

**Software Requirement Document (SRS)**

**IOT based-Biometric Attendance System**

**Version 1.0**

|  |  |
| --- | --- |
| **Project Code** | **20F-12** |
| **Supervisor** | Dr. Raheel Ahmed Memon |
| **Project Manager** | Farwa Shoail |
| **Project Team** | Farwa Shoail   1. Shareem Mughal   Faisal Junaid |
| **Submission Data** |  |

## Document History

|  |  |  |  |
| --- | --- | --- | --- |
| **Version** | **Name of Person** | **Date** | **Description of Change** |
| **1.0** | M.Shareem Mughal | 21-03-2024 | Create Document and Added scope |
| **1.0** | M.Shareem Mughal | 27-03-2024 | Added functional requirements |
| **1.0** | M.Shareem Mughal | 27-11-2023 | Added use cases and non-functional requirements. |

## Distribution List

|  |  |
| --- | --- |
| **Name** | **Role** |
| **Dr. Raheel Ahmed Memon** | Supervisor |
| **Farwa Sohail** | Project Manager |
| **M Shareem Mughal** | Group Member |
| **Faisal Junaid** | Group Member |

## Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Version** | **Sign-off Authority** | **Reason for change** | **Sign-off Date** |
| **1.0** | M.Shareem Mughal | Initial Draft |  |
|  |  |  |  |

## Document Sign-off

|  |  |  |  |
| --- | --- | --- | --- |
| **Version** | **Sign-off Authority** | **Project Role** | **Sign-off Date** |
| **1.0** | Dr. Raheel Memon | Supervisor |  |
| **1.0** | 1. Shareem Mughal | Group Member |  |
|  |  |  |  |
|  |  |  |  |

**Supervisor: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Project Manager: \_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Table of Contents**

[Document History 2](#_Toc151404236)

[Distribution List 2](#_Toc151404237)

[Revision History 3](#_Toc151404238)

[Document Sign-off 3](#_Toc151404239)

[1. Introduction 7](#_Toc151404240)

[1.1. Purpose of Document 7](#_Toc151404241)

[1.2. Intended Audience 7](#_Toc151404242)

[1.3. Document Conventions 8](#_Toc151404243)

[2. Overall System Description 9](#_Toc151404244)

[2.1. Project Background 9](#_Toc151404245)

[2.2. Project Scope 9](#_Toc151404246)

[2.3. Not in Scope 10](#_Toc151404247)

[2.4. Project Objectives 10](#_Toc151404248)

[2.5. Stakeholders 11](#_Toc151404249)

[2.6. Operating Environment 11](#_Toc151404250)

[2.7. System Constraints 11](#_Toc151404251)

[2.7.1 Software constraints: 12](#_Toc151404252)

[2.7.2 Hardware constraints: 12](#_Toc151404253)

[2.7.3 Cultural constraints: 12](#_Toc151404254)

[2.7.5 User constraints: 13](#_Toc151404255)

[2.8. Assumptions & Dependencies 13](#_Toc151404256)

[3. External Interface Requirements 14](#_Toc151404257)

[3.1. Hardware Interfaces 14](#_Toc151404258)

[3.2. Software Interfaces 14](#_Toc151404259)

[3.3. Communications Interfaces 16](#_Toc151404260)

[4. Functional Requirements 17](#_Toc151404261)

[4.1. Functional Hierarchy 17](#_Toc151404262)

[4.2. Use Cases 18](#_Toc151404263)

[4.3 Use Case: IOT based Biometric Attendance Management System 20](#_Toc151404264)

[5. Non-functional Requirements 40](#_Toc151404265)

[5.1. Performance Requirements 40](#_Toc151404266)

[5.1.1 Swift Interaction: 41](#_Toc151404267)

[5.1.2 Prompt Alerts: 41](#_Toc151404268)

[5.1.3 Efficient Data Handling: 41](#_Toc151404269)

[5.1.4 Reliability Assurance: 41](#_Toc151404270)

[5.1.5 Normalized Database: 41](#_Toc151404271)

[5.2. Safety Requirements 41](#_Toc151404272)

[5.2.1 Reliability 41](#_Toc151404273)

[5.2.2 Predicted Behavior 41](#_Toc151404274)

[5.2.3 Data Integrity 41](#_Toc151404275)

[5.2.4 Policy 42](#_Toc151404276)

[5.3. Security Requirements 42](#_Toc151404277)

[5.3.1 Encryption 42](#_Toc151404278)

[5.3.2 Secure Network Protocols 42](#_Toc151404279)

[6. References 42](#_Toc151404280)

[7. Appendices 42](#_Toc151404281)

**List of Tables**

[Table 1: Terms and Descriptions 8](#_Toc151404282)

[Table 2: Stakeholders and focus 11](#_Toc151404283)

[Table 3: Software Interfaces and Descriptions 15](#_Toc151404284)

[Table 4: Primary actors and Use cases 19](#_Toc151404285)

[Table 5: Sign up Use case 21](#_Toc151404286)

[Table 6: Login Use case 23](#_Toc151404287)

[Table 7: Veiw Attendance Use case 24](#_Toc151404288)

[Table 8: Generate Report Use case 24](#_Toc151404289)

[Table 9: View Announcements Use case 25](#_Toc151404290)

[Table 10: Mark Attendance Use case 26](#_Toc151404291)

[Table 11: View their Respective Attendance Use case 27](#_Toc151404292)

[Table 12: Announcement Use case 28](#_Toc151404293)

[Table 13: Attendance Management Use case 29](#_Toc151404294)

