Software Requirement Specifications

<Project Title>

Submitted by

Muhammad Ali Ilyas (01251)

Submitted to

Ms. Kainat

Department of Information Technology

Faculty of Computing

The Islamia University of Bahawalpur

Meeting Details

Sr No	Details	Date	Supervisor Signature

Summary

The Smart Attendance System using QR Code is designed to automate attendance marking in a simple, reliable, and cost-effective way. Each user (student or employee) receives a unique QR code that contains their identification details. When scanned using a webcam or mobile camera, the system decodes the information and records attendance along with the date and time. All data is stored automatically in a CSV file or database.

The system is built using Python and includes modules for QR code generation, QR code scanning, attendance logging, and database management. An optional GUI may also be included for easier interaction.

Key features include user registration, QR code creation, real-time scanning, automatic attendance recording, reporting, and exporting attendance data. The system focuses on accuracy, speed, ease of use, and minimal hardware requirements.

Non-functional requirements emphasize performance (fast scanning), security (protected data), reliability (consistent operation), usability (easy for all users), and portability (works across operating systems).

Overall, the system provides an efficient and contactless attendance solution suitable for schools, colleges, offices, and events, with potential for future upgrades such as cloud integration or face recognition.

Contents

1. Introduction

The Smart Attendance System using QR Code is designed to modernize and streamline the process of recording attendance in educational institutions, offices, and organizations. Traditional attendance methods such as manual entry, paper registers, or calling names are time-consuming, prone to human errors, and difficult to manage for large groups. This system provides a fast, contactless, and automated solution by using QR codes as unique identifiers for each student or employee.

In this system, every user is assigned a unique QR code that stores their identification details. When the QR code is scanned through a webcam or mobile camera, the system decodes the information and marks attendance automatically along with the date and time. The data is then stored securely in a digital database or CSV file for easy retrieval and analysis.

Developed using Python, the system integrates QR code generation, scanning, data logging, and optional user interface elements. Its simplicity, low cost, and reliability make it a practical and scalable solution for institutions looking to improve efficiency and reduce manual workload. This SRS document outlines all the functional and non-functional requirements needed to develop and deploy the Smart Attendance System effectively.

1.1. Purpose

The purpose of this Software Requirements Specification (SRS) document is to define the functional and non-functional requirements for the Smart Attendance System using QR Code. The system automates the attendance process by generating unique QR codes for each student or employee and scanning them to record attendance with date and time. The document serves as a guide for developers, testers, and stakeholders throughout the development lifecycle.

1.2. Scope

The Smart Attendance System will:

- Generate unique QR codes for each registered user.
- Scan QR codes using a webcam or mobile camera.
- Decode user information and record attendance automatically.
- Store attendance data in a database or CSV file.
- Provide reports and data access for analysis.
- Optionally offer a basic user interface for easy operation.

The system will be used in schools, colleges, universities, offices, and events to ensure accurate and contactless attendance tracking.

.

1.3. Definitions

- **QR** Code: A machine-readable label that stores user identification data.
- **Attendance** Log: A stored record consisting of user ID, date, time, and status.
- **Database**: Digital storage system for user and attendance records.
- **UI (User Interface)**: Optional frontend interface for interaction.

1.4. Technologies

- **Programming Language**: Python
- **Libraries/Modules**: `qrcode`, `opencv-python`, `pyzbar`, `pandas`, `datetime`, `tkinter` (optional)
- **Database**: CSV files or SQLite/MySQL
- **Hardware**: Webcam or mobile camera

2. Overall Requirements

2.1. Product Perspective

The system is standalone and does not require integration with external systems. It uses Python for both QR generation and scanning. Attendance data is stored locally or on a server depending on deployment.

2.2 Product Features

- QR code generation for every user.
- QR code scanning through webcam/mobile camera.
- Automatic attendance marking.
- Real-time date and time logging.
- Data storage in CSV or database.
- Basic reporting and data filtering.
- Optional GUI for easier interaction.

