CRUD Matrix for Smart Class System

CHAPTER 5

SUMMARY

This Software Requirements Specification (SRS) document provides a comprehensive outline for the development of the Smart Class system for the Pejabat Pembangunan dan Harta (PPH) UMT. The system is aims to modernize classroom management by integrating the features such as AI-powered transcription and summarization of lectures, IoT-based control of utilities, and automated RFID attendance tracking to make clock in attendance is more easy to a student and easier to lecturer to marked the attendance. It is designed to improve the experience of students, lecturers, and also PPH staff by offering functionalities like lecture summarization, utility monitoring, and student attendance report generation.

The document elaborates on the system's purpose, scope, and objectives, requirement elicitation technique which basically explain how we get the requirement. Basically, our requirement sources is from stakeholders which is PPH Staff, Lecturers and Students who actually are the user of the Smart Class system. Moreover, we also analyze the documents and also the existing system, This SRS Documentation also offering clear definitions of functional and non-functional requirements. Various diagrams, including use case, class, and activity diagrams, are included to demonstrate workflows, data relationships, and system behavior. These visuals provide insights into how different components, such as students, lecturers, and the PPH team, interact within the system. Additionally, each section carefully outlines the technical, operational, and user-related aspects of the Smart Class, ensuring that all requirements are properly documented.

By addressing system functionalities and ensuring scalability, this SRS serves as a critical guide for the design and implementation phases. It establishes a shared understanding among stakeholders, laying the foundation for an innovative and efficient Smart Class system tailored to meet the PPH needs of the system.

REFERENCES

Chan, E. K. F., Othman, M. A., and Abdul Razak, M. A. (2017). Iot based smart classroom system. *Journal of Telecommunication, Electronic and Computer Engineering*, 9(3-9):95–100.

Hercog, D., Lerher, T., Truntič, M., and Težak, O. (2023). Design and implementation of esp32-based iot devices. *Sensors*, 23(15).

khan, I. and Rashid, D. (2024). Internet of things (iot) its applications and enabling technologies in healthcare: A review. *Educational Administration Theory and Practices*, 30.

Salame, I. I., Tuba, M., and Nujhat, M. (2024). Note-taking and its impact on learning, academic performance, and memory. *International Journal of Instruction*, 17(3):599–616.

Stryker, C. and Kavlakoglu, E. (2024). Artificial intelligence.

Wikipedia Contributors (2019). Radio-frequency identification.