[Table 14: Manage profile Use case 29](#_Toc151404295)

[Table 15: Veiw Report Use case 30](#_Toc151404296)

[Table 16: Create Employee Profile Use case 31](#_Toc151404297)

[Table 18: Create Teacher Profile Use case 33](#_Toc151404298)

[Table 19: Report Button Use case 34](#_Toc151404299)

**List of Figures**

[Figure 1: Context Diagram 16](#_Toc150543222)

[Figure 2 Functional Hierarchy of System 18](#_Toc150543223)

[Figure 3 Use case Diagram of System 20](#_Toc150543224)

# Introduction In today's fast-evolving world, the demand for effective attendance management systems has become increasingly critical. The rapid advancement of Internet of Things (IoT) technology presents an opportunity to redefine attendance tracking. By harnessing IoT devices alongside biometric authentication methods, organizations can revolutionize attendance management. Our proposed solution, the "**IoT-Based Biometric Attendance Management System**" addresses prevalent challenges by integrating IoT devices with biometric authentication capabilities. Iris recognition enhances system efficacy, ensuring reliable individual identification across various applications. However, the existing Biometric attendance management systems play a pivotal role in various organizations, offering a seamless and efficient method for tracking employee attendance. However, despite their widespread adoption, several challenges hinder their optimal functionality and usability. One of the primary issues faced by biometric attendance management systems is the high costs associated with their components. The procurement of biometric devices, such as fingerprint scanners or facial recognition cameras, often demands a significant financial investment. Additionally, these systems frequently rely on **proprietary or specialized components,** making replacements costly and limiting options for cost-effective maintenance. Furthermore, the maintenance of biometric attendance management systems poses another challenge, as it often requires specialized expertise and resources. Regular calibration, software updates, and troubleshooting are essential for ensuring the **accuracy** and **reliability of these systems.** However, the associated expenses for maintenance activities can accumulate over time, straining organizational budgets and resources. Moreover, many biometric attendance management systems operate under **Service Level Agreements (SLAs)** that entail subscription fees for ongoing support and updates. These fees can be substantial and may present financial constraints for organizations. Addressing these challenges is crucial to enhancing the **accessibility**, **affordability**, and **effectiveness** of biometric attendance management systems. By finding solutions to reduce **the high costs of components**, **non-replaceable components,** **expensive maintenance**, and SLA subscription fees, organizations can optimize the functionality and value proposition of their attendance management systems. This project aims to explore innovative approaches to overcome these obstacles, ultimately facilitating the seamless integration and operation of biometric attendance management systems within various organizational contexts.

## 1.1. Purpose of Document

This document serves as a comprehensive guide outlining the software requirements for the development of the **IOT-Based Biometric Attendance System**. It defines the scope, features, and constraints of the system, providing a foundation for effective project planning, development, and evaluation. Its primary purpose is to ensure a shared understanding of the project among stakeholders and guide the development team throughout the software development life cycle.

## 1.2. Intended Audience

This document is intended for our project's key stakeholders, including Universities, Small Organiztions, Hospitals, others Educational institutions. The project supervisor will utilize the document to provide guidance and assessment, while team members will refer to it for technical details and requirements. Committee members will use the SRS to evaluate the project's progress and compliance. This document ensures effective communication and understanding among these stakeholders, facilitating the successful execution, validation and verification of the project.

## 1.3. Document Conventions

The Software Requirement Specification document is written in **Times New Roman** font of size 12, font of size 14 for Subheadings, font of size 16 for Headings. Each functional requirement has its own priority according to its priority mentioned.

Further, following terms are used throughout the document and in this context their description is provided as:

|  |  |
| --- | --- |
| **Term** | **Description** |
| Raspberry Pi | The Raspberry Pi is a very cheap computer that runs Linux, but it also provides a set of GPIO (general purpose input/output) pins, allowing you to control electronic components for physical computing and explore the Internet of Things (IoT). |
| Internet of Things(IOT) | The Internet of things describes devices with sensors, processing ability, software and other technologies that connect and exchange data with other devices and systems over the Internet or other communications networks. |
| Attendance Tracking | Real-time monitoring of Attendance and abscense. |
| Users | Different roles interacting with the system, including teacher, employees, staff members, e. |
| React | JavaScript library for building user interfaces, providing a declarative and efficient way to create interactive and dynamic web applications. |
| Node JS | JavaScript Runtime, used for backend development |
| SQL | Database |
| API | Application programming interface |

Table 1: Terms and Descriptions

# 2. Overall System Description

This section of the document explains the background, scope, objectives and other essential information of the System.

## 2.1. Project Background

This project needed because there is urgent need of cost effective and secure biometric attendance system which provide convienet features to empolyees and staff. So, we are creating a **IOT-based Attandance Management System** to fix these issues. This system will make it easier for teachers and students by using technology.we want to ensure cost-effective solution by leveraging Commercial Off-The-Shelf (COTS) component making it accessible to educational insititution and related organization. Futhurmore, This system also ensure secure transmission between components and local server to provide confidentiality and data protection. We will also be creating dashboard for the admin which provide features like user registration and report generation.

## 2.2. Project Scope

The project scope encompasses the development of a comprehensive IOT-based biometric attendance Management System, offering employees and instructors a centralized device for streamlined marking attendance. Here’s what it will do.

For ***employees and Teacher***, it is aimed to achieve the following features and none other than these:

* Easy user registration on web portal.
* Reduce registration time.
* On-hand Registration.
* After reviewing, user receive confirmation email.
* Able to see their attendance on web portal.
* Mark attendance with minial time and error.

