



**Tshwane University  
of Technology**

*We empower people*

**FACULTY  
OF  
INFORMATION  
AND  
COMMUNICATION TECHNOLOGY**

**NATIONAL DIPLOMA: COMPUTER SYSTEMS  
ENGINEERING**

**INTELLIGENT INDUSTRIAL SYSTEMS:  
IDC30BI/PJD301B**

**USER MANUAL**

**PROJECT TITLE:...**Student Attendance Management System.....

**LECTURER NAME...**Muwanguzi Mark Ntume and Chunling Du.....

**STUDENTS NAME:...**MALOGADIHLARE TEBOGO MARVIN.....

**STUDENTS NUMBER.....**218161693.....

**EMAIL ADDRESS.....**218161693@tut4life.ac.za.....

**CONTACT NUMBERS .....**0633675240.....

# TABLE OF CONTENTS

---

1.	SCOPE.....	3
2.	REFERENCED DOCUMENTS.....	3
3.	SOFTWARE SUMMARY.....	3-7
4.	ACCESS TO THE SOFTWARE.....	7-9
5.	PROCESSING REFERENCE GUIDE.....	9
6.	QUERY PROCEDURES.....	10
7.	NOTES.....	10
8.	REFERENCE.....	11

## 1. SCOPE

### 1.1 Identification.

This document describes the Student Attendance Management System (SAMS), designed to automate attendance tracking using facial recognition and RFID technology. The current version is 1.0 which is compatible with all major web browsers on desktop and mobile platforms.

### 1.2 System overview.

The system allows users (students, instructors, and administrators) to log in, manage their profiles, and track attendance. It provides real-time updates and supports administrative tasks such as managing user accounts and viewing attendance records. The system was developed in response to the need for efficient attendance management in educational institutions. The website is hosted on Render.com and the database on AWS, therefore it is accessible from anywhere there is internet connection. Users are students, instructors and admins. The developer is Tebogo Marvin Maloagadithlare.

### 1.3 Document overview.

This manual provides instructions for users to access and use the SAMS. It includes operational guidelines, error handling, and contact information for support. The host of the website Render.com provides SSL security, it uses https and the passwords are encrypted in the database.

## 2. REFERENCED DOCUMENTS

- System Requirements Document (SRD)
- Software Development Lifecycle (SDLC) Overview
- YouTube

## 3. SOFTWARE SUMMARY

2

3

### 3.1 Software application.

SAMS is designed for efficient tracking and management of student attendance using facial recognition technology and RFID tags. The software helps with accuracy and minimizes manual attendance processes.

### 3.2 Software inventory.

Front-end: HTML, CSS, JavaScript Back-end:

Spring Boot, Hibernate Database:

MySQL Libraries:

Face-Api.js for face recognition, various JavaScript libraries for UI enhancements, and various libraries for RFID Tags like MFRC522 library, etc.

Software code is on Github in case of an emergency.

### 3.3 Software environment.

Hardware Requirements

4 GB RAM minimum

100 GB Disk space  
Software Requirements  
Java 17+  
MySQL (Also an online DB like AWSDB if you are going to host the website)  
Web Browser (latest versions of Chrome or Edge)  
Git For Windows and Github account  
Account on Render.com

### 3.4 Software organization and overview of operation.

#### a. Logical Components of the Software

The Student Attendance Management System (SAMS) consists of several logical components from the user's point of view:

- **User Interface (UI):** This component is the front-end of the system where users (students, instructors, and administrators) interact with the system. The UI allows users to log in, manage profiles, and access features related to attendance management.
- **API Layer:** The back-end system is powered by a Spring Boot RESTful API. This layer handles all the requests from the user interface, processes them, and returns responses. The API is responsible for interacting with the database, performing CRUD operations (Create, Read, Update, Delete) on entities like users, students, classes, and attendance.
- **Database:** SAMS uses a MySQL database to store user data, attendance records, and class/course details. Key data files include tables for:
  - **Users:** Stores login credentials and roles (admin, instructor, or student).
  - **Students:** Contains student profiles.
  - **Clazz:** Holds information about different classes and their schedules.
  - **Attendance:** Logs each student's attendance per class session.
- **Facial Recognition & RFID Modules:** The system includes Face-API.js-based facial recognition to automate attendance marking and RFID technology for scanning student tags. The facial recognition module accesses the camera to detect and verify student faces, while the RFID module reads data from RFID tags for student identification.

