Project Name
Automated Attendance System for Rural Schools

---

1. Introduction

1.1 Purpose

The main goal of this document is to describe the requirements for an Automated Attendance System. This system helps teachers in rural areas take attendance faster using AI and face recognition, replacing the old manual register books.

1.2 Scope

This project is a mobile and web application. It allows teachers to register students, take a photo of the class to mark attendance, and save the data. Because rural areas have bad internet, the app works offline and uploads data later. It also sends a simple SMS to parents if their child is not in school.

1.3 Definitions and Acronyms

- **AAS:** Automated Attendance System.
- **Biometrics:** Using a person's face or features for identification.
- **Offline Mode:** The ability of the app to work without an internet also connection.
- **SMS:** Short Message Service (Normal text message).
- **CSV:** A file format used to upload lists of student names.

1.4 Intended Audience

- Project guides and college professors.
- Students working on the development.
- School staff who will use the software.

---

2. General Description

2.1 Perspective

This system is a standalone tool designed for schools. It has two parts: a mobile app for the teachers to use in the classroom and a website for the principal to check the final records. It is built to work on cheap Android tablets or phones.

2.2 Main Functions

- Registering students with their photos.
- Taking attendance by scanning faces.
- Storing records locally when there is no network.
- Sending auto-alerts to parents.
- Generating simple attendance reports (daily/monthly).

2.3 User Types

- **Teachers:** They take the attendance and update student info.
- **Administrators/Principals:** They view the overall reports for the whole school.

2.4 Operating Environment

- **Mobile:** Any basic Android device (Version 8.0 or higher).
- **Web:** Any standard browser like Chrome or Firefox.
- **Database:** SQL-based storage for student records.

2.5 Constraints

- Only 1-2 students are developing this (limited manpower).
- The project must be finished within the semester (approx. 3-4 months).
- Must work on low-end phones with small storage.

2.6 Assumptions

- Teachers have a basic understanding of how to use a smartphone.
- The school will have internet access at least once a week to sync data.
- The camera on the device is clear enough to recognize faces.

---

3. Functional Requirements

3.1 Student Registration

- **FR1:** The system shall allow teachers to enter student name, roll number, and class.
- **FR2:** The system shall capture a reference photo of the student for the AI model.

3.2 Attendance Marking

- **FR3:** The teacher shall be able to take a group photo of the class.
- **FR4:** The AI should identify which students are present and mark them automatically.
- **FR5:** There must be an option to manually mark a student present if the AI misses them.

3.3 Data Storage & Sync

- **FR6:** The app shall save all attendance data on the phone memory if there is no Wi-Fi/Data.
- **FR7:** The app shall upload the saved data to the server as soon as the internet is back.

3.4 Notifications

- **FR8:** The system shall send an SMS to the parent's phone number if a student is absent.
- **FR9:** The teacher should be able to turn off these alerts if needed.

3.5 Reports

- **FR10:** The system shall show a list of students with low attendance.
- **FR11:** The principal should be able to download the attendance sheet as a PDF.

---

4. System Design & UML Diagrams

4.1 Use Case Diagram

This diagram shows how different users (Teacher and Admin) interact with the system.

- **Teacher:** Login, Register Student, Capture Image, View Class Report.
- **Admin:** Manage Teachers, View School Analytics, Database Backup.

4.2 System Architecture

The system follows a 3-tier architecture:

1. **Presentation Layer:** The Android App and Web Dashboard.
2. **Application Layer:** The AI processing engine (Face recognition logic).
3. **Data Layer:** SQLite for local storage and Firebase/MySQL for cloud storage.

4.3 Data Flow Diagram (DFD) - Level 0

The DFD shows how information moves from the teacher's camera to the final database and eventually to the parent’s phone via SMS.

---

5. System Methodology

5.1 Proposed Algorithm for Face Recognition

To make the report look more technical, explain the steps the AI takes:

1. **Image Acquisition:** Capturing a clear photo of the classroom.
2. **Pre-processing:** Grayscale conversion and noise reduction to handle low-light rural classrooms.
3. **Face Detection:** Locating faces using a pre-trained model like Haar Cascades or MTCNN.
4. **Feature Extraction:** Identifying unique points on the face (eyes, nose, mouth).
5. **Matching:** Comparing these points with the student photos stored during registration.

5.2 Implementation Tools (Tech Stack)

- **Frontend:** XML and Java/Kotlin for the Android App.
- **Backend:** Node.js or Python (Flask) for the server.
- **Database:** SQLite (Local) and Firebase (Cloud).
- **AI Library:** OpenCV or Google ML Kit.
- **Development Tool:** Android Studio.

---

6. Detailed Analysis of Rural Challenges

6.1 Low Connectivity Solutions

Explain how you solved the internet problem. Use a "Store-and-Forward" mechanism where the app checks for a network ping; if it fails, it flags the record as "Pending" and retries every 30 minutes in the background.

6.2 Hardware Compatibility

Since rural schools use old tablets, the app uses **Edge Computing**. This means the face recognition happens on the phone itself instead of sending a heavy 5MB image to a server, saving data and time.

---

7. Testing & Quality Assurance

7.1 Unit Testing

Each part of the code (like the SMS trigger or the Login screen) is tested individually to make sure there are no bugs.

7.2 Accuracy Testing

We test the AI model with different scenarios:

- **Varying Light:** Testing in dim classroom light.
- **Obstructions:** Testing if a student is wearing a mask or glasses.
- **Distance:** Testing if students in the back row are detected.

7.3 User Acceptance Testing (UAT)

A small group of teachers will test the app to see if the buttons are easy to find and if the "Offline Mode" works as promised.

---

8. Conclusion & Future Enhancements

8.1 Summary

The Automated Attendance System for Rural Schools provides a cost-effective way to track student presence. It removes the burden of paperwork from teachers and keeps parents informed.

8.2 Future Scope

- **Voice Recognition:** Adding voice-based attendance for backup.

- **Mid-Day Meal Tracking:** Using the same system to count students for government meal programs.
- **Performance Prediction:** Using the attendance data to predict which students might fail exams due to low presence.

Architecture Diagram