

UNIVERSITY OF BUEA



FACULTY OF ENGINEERING AND TECHNOLOGY
DERPARTMENT OF COMPUTER ENGINEERING

TASK 3: REQUIREMENT ANALYSIS

COURSE CODE: CEF 440

COURSE TITLE: INTERNET PROGRAMMING and MOBILE
PROGRAMMING

COURSE INSTUCTOR: DR NKEMENI Valery

BY GROUP 17

DATE: May 2025

SN	NAME	MATRICULE
1	AKENJI FAITH SIRRI	FE22A142
2	DYL PADARAN AMBE MUNJO	FE22A193
3	KONGNYU DESCHANEL	FE22A234
4	NDI BERTRAND	FE22A252
5	NDUKIE EBOKE BLANDINE	FE22A254

Contents

Software Requirements Specification (SRS)	1
1. Introduction	1
1.1 Purpose	1
1.2 Project Scope	1
1.3 Intended Audience and Reading Suggestions	1
1.4 Document Conventions	1
This document uses the following conventions	1
GPS	1
Global Positioning System.....	1
FR.....	1
Functional Requirement.....	1
NFR.....	1
Non-Functional Requirement.....	1
DB	1
Database.....	1
UI	1
User Interface.....	1
2. Overall Description	2
2.1 Product Perspective	2
2.2 Product Functions	2
2.3 User Classes and Characteristics	2
2.4 Operating Environment	2
2.5 Design and Implementation Constraints	2
2.6 Assumptions and Dependencies	3
3. Requirements	3
3.1 Functional Requirements	3
3.2 Non-Functional Requirements	4
4. External Interface Requirements	5
4.1 User Interfaces	5
4.2 Hardware Interfaces	5
4.3 Software Interfaces	5
4.4 Communications Interfaces	5

• Internet Connectivity: Required for real-time data synchronization.	5
5. Requirement Priority	5
Must Have (M)	5
Should Have (S)	5
Could Have (C).....	6
Won't Have (W).....	6
6. Feasibility Studies.....	7
6.1 Timing Conflicts with Facial Recognition Trigger	7
6.2 Disruptions in Attendance Timing.....	7
6.3 Student Dismissal by Lecturer.....	8
6.4 Valid Absence from Lecture	8
6.5 Network Dependency	8
Example in our situation:	9
6.6 Device Compatibility	9
6.6 False Positives/Negatives in Facial Recognition.....	9
7. Conclusion	10
7.1 Review and Analysis of Requirements	10
7.2 Validation with Stakeholders	11

Software Requirements Specification (SRS)

1. Introduction

1.1 Purpose

To design and implement a mobile-based attendance management system that uses facial recognition and geofencing technologies to ensure quick, secure, and accurate recording of student attendance in real-time, while also enabling both students and instructors to monitor attendance status efficiently.

1.2 Project Scope

The system aims to:

- Automate attendance recording using facial recognition.
- Ensure students are within designated areas using geofencing.
- Provide real-time attendance data to administrators and educators.

1.3 Intended Audience and Reading Suggestions

This document is intended for:

- **Developers:** To understand the system's functionalities and constraints.
- **Project Managers:** To oversee the development process.
- **Stakeholders:** To ensure the system meets organizational needs.
- **Testers:** To design test cases based on requirements.

1.4 Document Conventions

This document uses the following conventions

GPS	Global Positioning System
FR	Functional Requirement
NFR	Non-Functional Requirement
DB	Database
UI	User Interface

2. Overall Description

2.1 Product Perspective

The system is a standalone mobile application compatible with Android and iOS platforms, integrated with backend services for data storage and processing.

2.2 Product Functions

- **User Authentication:** Secure login for students and staff.
- **Facial Recognition:** Capture and verify student identities.
- **Geofencing:** Validate student location during attendance.
- **Attendance Records:** Maintain and display attendance logs.
- **Notifications:** Alert users about attendance status.

2.3 User Classes and Characteristics

- **Students:** Use the app to mark attendance.
- **Educators:** Monitor and manage class attendance.
- **Administrators:** Oversee system operations and data management.

2.4 Operating Environment

- **Mobile Platforms.** Android/Desktop, more configuration for iOS later on as most users are Android users and for the ones with iPhones, a huge chunk have Android as a second)
- **Backend:** Cloud-based services (e.g., Firebase) and SQL MySQL with links to API services
- **Database:** NoSQL database for secondary authentication possibly, SQL Database for Role based Management and

2.5 Design and Implementation Constraints

- Compliance with data protection regulations.
- Limited device resources (battery, storage).
- Network dependency for real-time operations.

2.6 Assumptions and Dependencies

- Users have compatible devices.
- Stable internet connection for real-time features.