For ***Admin,*** the System must achieve the following features and none other than these.

* Organize and Manage Attendance in database effectively.
* Interative Dashboard.
* Instant report generation.
* Creation of Profile.
* Profile Management.

## 2.3. Not in Scope

IOT-Based Biometric Attendance Management System has a specific focus, and certain things are not part of its plan. It won't include features that allow user to apply for leave. It also won't connect with systems outside the university, keeping things secure. Features unrelated to academic work are not on the list either. Plus, the system is meant only for use within the university, not outside.

## 2.4. Project Objectives

The objectives of our proposed system are:

* + Ensure cost-effectiveness by leveraging Commercial Off-The-Shelf (COTS) components.
  + Maintain high accuracy and high performance.
  + Transfer of data between components and server will be through encrypted means.
  + Establish an organized attendance management system.
  + Replace properitary Hardware which can not repaired or changed.
  + User able to register on the register or contact the admin for approval.
  + Develop an intuitive and user-friendly dashboard for administration.
  + Generate a report for an indiviual.

## 2.5. Stakeholders

|  |  |
| --- | --- |
| **Specific Stakeholder** | **Focus** |
| **Primary Stakeholder** | |
| **Sukkur Iba University** | Students will utilize the system to submit and manage final year project ideas, track progress, and streamline communication with supervisors |
| **Secondary Stakeholder** | |
| **Healthcare Facilites** | Project supervisors will utilize the system to oversee and guide students throughout the final year project process. |
| **Committee members** | Committee members will be involved in evaluation of projects and seeing progress of student projects. |
| **Administrators** | Administrators will be involved in maintaining the system, ensuring its smooth operation, and managing access and permissions. |
| **University Management** | University management will have an interest in the efficient functioning of the system to enhance the overall final year project management process. |

Table 2: Stakeholders and focus

## 2.6. Operating Environment

The System will operate with a combination of following environments.

* **User Interface:** Operates on web-based platforms for accessibility on laptops, tablets, and smartphones.
* **Database:** SQL Database.
* **Development Frameworks:** Utilizes modern development frameworks such as ReactJS for the frontend and Node.js for the backend.
* **Responsive Design:** Adopts a responsive design for a consistent user experience across various devices

## 2.7. System Constraints

The final-year project management system operates within specific constraints, critical for its development and functionality. Failing to adhere to these constraints may impede the system's performance and effectiveness.

### 2.7.1 Software constraints:

* **Compatibility with Existing Systems:** The software should integrate seamlessly with current systems and databases.
* **Database System:** The system relies on an SQL database for data storage and retrieval.
* **Version Control System:** GIT is crucial for effective version control, facilitating collaborative development.
* **Regular Updates and Maintenance:** Ongoing updates and maintenance are vital for system reliability and performance.

### 2.7.2 Hardware constraints:

* **Internet Connection Stability:** A stable and reliable internet connection is essential for accessing and using the system. Users in areas with limited connectivity may experience disruptions, affecting their ability to use the system effectively.
* **Device Compatibility**: The system should be accessible on a small lcd so it's important to optimize the user interface for smaller screens to ensure a seamless user experience.
* **Availability of Compatible Hardware:** Ensuring hardware supports iris recognition and fingerprint sensor.

### 2.7.3 Cultural constraints:

* **Language Constraint:** This proposed system will be available only in the English language.
* **Legal and Regulatory Considerations:** Compliance with laws and regulations regarding biometric data handling is essential.

**2.7.4 Environmental constraints:**

* **Internet Connectivity Dependency:** The system relies on a stable internet connection for its operation. Users must have consistent access to the internet to use the system effectively.
* **Effects of Environmental Factors:** Factors like dust or humidity can impact the performance of hardware.
* **Impact of Lighting Conditions:** Poor lighting can affect the accuracy of iris recognition.
* **Availability of Power Sources:** Ensuring IoT devices have reliable power sources in various settings.

### 2.7.5 User constraints:

* **Familiarity and Comfort with Biometric Technology:** Users may be hesitant or unfamiliar with using biometric methods for attendance.
* **Accessibility for Users with Disabilities:** The system should accommodate users with disabilities for inclusivity.
* **Instruction Requirements:** Users may need instruction to understand how to use the system effectively.

## 2.8. Assumptions & Dependencies

For the IOT-based Attendance Management System, we have the following dependencies.

* **Reliable Internet Access:** Successful system implementation assumes that users have reliable internet access. The system's functionality and accessibility rely on a stable internet connection.
* **Web Browser Compatibility:** The system's performance is dependent on web browser compatibility and updates. Users are expected to use up-to-date web browsers to ensure an optimal user experience.
* **External Services:** System functionality is dependent on the availability and support of external services, such as APIs, third-party integrations, and external databases. The system relies on these services for various features and data retrieval.
* **Security Compliance:** Users are expected to adhere to security measures to maintain account and data integrity. This includes following best practices for password security and account protection.
* **Timely User Interactions:** System responsiveness relies on timely user interactions and communication. Users should engage with the system promptly to ensure effective project management and communication.

# 3. External Interface Requirements

In this section of the document, a comprehensive description is provided for hardware, software and communication interfaces and their relation to achieve the expected behavior of the system.