APPENDICES

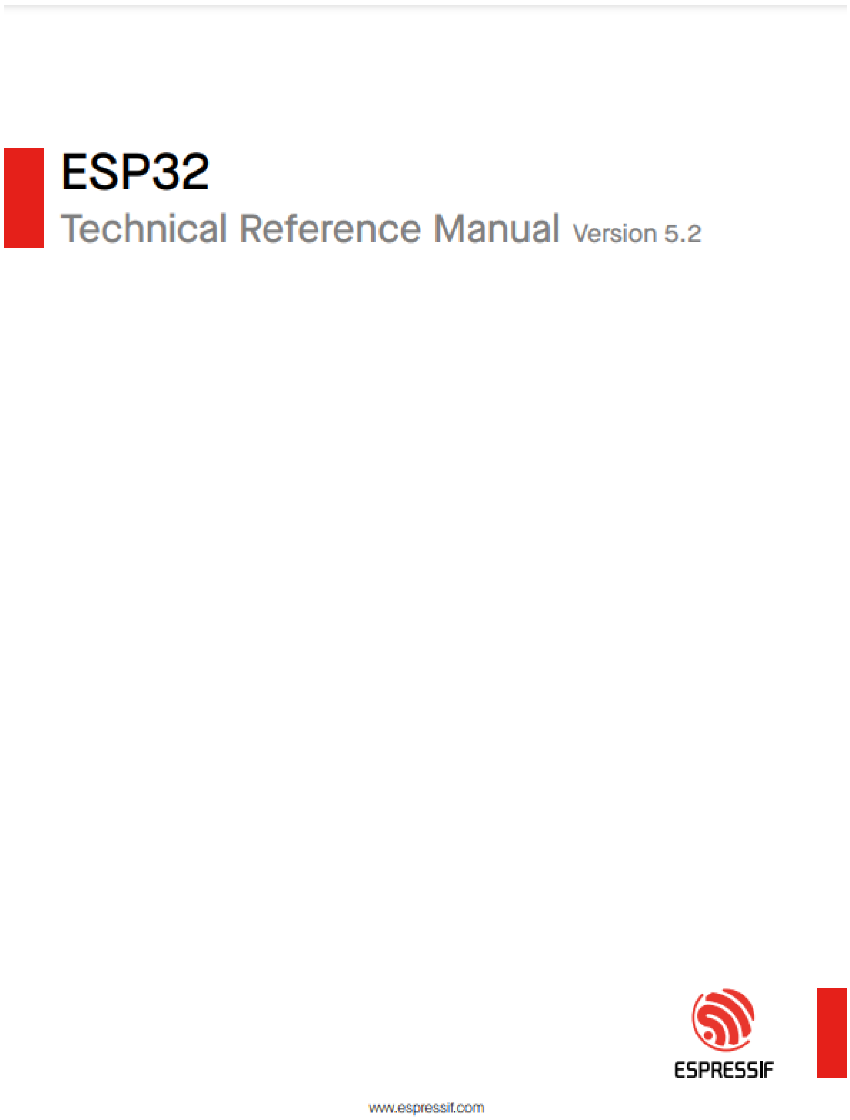
APPENDIX A: REFERENCE MATERIALS

.1 Referenced Documents

Below are the documents and resources utilized during the requirement elicitation phase, along with their respective front-page images for reference.

A.1 Technical Overview: ESP32 Microcontroller

- **Description**: This webpage provides technical specifications and integration details for the ESP32 microcontroller used in the project.
- **Source**: ESP32 Technical Reference Manual


ESP32

Technical Reference Manual Version 5.2

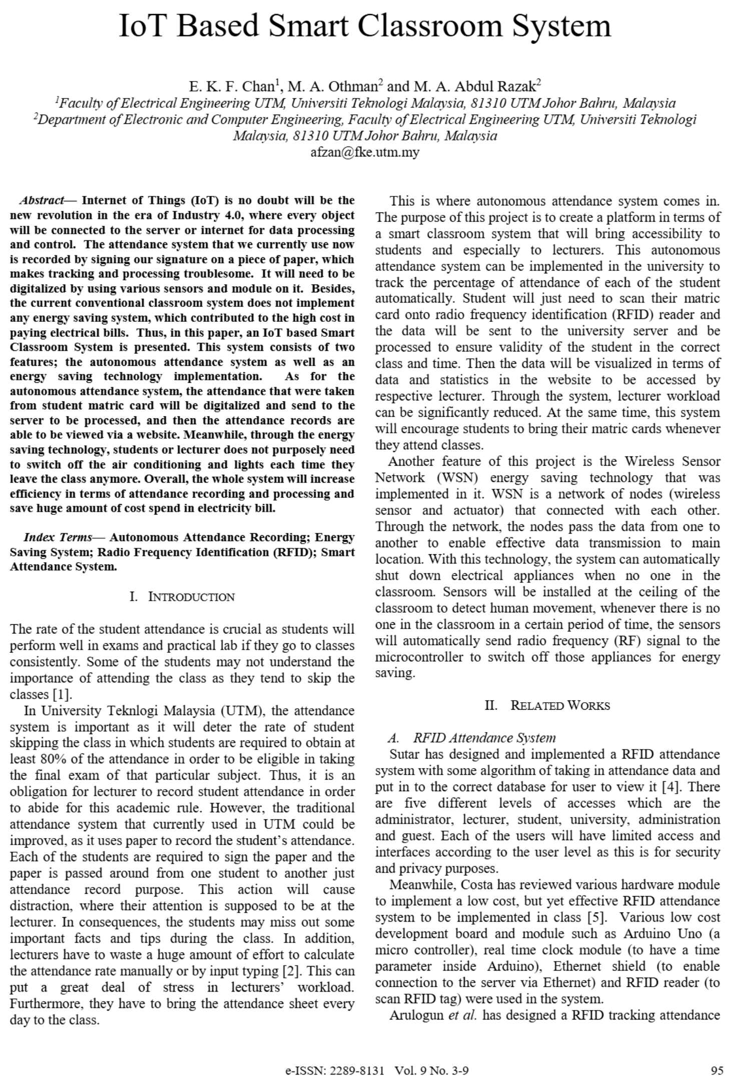
ESPRESSIF

www.espressif.com

Figure 1: Screenshot of ESP32 Technical References on Espressif Website

A.2 Article Paper: RFID Attendance System UTM

- **Description**: This article paper outlines the process on doing the RFID attendance tracking and also using motion detector.
- **Source**: Chan, E. K. F. and Othman, M. A. and Abdul Razak, M. A., IoT Based Smart Classroom System , 2017
- **Link**: Go to Article



IoT Based Smart Classroom System

E. K. F. Chan[1], M. A. Othman[2] and M. A. Abdul Razak[2]
[1]*Faculty of Electrical Engineering UTM, Universiti Teknologi Malaysia, 81310 UTM Johor Bahru, Malaysia*
[2]*Department of Electronic and Computer Engineering, Faculty of Electrical Engineering UTM, Universiti Teknologi Malaysia, 81310 UTM Johor Bahru, Malaysia*
afzan@fke.utm.my