#### b. Performance Characteristics

##### 1.Types, volumes, rate of inputs accepted:

- Inputs accepted include student login information, RFID tag scans, facial recognition data, and class attendance submissions.
- The system can handle high volumes of login and attendance requests simultaneously during peak times, such as class start times.

##### 2.Types, volume, accuracy, rate of outputs that the software can produce:

- The system outputs include detailed attendance reports for administrators and instructors, and personal attendance logs for students.
- Attendance accuracy is maintained through facial recognition and RFID matching.
- Output generation (attendance reports) occurs in real-time.

##### 3.Typical response time and factors that affect it:

- The average response time for login and data queries is under 3 seconds, assuming the system is not under heavy load.
- Factors affecting response time include network speed, server load, and database size.

#### 4. Typical processing time and factors that affect it:

- Attendance processing (face recognition or RFID tag validation) typically takes less than a second, depending on hardware and internet speed.
- Delays might occur with large volumes of simultaneous attendance submissions.

#### 5. Limitations:

- The system is designed to handle up to 1,000 users, including students, instructors, and admins.
- The system may experience slowdowns if this number is exceeded without scaling the database and server resources.

#### 6. Error rate:

- Expected error rates are low due to the use of RFID and facial recognition technologies. Misidentifications during face recognition could occur in about 20% of cases, depending on the lighting and camera quality.

#### 7. Reliability:

- The system is expected to have 99.9% uptime if deployed on a robust cloud platform like Render.com. Regular maintenance and database backups are advised to ensure reliability.

#### c. Relationship of the Functions Performed by the Software with Interfacing Systems, Organizations, or Positions

- **Administrators:** Manage the overall system, including user roles, class schedules, and reports.
- **Instructors:** Access attendance records for their assigned classes and track students' attendance.
- **Students:** Mark attendance and access their own attendance history.

SAMS interfaces with third-party systems for facial recognition Face-API.js and communication systems such as email notifications to inform students or instructors of attendance status.

#### d. Supervisory or Security Controls

- **Password Protection:** All users must log in with a unique username and password. Passwords are stored securely in the database using hashing algorithms to prevent unauthorized access.
- **Role-based Access Control (RBAC):** Different user roles (admin, instructor, student) have specific privileges. Administrators have full access, instructors have access to class and student data, while students can only access their personal attendance records.
- **Data Encryption:** All sensitive communication between the client and server, including login credentials and personal data, is encrypted using HTTPS to protect against interception.
- **Session Management:** User sessions are managed securely using Spring Boot's session management features, preventing unauthorized access after login.

### 3.5 Contingencies and alternate states and modes of operation.

In times of emergency or during unexpected system states, the SAMS software will handle the following contingencies:

- **Database Downtime:** If the database becomes unavailable (due to server failure or network issues), the system will display an appropriate error message, and all data submissions (such as attendance marking) will be queued until the system is back online. No new data can be added or modified while the database is down. Administrators will manage such outages.
- **Offline Mode for RFID Attendance:** In case of a server or network issue, students can still mark attendance using the RFID system. The data will be stored locally on the reader device and uploaded to the server once the connection is restored. This ensures that no attendance data is lost during outages.
- **Manual Attendance Logging:** In the event that both RFID and facial recognition systems are down, instructors can manually log student attendance via the web interface. This will serve as a backup procedure, and attendance data can be later verified and adjusted.
- **System Maintenance Mode:** When the system undergoes scheduled maintenance or upgrades, users attempting to access the system will be presented with a "Maintenance Mode" page. No operations can be performed during this period, but users will be informed of the expected downtime and when the system will be operational again.