## 3.1. Hardware Interfaces

|  |  |
| --- | --- |
| **Hardware Interface** | **Description** |
| **Raspberry Pi** | Raspberry pi will act as central controller for other hardware component like camera and fingerprint sensor and directly intract with server. |
| **Raspberry Pi Camera Module 2 NoIR** | Noir V2.1 camera module will be used for image acquisition of iris. |
| **Fingerprint Sensor** | Fingerprint Sensor will be use to scan finger prints for authentication |
| **Servo Motor** | Servo motor will be use to move the camera to find the iris or face. |
| **Connecting Cable** | Cable will be use connect other hardware component to Raspberry pi |
| **LCD** | LCD will be use to display the main software interface where user interact with device |
| **SD Card** | SD card will act as storage for Raspberry pi (Operating System) |

*Table 3: Hardware Interfaces and Descriptions*

The IOT-based Biometric Attandance Management System is user-friendly and adaptable to standard computing devices like laptops, tablets, or smartphones. You can use the system with a basic buttons attach to device , and it efficiently processes data without putting a heavy load on device. the system's design ensures a smooth and responsive interface.

Along with that, database and application logic will be managed separately and will be hosted on a third-party service provider.

## 3.2. Software Interfaces

There will be multiple required Software Interfaces, tools and libraries for System to operate.

|  |  |
| --- | --- |
| **Software Interface** | **Description** |
| **JavaScript** | The scripting language employed for dynamic and interactive elements in the IOT-based Biometric Attendance Management System. |
| **ReactJS** | A JavaScript library utilized for building a responsive and user-friendly frontend interface in the project |
| **MySQL** | The relational database management system employed to store and manage structured data within the project |
| **Node.js** | The server-side runtime environment facilitating efficient server operations and interactions in the Final Year Project Management System. |

Table 4: Software Interfaces and Descriptions

The IOT-based Biometric Attendance Management System is designed to seamlessly integrate with standard software environments commonly used in educational institutions. The device leverages **JavaScript** for dynamic and interactive features, and it is built using the **ReactJS** framework for the frontend, providing a modern and responsive user interface. The backend is developed using Node.js, facilitating efficient server-side operations. Data is stored and managed using **MySQL** as the relational database management system, ensuring a robust and structured storage mechanism. The system's reliance on widely used and established technologies enhances its interoperability and ease of integration with existing software infrastructures.

It also utilizes a server-side architecture. The backend of the system is developed using **Node.js**, which is a runtime environment for executing JavaScript code on the server. Node.js enables efficient handling of server-side operations and supports asynchronous, non-blocking I/O, making it well-suited for scalable and real-time applications.

## 3.3. Communications Interfaces

To facilitate secure communication between a Raspberry Pi and a server via TCP/IP, RSA encryption is employed. Initially, the Raspberry Pi generates a pair of cryptographic keys: a public key for encryption and a private key for decryption. When transmitting data to the server, the Raspberry Pi encrypts the information using the server's public key, ensuring that only the intended recipient can decipher it. This encrypted data is then sent over a TCP/IP connection to the server. Upon receiving the encrypted message, the server utilizes its private key to decrypt the data, thus maintaining confidentiality during transmission. This robust encryption scheme ensures the integrity and confidentiality of the communication between the Raspberry Pi and the server, safeguarding sensitive information from unauthorized access or tampering. Additionally, a stable and reliable internet connection is essential for achieving the optimal performance of the system.

The Context Diagram for the system is shown below, which is drawn using draw.io (Draw.io, n.d.). It contains external entities, Software Interfaces, Services and Data flow.

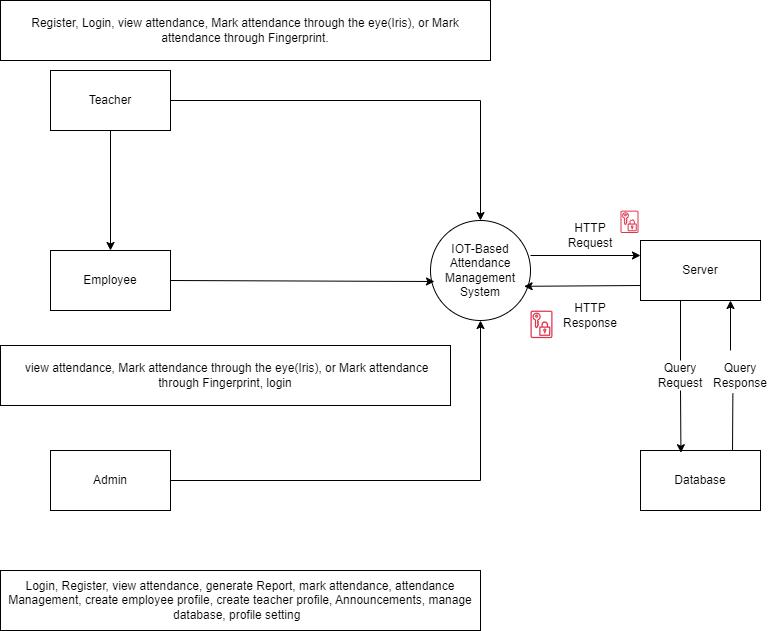


Figure 1: Context Diagram of IOT-Based Biometric Attendance Management System

# 4. Functional Requirements

+This section deeply explains the functional requirements for the system.

## 4.1. Functional Hierarchy

All possible features of System are listed in below diagram, drawn using draw.io (Draw.io, n.d.).