***Abstract*— Internet of Things (IoT) is no doubt will be the new revolution in the era of Industry 4.0, where every object will be connected to the server or internet for data processing and control. The attendance system that we currently use now is recorded by signing our signature on a piece of paper, which makes tracking and processing troublesome. It will need to be digitalized by using various sensors and module on it. Besides, the current conventional classroom system does not implement any energy saving system, which contributed to the high cost in paying electrical bills. Thus, in this paper, an IoT based Smart Classroom System is presented. This system consists of two features; the autonomous attendance system as well as an energy saving technology implementation. As for the autonomous attendance system, the attendance that were taken from student matric card will be digitalized and send to the server to be processed, and then the attendance records are able to be viewed via a website. Meanwhile, through the energy saving technology, students or lecturer does not purposely need to switch off the air conditioning and lights each time they leave the class anymore. Overall, the whole system will increase efficiency in terms of attendance recording and processing and save huge amount of cost spend in electricity bill.**

***Index Terms*— Autonomous Attendance Recording; Energy Saving System; Radio Frequency Identification (RFID); Smart Attendance System.**

I. Introduction

The rate of the student attendance is crucial as students will perform well in exams and practical lab if they go to classes consistently. Some of the students may not understand the importance of attending the class as they tend to skip the classes [1].

In University Teknlogi Malaysia (UTM), the attendance system is important as it will deter the rate of student skipping the class in which students are required to obtain at least 80% of the attendance in order to be eligible in taking the final exam of that particular subject. Thus, it is an obligation for lecturer to record student attendance in order to abide for this academic rule. However, the traditional attendance system that currently used in UTM could be improved, as it uses paper to record the student's attendance. Each of the students are required to sign the paper and the paper is passed around from one student to another just attendance record purpose. This action will cause distraction, where their attention is supposed to be at the lecturer. In consequences, the students may miss out some important facts and tips during the class. In addition, lecturers have to waste a huge amount of effort to calculate the attendance rate manually or by input typing [2]. This can put a great deal of stress in lecturers' workload. Furthermore, they have to bring the attendance sheet every day to the class.

This is where autonomous attendance system comes in. The purpose of this project is to create a platform in terms of a smart classroom system that will bring accessibility to students and especially to lecturers. This autonomous attendance system can be implemented in the university to track the percentage of attendance of each of the student automatically. Student will just need to scan their matric card onto radio frequency identification (RFID) reader and the data will be sent to the university server and be processed to ensure validity of the student in the correct class and time. Then the data will be visualized in terms of data and statistics in the website to be accessed by respective lecturer. Through the system, lecturer workload can be significantly reduced. At the same time, this system will encourage students to bring their matric cards whenever they attend classes.

Another feature of this project is the Wireless Sensor Network (WSN) energy saving technology that was implemented in it. WSN is a network of nodes (wireless sensor and actuator) that connected with each other. Through the network, the nodes pass the data from one to another to enable effective data transmission to main location. With this technology, the system can automatically shut down electrical appliances when no one in the classroom. Sensors will be installed at the ceiling of the classroom to detect human movement, whenever there is no one in the classroom in a certain period of time, the sensors will automatically send radio frequency (RF) signal to the microcontroller to switch off those appliances for energy saving.

II. Related Works

A. RFID Attendance System

Sutar has designed and implemented a RFID attendance system with some algorithm of taking in attendance data and put in to the correct database for user to view it [4]. There are five different levels of accesses which are the administrator, lecturer, student, university, administration and guest. Each of the users will have limited access and interfaces according to the user level as this is for security and privacy purposes.

Meanwhile, Costa has reviewed various hardware module to implement a low cost, but yet effective RFID attendance system to be implemented in class [5]. Various low cost development board and module such as Arduino Uno (a micro controller), real time clock module (to have a time parameter inside Arduino), Ethernet shield (to enable connection to the server via Ethernet) and RFID reader (to scan RFID tag) were used in the system.

Arulogun *et al.* has designed a RFID tracking attendance

Figure 2: Front Page of IoT Based Smart Classroom System Article

A.3 Software Documentation: APIs Documentation

- **Description**: Software documentation for APIs and tools such as Google's Speech-to-Text API and Gemini AI API provides the technical specifications and usage guidelines on that. These documents were instrumental in integrating API of AI functionalities, such as lecture transcription and summarization, into the system.