3. Requirements

3.1 Functional Requirements

- **FR1: User Authentication:** The system shall authenticate users using secure credentials
- **FR2: Facial Data Storage:** The system will securely capture and store facial recognition data to ensure user privacy and data protection.
- **FR3: Camera Activation Timing:** The camera will capture live feeds and is prompted once 70% of the lecture time has elapsed.
- **FR4: Location Verification:** The system will confirm that users are present within specific, predefined geographic areas (geofence)
- **FR5: Attendance Recording:** Attendance will be automatically recorded after verifying both the user's facial identity and their location.
- **FR6: Attendance Management for Educators:** Educators will have the ability to view and download attendance records.
- **FR7: Class Notifications:** The system will send notifications to alert users when a class is about to start or stop.
- **FR8: Class Time Reset:** Lecturers can reset the class start and stop time if necessary.
- **FR9: Attendance Analytics:** Students and staff will be able to view detailed analytics regarding attendance trends and patterns.
- **FR10: Triggering Facial Recognition:** Lecturers can initiate the facial recognition process at their discretion.
- **FR11: Filtering Attendance Records:** Users can filter attendance records based on specific criteria, such as course, date, or student.
- **FR12: Geofence Management:** The system will allow administrators to create, modify, and manage geofenced areas.

- **FR13: Course Management:** The system should be capable of permitting students to register or drop courses. Admins should also be capable of creating courses and assigning lecturers to created course
- **FR14: Attendance Override:** Lecturers can manually override attendance records if needed.
- **FR15: Bluetooth Beckoning:** The system will support Bluetooth functionality to facilitate user interaction or notifications.

3.2 Non-Functional Requirements

- **NFR1: Accuracy:** These requirements guarantee the system correctly identifies student locations within geofences and records attendance data precisely, minimizing errors in tracking and data recording. It focuses on the precision and correctness of data.
- **NFR2: Reliability and Efficiency:** These requirements ensure the system is dependable and consistently available when needed, minimizing downtime and ensuring continuous operation while minimizing resource usage such as battery and provides efficient response time.
- **NFR3: Security:** This requirement focuses on protecting sensitive student data such as location and system access from unauthorized individuals, ensuring privacy, and preventing misuse of information.
- **NFR4: Performance:** These requirements ensure the system is fast and efficient, responding quickly when tracking attendance and handling many users without slowing down. It focuses on speed and efficiency under normal and peak loads.
- **NFR5: Usability:** This requirement ensures the system is easy and intuitive for students, teachers, and administrators to use through the creation of an intuitive and user-friendly interface.
- **Offline Functionality:** This requirement ensures application still works properly in poor or no connection.
- **NFR6:** The system shall be scalable to accommodate multiple institutions.

4. External Interface Requirements

4.1 User Interfaces

- **Login Screen:** For user authentication.
- **Attendance Screen:** To capture facial data and verify location.
- **Dashboard:** For educators to monitor attendance.

4.2 Hardware Interfaces

- **Camera:** For facial recognition.
- **GPS Module:** For location tracking.

4.3 Software Interfaces

- **Facial Recognition API:** For processing facial data.
- **Geolocation API:** For determining user location.

4.4 Communications Interfaces

- **Internet Connectivity:** Required for real-time data synchronization.

5. Requirement Priority

Requirements were categorized using the MoSCoW (Must Have, Should Have, Could Have and Won't Have) method:

Must Have (M)

- **Definition:** These are critical requirements that must be delivered for the project to be considered a success.
- **Importance:** Without these features, the system will fail to meet its core objectives and may not function properly.
- **Example:** User authentication and facial data storage.

Should Have (S)

- **Definition:** Important requirements that add significant value but are not vital for immediate delivery.
- **Importance:** These features are highly desirable and enhance the user experience but can be postponed if necessary.
- **Example:** Class notifications.

Could Have (C)

- **Definition:** Nice-to-have features that can improve the system but are not essential for its primary function.
- **Importance:** These features can be implemented if time and resources allow but do not impact the project's success.
- **Example:** Bluetooth beckoning

Won't Have (W)

- **Definition:** Features that are agreed upon as not being included in the current project scope.
- **Importance:** These requirements may be considered for future phases but are not a priority for the current development cycle.
- **Example:** Integration with other university systems.

Must Have	Should Have	Could Have	Won't Have (For now)
User Authentication	Class Notifications	Notification to users for missed check-ins	Integration with other university systems
Facial Data Storage	Geofence Management	Export attendance reports	
Location Verification	Course Management	Bluetooth Beckoning	
Attendance Recording			
Attendance Override			
Triggering Facial Recognition			

6. Feasibility Studies

This section identifies and addresses potential technical and operational challenges in implementing a mobile-based attendance management system that combines geofencing and facial recognition. Solutions are proposed to ensure the system is both realistic and practical in real-world academic settings.