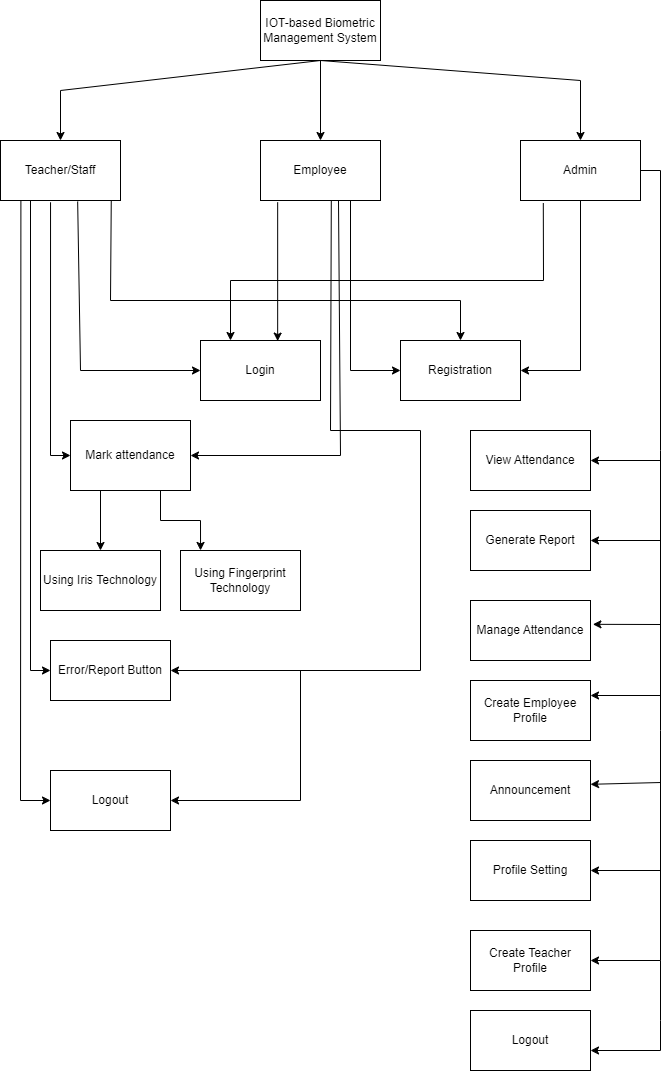


Figure 2 Functional Hierarchy of System

## 4.2. Use Cases

There will be three users of the system. The use cases are as below.

|  |  |  |
| --- | --- | --- |
| **S/No** | **Primary Actor** | **Use case** |
| **1** | **Teacher/Faculty** | Registration |
| Login |
| Authentication method selection |
| Mark Attendance through Fingerprint |
| Mark Attendance through Iris |
| Report button |
| Verification message |
| **2** | **Employee** | Registration |
| Login |
| Authentication method selection |
| Mark Attendance through Fingerprint |
| Mark Attendance through Iris |
| Report button |
| Verification message |
| **3** | **Admin** | Signup/Registration |
| Login |
| Create Employee Profile |
| Create Teacher Profile |
| Attendance Management |
| Profile Setting |
| Report Generation |
| Announcement |
| View Employee Attendance |
| View Teacher/Faculty Attendance |