- **Sources**:
  - Google's Speech-to-Text API Documentation
  - Gemini AI API Documentation

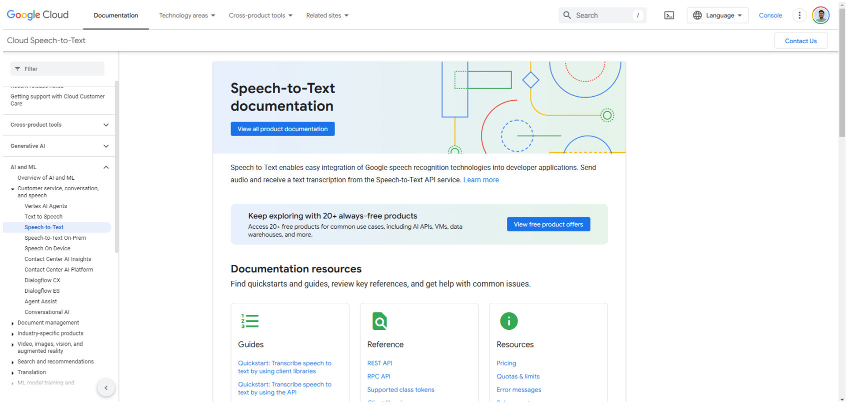


Figure 3: Print Screen on Speech-to-Text Google's API

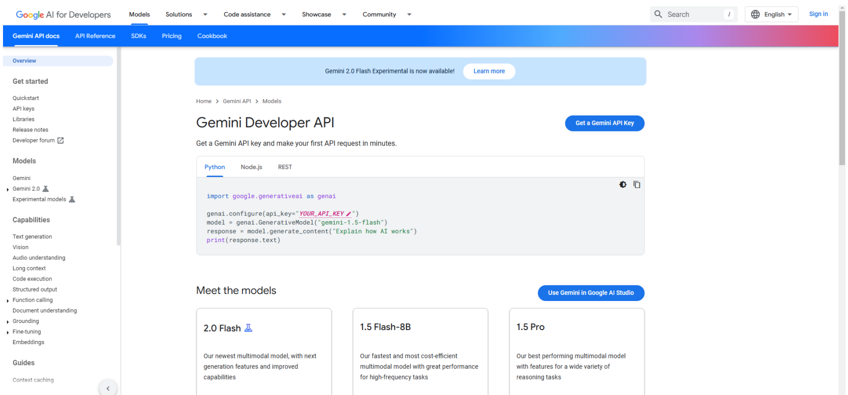


Figure 4: Print Screen on Gemini AI Google's API

APPENDIX B: STAKEHOLDER MEETINGS

.2 Meeting Details

Details of the meetings with stakeholders are summarized below:

- **Date**: 23 October 2024
- **Stakeholders**: Mr. Mohd Riezuan Razali (PPH Staff)
- **Agenda**: Gather the requirements of the system and explain a bit about the idea to make this system for the PPH.
- **Outcomes**: The requirements were collected such as PPH Staff want to control the utilities from a far to turn off it, utility problem can be reported through apps.

.3 Meeting Pictures

Figure 5: After finished meeting with Stakeholders on 23 Oct 2024

APPENDIX C: STAKEHOLDER QUESTIONS

.4 Questionnaire

Below are questions and responses from interactions with PPH Staff:

- **Question 1** : Is there any other system that similar to the system that I proposed to you now ?

  - **Answer**: No, not yet. We don't have system like this before.

- **Question 2** : Is there a plan to make such system in the UMT especially now or in the future ?

  - **Answer**: Yes, actually we once had a discussion on making a so called "SmartClass" like the features is can record the live class and student who can't attend the class , they can join the class on that platform. The class also were recorded and later student can watch it. That are basically the system that we had proposed, but we dont proceed on that proposed idea cause there is no needs on that systems now.

- **Question 3** : What do you think on this system if we can achieved to develop it and use it in the class ?

  - **Answer**: I think this is a good project or system if we can achieved to develop it cause it will give a lot of benefits right to the student especially right.

- **Question 4** : Do you suggest to build Apps based or Web based ?

  - **Answer**: Definitely App-based.

- **Question 5** : Are there any concerns about security when using this system (for example, who can control the features available on the app) ?

  - **Answer**: Yes, actually i would prefer for the control for features like control the utilities , i would prefer only PPH Staff and Lecturers can control it rather than letting the students can control it. Because , the probability on student misuse it is high right.

- **Question 6** : To what extent is automation needed? For example, should the lights be turned on automatically when someone enters the room?

  - **Answer**: I think it would be good if the utility like lights and etc will auto turn off if no person detected in the classroom. This can save the electricity right.

- **Question 7** : What are your views on this project?

  - **Answer**: This project is very good for now and the future. I would love to see this project works later.