### 3.6 Security and privacy.

The SAMS software has several built-in features to ensure both security and privacy of user data:

- **Authentication and Authorization:** Users must log in with a valid username and password to access the system. Role-based access control (RBAC) ensures that users can only perform actions authorized by their role (e.g., student, instructor, or admin). Admins manage permissions for different roles.
- **Data Encryption:** All sensitive data transmitted between the client and server (such as login credentials) is encrypted using HTTPS. This prevents unauthorized access or interception of data during transmission.
- **Password Protection:** User passwords are securely stored using hashing algorithms. Even system administrators cannot retrieve users' plain-text passwords.
- **Privacy Considerations:**
  - **Attendance Data:** Student attendance data is only accessible to authorized users (students themselves, instructors, and administrators).
  - **Personal Information:** Personal data (e.g., names, emails) collected during the registration process is securely stored and only used for the purposes of attendance tracking and management.
  - **Facial Recognition and RFID:** Any biometric data (facial descriptors) is stored securely, and access to this information is restricted to authorized personnel only.
- **Unauthorized Copying:** Unauthorized copying or distribution of the SAMS software or related documentation is strictly prohibited. Violation of this policy may result in legal action. Users are warned against making or distributing illegal copies of the software.

### 3.7 Assistance and problem reporting.

If users encounter issues while using the SAMS software or require assistance, the following points of contact and procedures are available:

**Helpdesk Support:** For any technical problems or assistance, users can contact the SAMS helpdesk at:

- **Email:** samsattendance@gmail.com
- **Phone:** +27(00 000 000)
- **Hours:** Monday to Friday, 9 AM to 5 PM (local time)

**Software Updates:** It is recommended that users regularly check for updates to ensure they are running the latest version of SAMS with all security patches applied.

## 4. ACCESS TO THE SOFTWARE

This section shall contain step-by-step procedures oriented to the first time/occasional user. Enough detail shall be presented so that the user can reliably access the software before learning the details of its functional capabilities. Safety precautions, marked by WARNING or CAUTION, shall be included where applicable.

### 4.1.1 Access control.

The SAMS system employs robust access control to ensure only authorized users can interact with the application.

- **Obtaining a Password:**

- New users are create a password during registration.

- **Security and Privacy Considerations:**

- All passwords are securely hashed and stored.
  - Only authorized users (students, instructors, admins) can access specific data in the SAMS system, ensuring privacy.

### 4.1.2 Installationandsetup.

**User Authentication:** The system requires a valid login (username and password) for access. User roles (student, instructor, admin) are assigned based on user type, which affects what functionalities they can access.

**Software Installation:**

SAMS is a web-based application accessible via a browser, so no local installation is required.

**Configurations:**

For administrators, configuration of system-wide settings, such as attendance parameters, can be done through the "Admin Dashboard."

**File and Data Management:**

No local files need to be deleted or overwritten. Data backups are managed by the system administrators.

### 3.8 Initiating procedure.

The following steps outline how users begin their work with SAMS:

**1.Login:** Users navigate to the SAMS login page, enter their credentials, and log in.

**2.Dashboard Access:**

- Students will see their attendance history and upcoming classes.
- Instructors can view attendance records, mark attendance, and manage classes.
- Admins have full control, including user management, class management, and system settings.

**3.Checklist for Problem Solving:**

- Ensure you have the correct login credentials.
- Check internet connection if experiencing access issues, or reload the page.
- Contact support if login issues persist.

### 3.9

### 3.10 Description of inputs.

#### 4.3.1 Input conditions.

**Reason for Input:** User inputs are primarily for marking attendance, managing classes, registering new users, and updating personal information.