Table 4: Primary actors and Use cases

## 4.3 Use Case:

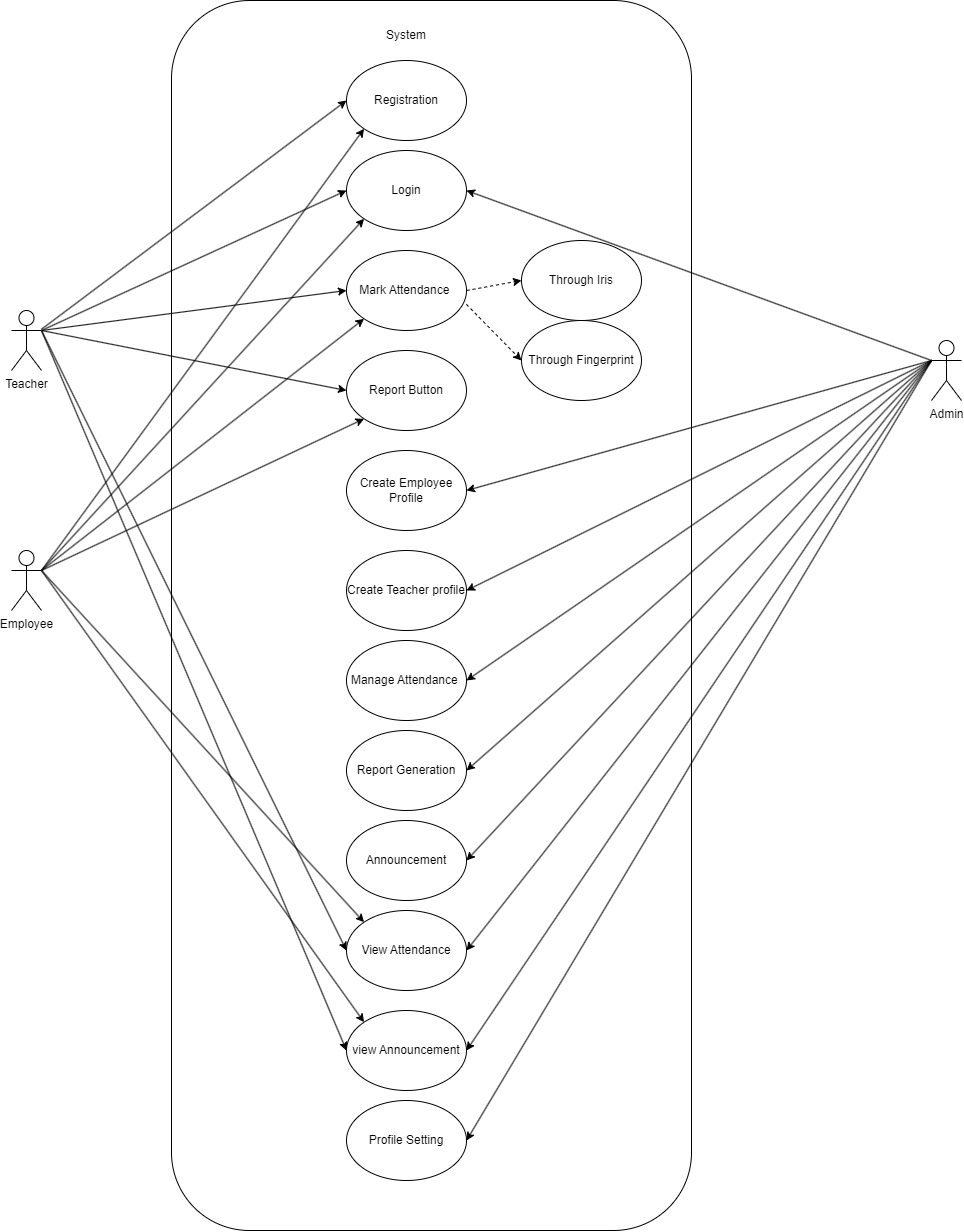


Figure 3 Use case Diagram of System

**4.3.1 Sign up**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **UC-01: Sign up** | | | | |
| **Use case Id:** | | UC-01 | | |
| **Actors:** | | Employee, Teacher | | |
| **Feature:** | | Authentication | | |
| **Pre-condition:** | | User must be connected to the internet  User should give the information as per requirement | | |
| **Scenarios** | | | | |
| **Step#** | **Action** | | | **Software Reaction** |
| **1.** | User opens the Website | | | The system will display the options to sign up |
| **2.** | The Users request for sign up | | | The system will display the required fields to fill |
| **3.** | The users should enter information in the required fields | | | The system will store the information in the database |
| **Alternate Scenarios:** | | | | |
| **1a:** User should fill all the required fields, if not then 2b  **1b:** User is asked to refill all the required fields | | | | |
| **Post Conditions** | | | | |
| **Step#** | **Description** | | | |
| **1** | User is registered into the system and can use the extended features of system | | | |
| **2** | User has an account | | | |
| **Use Case Cross referenced** | | | None | |

Table 5: Sign up Use case

**4.3.2 Login**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **UC-02: Login** | | | | |
| **Use case Id:** | | UC-02 | | |
| **Actors:** | | Employee, Admin, Teacher | | |
| **Feature:** | | Authentication | | |
| **Pre-condition:** | | Users should sign up for the system  User must be connected to the internet  Users should have an authentic and registered account in the system | | |
| **Scenarios** | | | | |
| **Step#** | **Action** | | | **Software Reaction** |
| **1.** | The User enters the registered credentials | | | The system verifies the user’s login email and password |
| **2.** | The User clicks the login button | | | The system will display overall view of the system |
| **Alternate Scenarios:** | | | | |
| **1a:** User should fill all the required fields, if not then 1b  **1b:** User is asked to refill all the required fields  **2a:** If user does not provide the registered login email and password then 2b  **2b:** User is asked to try again  **3a:** If user forgets the password, then 3b  **3b:** The system will allow user to create a new password | | | | |
| **Post Conditions** | | | | |
| **Step#** | **Description** | | | |
| **1** | The user is logged into the system | | | |
| **2** | The user has access to the functions of the system | | | |
| **Use Case Cross referenced** | | | None | |

Table 6: Login Use case

**4.3.6 View Attendance**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **UC-06: View Attendance** | | | | | |
| **Use case Id:** | | UC-06 | | | |
| **Actors:** | | | Teacher, Employee,Admin | | |
| **Feature:** | | | Admin and Students can view attendance | | |
| **Pre-condition:** | | User must be logged into the system | | | |
| **Scenarios** | | | | | |
| **Step#** | **Action** | | | | **Software Reaction** |
| **1.** | User click on view attendance on Sidebar | | | | System will show attendance of students. |
| **Post Conditions** | | | | | |
| **Step#** | **Description** | | | | |
| **1** | User can view attendance | | | | |
| **Use Case Cross referenced** | | | | Login | |

Table 7: View Attendance Use case

**4.3.7 Generate Report**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **UC-07: Generate Reports** | | | | | |
| **Use case Id:** | | UC-07 | | | |
| **Actors:** | | | Admin | | |
| **Feature:** | | | Admin generate Employee/Teacher attendance reports | | |
| **Pre-condition:** | | User must be logged into the system | | | |
| **Scenarios** | | | | | |
| **Step#** | **Action** | | | | **Software Reaction** |
| **1.** | User click on generate button | | | | System will generate attendance of particular Employee/Teacher |
| **Post Conditions** | | | | | |
| **Step#** | **Description** | | | | |
| **1** | System will show whether the file generated successfully or not. | | | | |
| **Use Case Cross referenced** | | | | Login | |