.

2.3 User characteristics

- **Admin/Teacher/Manager**
    - Generates QR codes
    - Views attendance reports
    - Manages database
- **Student/Employee**
    - Shows QR code for scanning

Basic computer literacy is sufficient.

2.4 Constraints

- Requires a functioning camera.

- Lighting conditions may affect scanning quality.
- QR code must be visible and undamaged.
- Python environment must be properly configured.

2.5 Assumptions & Dependencies

- Users will have their QR code available at the time of attendance.
- The system has access to the camera.
- Admin will maintain data base Accuracy

3. Specific Requirements

3.1 Functional Requirements

3.1.1 User Registration

- The system shall allow the administrator to register new users.
- The system shall generate a unique user ID for each person.

3.1.2 QR code Generation

- The system shall generate a QR code containing user information (ID, name).
- The system shall save QR codes as image files.

3.1.3 QR Code Scanning

- The system shall scan QR codes using a camera.
- The system shall decode the scanned QR code.
- The system shall validate the user information.

3.1.4 Attendance Marking

- The system shall record attendance automatically after successful scan.

- The system shall store date, time, and user ID in a database/CSV file.
- The system must prevent multiple entries for the same user within a restricted time (optional).

3.1.5 Database operations

- The system shall store all attendance records in CSV or SQL database.
- The system shall allow exporting attendance records.
- The system shall allow filtering by date, user, or status.

3.1.6 Reporting

- The system shall generate attendance reports.
- The system shall show daily, weekly, or monthly attendance summaries.

3.1.7 Optional GUI

If UI enabled:

- The system shall provide buttons for scanning, viewing reports, and generating QR codes.

3.2 Non-Functional Requirements

3.2.1. Performance

- QR scanning should process within 1–2 seconds.
- The system should support up to 1000 users without performance loss.
- **3.2.2 Security**
   User data stored in QR codes should not expose sensitive information.

- Database must be protected from unauthorized access.

3.2.3 Reliability

- The system should operate consistently under normal lighting conditions.
- Data should not be lost even if the system shuts down unexpectedly.

3.2.4 Usability

- Interface must be simple and easy to understand.
- Users should scan the QR code quickly without extra steps.

3.2.5 Portability

- The system should run on Windows, macOS, and Linux.
- Database and QR code files should be portable.

4. System Architecture

4.1 Components

- QR Code Generator Module
- QR Code Scanner Module
- Attendance Logger
- Database Handler
- Optional UI Module

4.2 Data Flow

- Admin registers user → QR Generated
- User shows QR → Camera scans
- QR data decoded → User authenticated
- Attendance logged in database
- Report generated when requested

5. Future Enhancements

- Face Recognition integration.
- Cloud storage and online dashboard.

- Mobile app support.
- SMS/Email notifications after attendance.

6. Conclusion

The Smart Attendance System using QR Code provides a cost-effective, fast, and reliable solution to automate attendance management using Python. This SRS document outlines all necessary requirements to guide the development and implementation of the system.

- **Reference**

- ➢ Python Software Foundation. *Python Documentation*. Available: https://www.python.org/doc/
- ➢ Segno. *QR Code Generation in Python*. Available: https://pypi.org/project/segno/
- ➢ Pyzbar. *Python Barcode and QR Code Decoding*. Available: https://pypi.org/project/pyzbar/
- ➢ OpenCV Documentation. *Computer Vision with Python*. Available: https://docs.opencv.org/
- ➢ Kaur, P., & Singh, J. (2020). *QR Code Based Attendance System: A Review*. International Journal of Computer Applications, 175(6), 25–30.
- ➢ Pandas Documentation. *Data Handling in Python*. Available: https://pandas.pydata.org/docs/

Document Version : 1.0
Last Update : 4-Dec-2025
Status : Draft for review.