**Frequency:** Input can be on-demand, such as daily attendance marking or periodic updates for class schedules.

**Input Medium:** Input is done through forms on the web-based interface.

#### 4.3.2 Input formats.

**Student Attendance:** Input consists of RFID scans or facial recognition via the mobile website.

**Admin Inputs:** For class creation or user edits, input is entered into specific fields on forms via the Admin Dashboard.

#### 4.3.3 Composition rules.

**Input Length:** Text inputs like names have a limit of 255 characters.

**Labeling and Formatting:** Fields like student name, class, and attendance time must be filled correctly to ensure accurate data recording.

#### 4.3.4 Input vocabulary.

#### 4.3.5 Sample inputs.

### 3.11 Description of outputs.

#### 4.4.1 General description.

**Reason for Outputs:** Outputs are generated to provide reports such as attendance records and user management reports.

**Frequency:** Reports are generated daily (attendance) or as requested by the user (on-demand).

#### 4.4.2 Output formats.

#### 4.4.3 Sample outputs.

**Headers and Footers:** Attendance reports include headers with dates and class names.

**Data Sections:** Reports contain rows representing each student's attendance time.



### 3.12 **Recovery and error correction procedures.**

When errors occur, the system generates specific error codes. Users can follow the troubleshooting steps listed in the error messages. The system also allows administrators to manually edit student and instructors in case of errors.

### 3.13 **Stopping and suspending work.**

Users can log out to stop work, or the system will automatically log out users after a period of inactivity.

## 5. PROCESSING REFERENCE GUIDE

This section details how to interact with the SAMS system and perform core functions.

### 3.14 Capabilities.

The SAMS system integrates:

- Attendance marking (via RFID and facial recognition)
- User registration
- Class and student management

### 3.15

Conventions.

**Colors:** Red indicates error fields, green indicates successful operations.

**Icons:** Standard icons are used for common operations like edit, delete, and save.

Processing procedures.

This section covers:

**Attendance Management:** How instructors mark attendance, either automatically via RFID or manually.

**Student Registration:** Admins can register new students or instructors, assigning them roles and classes.

### 3.16

### 3.17 Aspect of software use.

The software allows admins to manage:

- **Classes:** Create, update, or delete class schedules.
- **Students:** Add students, track attendance, and assign RFID tags.

### 3.18

### 3.19 Related processing.

Some operations like report generation or backups are handled in the background. These do not require user intervention but are critical for system reliability.

### 3.20 Data backup.

Daily data backups are performed automatically, ensuring no data is lost in case of a failure.

### 3.21 Recovery from errors, malfunctions, and emergencies.

The system provides detailed error messages to guide users through resolving common issues, such as login failures or database disconnections.

- 3.22 Messages.  
The system generates messages such as "Email already exists", "Username already exists" or "Attendance recorded successfully."
- 3.23 Quick-reference guide.  
Git commands:  
git add .  
git commit -m "message"  
git push origin master

## 6. QUERY PROCEDURES

- 3.24 Database/data file format.  
The system stores user and attendance data in a MySQL database. Each student record contains:
- ID**  
**First Name**  
**Last Name**  
**Attendance**  
**Classes**  
**UserId**  
**RFIDTagId**
- 3.25 Query capabilities.  
Admins can generate attendance reports for any period by querying the database.
- 3.26 Query preparation.  
Users can query data through the Admin Dashboard, which includes search filters for students, classes, and attendance dates.
- 3.27 Control instructions.

## 7. NOTES

This section includes:

- **Acronyms:**
  - **RFID:** Radio-Frequency Identification
  - **SAMS:** Student Attendance Management System

## 8. REFERENCE

[www.donskytech.com/esp32-rfid-web-server/](http://www.donskytech.com/esp32-rfid-web-server/)

[www.youtube.com/watch?  
v=p3Alecycvok4&ab\\_channel=bobbyctchan](http://www.youtube.com/watch?v=p3Alecycvok4&ab_channel=bobbyctchan)

[www.github.com/justadudewhohacks/face-api.js/](https://www.github.com/justadudewhohacks/face-api.js/)

**A. APPENDIXES**