Table 8: Generate report Use case

**4.3.8 View Announcement**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **UC-08: View Announcement** | | | | | |
| **Use case Id:** | | UC-08 | | | |
| **Actors:** | | | Admin, Teacher, Employee | | |
| **Feature:** | | | View the announcements such as faulty Component | | |
| **Pre-condition:** | | User must be logged into the system | | | |
| **Scenarios** | | | | | |
| **Step#** | **Action** | | | | **Software Reaction** |
| **1.** | User will see announcements on the top of the screen | | | | System will show all the announcements from latest to oldest |
| **Post Conditions** | | | | | |
| **Step#** | **Description** | | | | |
| **1** | The user will get a list of announcements | | | | |
| **Use Case Cross referenced** | | | | Login | |

Table 9: View Announcements Use case

**4.3.9 Mark Attendance**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **UC-09: Mark Attendance** | | | | | |
| **Use case Id:** | | UC-09 | | | |
| **Actors:** | | | Empolyee, Teacher | | |
| **Feature:** | | | User will mark attendance through Iris or through Fingerprint | | |
| **Pre-condition:** | | Device will be turned on | | | |
| **Scenarios** | | | | | |
| **Step#** | **Action** | | | | **Software Reaction** |
| **1.** | User will select the options if user wants to mark attendance through iris or fingerprint. | | | | The system will display two options |
| **2.** | User will place thumb on the device or place his Iris based on the option user selected. | | | | The system will show the screen of Iris or Fingerprint. |
| **3.** | Upon succession, the user will verification message | | | | The system will display message if the attendance is marked or not. |
| **Post Conditions** | | | | | |
| **Step#** | **Description** | | | | |
| **1** |  | | | | |
| **Use Case Cross referenced** | | | | Login | |

Table 10: Mark attendance Use case

**4.3.10 View their respective Attendances**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **UC-10: View Their Respective Attendances** | | | | |
| **Use case Id:** | | UC-10 | | |
| **Actors:** | | Employee, Teacher | | |
| **Feature:** | | User will be able to see their respective Attendance | | |
| **Pre-condition:** | | User must be logged into the system | | |
| **Scenarios** | | | | |
| **Step#** | **Action** | | | **Software Reaction** |
| **1.** | User select View Attendance option from sidebar | | | The system will display their attendance. |
| **Alternate Scenarios:** | | | | |
| **1a:** User can select specific date to display attendance of that date, if not then 1b  **1b:** System will show all attendance of all students in all dates. | | | | |
| **Post Conditions** | | | | |
| **Step#** | **Description** | | | |
| **1** | The user will be able to see the attendance of all. | | | |
| **Use Case Cross referenced** | | | Login | |

Table 11: View All Attendance Use case

**4.3.11 Announcement**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **UC-11: Announcement** | | | | |
| **Use case Id:** | | UC-11 | | |
| **Actors:** | | Admin | | |
| **Feature:** | | User will be able to make a new announcement | | |
| **Pre-condition:** | | User must be logged into the system | | |
| **Scenarios** | | | | |
| **Step#** | **Action** | | | **Software Reaction** |
| **1.** | User click on Announcement | | | The system will list of previous announcements and option to make a new announcement. |
| **2.** | User click on Make New Announcement button | | | The system will show form for announcement |
| **3.** | User click on Create Announcement | | | The system will create a new announcement which will be available to audience. |
| **Post Conditions** | | | | |
| **Step#** | **Description** | | | |
| **1** | The user will be able to see new announcement available | | | |
| **Use Case Cross referenced** | | | Login | |

Table 12: Announcement feature Use case

**4.3.12Attendance Management**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **UC-12: Attendance Management** | | | | |
| **Use case Id:** | | UC-12 | | |
| **Actors:** | | Admin | | |
| **Feature:** | | Add or Delete or update | | |
| **Pre-condition:** | | User must be logged into the system | | |
| **Scenarios** | | | | |
| **Step#** | **Action** | | | **Software Reaction** |
| **1.** | User clicks on the attendance Management in sidebar | | | The system will display a list of attendance of teacher/employee |
| **2.** | User click on mark attendance | | | System will display a form to add attendance |
| **3.** | User click on edit | | | The system will display a form to make changes to already existed. |
| **4.** | User click on delete | | | The system will delete batch from Database and UI. |
| **Alternate Scenarios:** | | | | |
| **1a:** User click on Add else 1b.  **1b:** User click on Edit else 1c.  **1c:** User click on Delete batch. | | | | |
| **Post Conditions** | | | | |
| **Step#** | **Description** | | | |
| **1** | The user will see a new maarked attendance. | | | |
| **2** | The user will see new version of already existed. | | | |
| **3** | The user will see marked is removed from system. | | | |
| **Use Case Cross referenced** | | | Login | |

Table 13: Attendance Management Use case

**4.3.13 Manage profile**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **UC-13: Manage profile** | | | | |
| **Use case Id:** | | UC-13 | | |
| **Actors:** | | Admin | | |
| **Feature:** | | The user will be able to change its profile information. | | |
| **Pre-condition:** | | User must be logged into the system | | |
| **Scenarios** | | | | |
| **Step#** | **Action** | | | **Software Reaction** |
| **1.** | User clicks on the manage profile. | | | The system will display the profile of the user |
| **2.** | User can edit his/her profile and save the updated information. | | | The system will update the information in the database. |
| **Alternate Scenarios:** | | | | |
| **1a:** The user must save the information after updating, if not then  **1b:** The information will not be updated and the user has to update it again | | | | |
| **Post Conditions** | | | | |
| **Step#** | **Description** | | | |
| **1** | The user will view his/her profile | | | |
| **2** | The user will edit the information in his/her profile | | | |
| **Use Case Cross referenced** | | | Login | |