6.1 Timing Conflicts with Facial Recognition Trigger

Issue:

The system is designed to trigger facial recognition after 70% of the lecture duration (approximately 1 hour and 20 minutes in a 2-hour class). However, lectures often start late due to the lecturer arriving late. For instance, if a class begins an hour late, facial recognition would still activate at the scheduled 1 hour 20-minute mark, which would actually be just 20 minutes after the late start.

Proposed Solution:

We introduce a **functional requirement** allowing the lecturer to **manually trigger the facial recognition module** at the actual start of the lecture. This ensures that the 70% threshold is calculated from the real-time start, not the scheduled one.

6.2 Disruptions in Attendance Timing

Issue:

Short-term exits (e.g., to the bathroom or for water) may affect the 70% presence calculation. It is unclear how the system would handle such breaks when measuring student presence.

Proposed Solution:

Use **continuous location and face recognition monitoring**, allowing students to temporarily exit the geofenced area (for a maximum of, say, 10 minutes). An exit duration threshold should be configurable. If the absence exceeds the threshold, the system flags it and excludes the student from full attendance credit unless manually validated by the lecturer.

6.3 Student Dismissal by Lecturer

Issue:

A student may have already had their attendance recorded, but later gets dismissed from class for misconduct.

Proposed Solution:

Add a **lecturer attendance override feature** to revoke or update a student's attendance status after initial check-in. This ensures the final attendance record reflects discipline-related absences.

6.4 Valid Absence from Lecture

Issue:

Students may miss a lecture due to valid reasons such as illness or emergencies. These absences should not be treated the same as unexcused ones.

Proposed Solution:

Add a **lecturer attendance override feature** to revoke or update a student's attendance status after initial check-in. This ensures the final attendance record reflects discipline-related absences.

6.5 Network Dependency

Constraint:

Real-time attendance tracking requires stable internet connectivity, which may not always be available.

Solution:

Design the system to work in **offline mode**. This will make use of Bluetooth beckoning

Bluetooth Beckoning refers to a feature that allows devices to communicate with one another using Bluetooth technology in a specific way. Here's a simple breakdown:

- **Purpose:** It enables one device to signal or "beckon" another device, often to initiate a connection or to send notifications.
- **Use Cases:**
 - **Proximity Alerts:** A device can alert users when they are near a particular location or device.
 - **User Interaction:** It can prompt users to take action, such as opening an app or responding to a notification.
- **Functionality:** Typically involves sending a signal or message that can trigger a response on the receiving device, enhancing user experience and interaction.

Example in our situation:

In an attendance system, Bluetooth beckoning notifies students when they enter a classroom, reminding them to check in or confirming their presence.

This feature leverages Bluetooth's capability for short-range communication, making it useful for various applications in mobile and wearable technology.

6.6 Device Compatibility

Constraint:

Some students may use devices with incompatible hardware (e.g., no facial recognition capability).

Solution:

Provide **alternative check-in methods**, such as QR code verification, for low-end devices, with strict controls to prevent abuse.

6.6 False Positives/Negatives in Facial Recognition

Constraint:

Facial recognition systems may misidentify students due to lighting, camera quality, or facial changes.

Solution:

Use **multi-sample verification** and periodic re-training of the recognition model. Incorporate manual verification fallback options for uncertain cases.

manual verification fallback options for uncertain cases.

6.7 Data breaching risks.**Constraint:**

User data being sent to VPS hosting services or public domain storage buckets.

Solution:

Major parts of the application like the model are deployed as an API endpoint which addresses data privacy as it could be deployed on the faculty's local server or a secure cloud store/service.

7. Conclusion

Requirement Analysis is a critical phase in the software development lifecycle. Following the data collected during the requirement gathering phase, this task involves interpreting, refining, and validating those requirements to ensure the system will meet stakeholder expectations and operate within technical constraints. In this report, we outlined the key activities involved in the requirement analysis process for a mobile-based attendance management system using geofencing and facial recognition.

The following steps were employed to successfully come up with the above SRS

7.1 Review and Analysis of Requirements

A thorough review of the collected requirements was conducted based on the following criteria:

- **Completeness:** Each functionality related to user check-in, face recognition, GPS validation, and admin control was reviewed for coverage.
- **Clarity:** All requirements were rewritten in clear, unambiguous language.
- **Technical Feasibility:** Requirements such as real-time facial recognition and geolocation were assessed for compatibility with available APIs and mobile devices.
- **Dependency Relationships:** For example, GPS validation must occur before facial recognition in the check-in workflow.

7.2 Validation with Stakeholders

A stakeholder validation session was held with:

- **Students** to confirm that their requirements are well understood.
- We were unable to validate requirements with staff upon the time of writing this report.

All requirements were presented and validated via walkthroughs and feedback. Minor updates were made based on the feedback, particularly around privacy and data consent notices.