Table 14: Manage profile Use case

**4.3.14 View Report**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **UC-14: View Report** | | | | |
| **Use case Id:** | | UC-14 | | |
| **Actors:** | | Admin | | |
| **Feature:** | | The user will be able to view reports submitted by Teacher or employee | | |
| **Pre-condition:** | | User must be logged into the system | | |
| **Scenarios** | | | | |
| **Step#** | **Action** | | | **Software Reaction** |
| **1.** | User clicks on View Report in sidebar | | | The system will display a list of reports submitted by teacher/Employee |
| **Post Conditions** | | | | |
| **Step#** | **Description** | | | |
| **1** | User can see the reports | | | |
| **Use Case Cross referenced** | | | Login | |

Table 15: View Report Use case

**4.3.15 Create Employee Profile**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **UC-15: Create Employee Profile** | | | | |
| **Use case Id:** | | UC-15 | | |
| **Actors:** | | Admin | | |
| **Feature:** | | The user will be able to create employee profile | | |
| **Pre-condition:** | | User must be logged into the system.  Employee has already put their details on the web portal. | | |
| **Scenarios** | | | | |
| **Step#** | **Action** | | | **Software Reaction** |
| **1.** | User clicks on create profile | | | The system will show the UI to create the profile |
| **2.** | User will select the option Employee profile | | | The system will give two options create employee or teacher profile. |
| **3.** | User will fetch information from database. | | | The system UI will be show field to create profile. |
| **4.** | User will click on done then profile will be created. | | | The system will show message |
| **Post Conditions** | | | | |
| **Step#** | **Description** | | | |
| **1** | Employee/Teacher can see their profile. | | | |
| **Use Case Cross referenced** | | | Login | |

Table 16: Create Employee Profile use case

**4.3.16 Create Teacher Profile**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **UC-16: Create Teacher Profile** | | | | |
| **Use case Id:** | | UC-16 | | |
| **Actors:** | | Admin | | |
| **Feature:** | | The user will be able to create Teacher profile | | |
| **Pre-condition:** | | User must be logged into the system | | |
| **Scenarios** | | | | |
| **Step#** | **Action** | | | **Software Reaction** |
| **1.** | User clicks on create profile | | | The system will show the UI to create the profile |
| **2.** | User will select the option Teacher profile | | | The system will give two options create employee or teacher profile. |
| **3.** | User will fetch information from database. | | | The system UI will be show field to create profile. |
| **4.** | User will click on done then profile will be created. | | | The system will show message |
| **Post Conditions** | | | | |
| **Step#** | **Description** | | | |
| **1** | Employee/Teacher can see their profile on website portal | | | |
| **Use Case Cross referenced** | | | Login | |

Table 17: Create Teacher Profile use case

**4.3.17 Report Button**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **UC-17: Report Button** | | | | |
| **Use case Id:** | | UC-17 | | |
| **Actors:** | | Teacher, Employee | | |
| **Feature:** | | The user will able to report if there is fault in the device or not able to mark attendance. | | |
| **Pre-condition:** | | Device must be ON. | | |
| **Scenarios** | | | | |
| **Step#** | **Action** | | | **Software Reaction** |
| **1.** | User clicks on Report Button | | | The system will take you another interface  And show text field with some radio options. |
| **2.** | User put his id on that text field. | | | The system will show some possible option why error occur. |
| **3.** | User click on done | | | The system will ping the request to the Admin to address the issue. |
| **Post Conditions** | | | | |
| **Step#** | **Description** | | | |
| **1** |  | | | |
| **2** |  | | | |
| **Use Case Cross referenced** | | | Login | |

Table 18: Report Button use case

# 5. Non-functional Requirements

This section outlines all the non-functional requirements that are intangible but will highly affect the system’s usability and acceptance.

## 5.1. Performance Requirements

The Final Year Project Management System is designed to be interactive, responsive, and user-friendly. It prioritizes efficient performance, and the following non-functional requirements outline specific performance benchmarks:

### 5.1.1 Swift Interaction:

The system ensures a quick response time of 0.1 seconds for user interactions, maintaining a seamless experience.

### 5.1.2 Prompt Alerts:

Alerts and pop-ups appear within 2 seconds, enhancing the user interface.

### 5.1.3 Efficient Data Handling:

Asynchronous processes for data uploading and retrieval ensure optimal system performance.

### 5.1.4 Reliability Assurance:

High system reliability instills user confidence in interaction, ensuring minimal downtime or errors.

### 5.1.5 Normalized Database:

Database normalization prevents redundant data, optimizing storage and retrieval efficiency.

## 5.2. Safety Requirements

Safety requirements should be taken care of and users must be warned of any possible loss. Following safety requirements must be considered:

### 5.2.1 Reliability

The website should not Crash while being in use.

### 5.2.2 Predicted Behavior

Appropriate warning messages must be displayed in case of DELETE, CONFIRM and such actions.

### 5.2.3 Data Integrity

User account and preferences must be prevented and data integrity must suffice.

### 5.2.4 Policy

The System must have Terms and Conditions described in easy-to-read text and user agreement shall be approved for offering services.

## 5.3. Security Requirements

The Security threats can occur as the system will highly rely on Network communication. So these must be addressed to provide privacy and data integrity. Following requirements must be fulfilled by the System:

### 5.3.1 Encryption

System must use standard Encryption techniques to secure the data.

### 5.3.2 Secure Network Protocols

Secure network protocols such as HTTP over TCP/IP must be used.

# 6. References

Draw.io. (n.d.). *Draw.io*. From Draw.io: https://app.diagrams.net/

# 7. Appendices

Not applicable.