Attendance Tracker Project

1. Mini Project Details

Mini-Project Title: QR Code-Based Attendance Tracker

Mini-Project Group Number: Group 1

Group Members:

Name	Class Roll	University Roll	Registration Number	Semester	Department
Linika Agarwal	69	10900223069	231090110597	3rd	IT
Misbah Rahaman	78	10900223078	231090110606	3rd	IT
Prakriti Samanta	84	10900223084	231090110612	3rd	IT
Priyanshu Roy	89	10900223089	231090110617	3rd	IT
Rebanta Biswas	98	10900223098	231090110626	3rd	IT
Sagnik Kumbhakar	107	10900223107	231090110635	3rd	IT
Samaira Shaw	110	10900223110	231090110638	3rd	IT
Sandipan Sasmal	111	10900223111	231090110639	3rd	IT

2. Objective of the Mini-Project

The primary objective of the project is to design and implement a QR Code-Based Attendance Tracking System that eliminates the traditional, error-prone, and time-consuming process of manual attendance recording. The system aims to provide a fast, reliable, and user-friendly solution for institutions to streamline their attendance management process, enhance accuracy, and generate insightful analytics.

3. Key Features

- QR Code-based attendance tracking.
- Analytics generation for month-wise and semester-wise attendance.
- Simple and intuitive GUIs for attendance management and QR code generation.

4. Folder and File Structure

Root Folder : Attendance-Tracker-Python-Project-main
This is the main directory containing all necessary files and subfolders for the project.

Subfolders :

1. **analytics:**
    - Stores attendance analytics data organized by months.
    - Example structure: analytics/January 2025/A/analytics.xlsx.
2. **analytics_sems:**
    - Presumably for semester-wise analytics.
    - Could follow a similar structure to analytics.
3. **attendance:**
    - Likely holds attendance logs or processed attendance sheets.
4. **generated_qrs:**
    - Stores generated QR codes, each representing a unique student or attendee.
5. **sandy:**
    - The exact purpose is unclear but may contain user-specific configurations or additional scripts.

Files in the Root Folder :

1. **.gitattributes:**

   Configuration file for Git to standardize line endings across environments.

2. **GUI.py:**

   Main script for launching the Attendance Tracker application interface.

3. **admin_pass.txt:**

   Contains the admin password for authentication (alohmora by default).

4. **attend.py:**

   Script for scanning QR codes, logging attendance, and performing basic checks.

5. **attendanceGUI.py:**

   GUI-based application for running attend.py with a visual interface.

6. **attendance_analytics.py:**

   Analytics script that processes attendance data from Excel sheets.

7. **bg.jpg:**

   Background image used in the GUI interface.

8. **generate.py:**

   Generates QR codes for each student based on data in students.xlsx.

9. **qr generater gui.py:**

   GUI for running the QR code generation script.

10. **students.xlsx:**

    Excel file containing student details, which is the data source for QR code generation.

5. Code Overview

Detailed summaries of scripts:

1. **GUI.py:**
   - Implements the main graphical interface for the application.
   - Features:
     - Dynamically resizes the background image.
     - Provides navigation to various features of the project.
2. **attend.py:**
   - Core logic for attendance tracking.
   - Uses pyzbar for QR code decoding and OpenCV for camera handling.
   - Logs scanned data in attendance files and verifies admin authentication via admin_pass.txt.

3. **attendanceGUI.py:**
   - A GUI wrapper for attend.py.
   - Displays logs in real-time and prompts for admin credentials as needed.

4. **attendance_analytics.py:**
    - Processes attendance data stored in analytics folders.
    - Generates monthly and stream-wise attendance summaries using Pandas.

5. **generate.py:**
    - Reads student details from students.xlsx.
    - Generates personalized QR codes saved in the generated_qrs folder.

6. **qr generater gui.py:**
    - Provides a GUI for running the generate.py script.
    - Allows users to trigger QR code generation without using the command line.

6. Setup Instructions

Prerequisites:

- Python 3.x.
- Libraries: tkinter, Pillow, pandas, openpyxl, MyQR, opencv-python, pyzbar.

Install required packages:

pip install pillow pandas openpyxl opencv-python MyQR pyzbar

Steps:

1. Extract the project folder.
2. Place the student data in students.xlsx. Ensure it contains fields like Name and ID.
3. Verify that admin_pass.txt contains the desired admin password.
4. Run the required script based on your use case.

7. Usage Guide

For Admin Authentication:

- The admin passcode is stored in admin_pass.txt. Update it if needed.

For QR Code Generation:

1. Run qr generater gui.py.
2. Ensure students.xlsx is up-to-date with student details.
3. Generated QR codes will appear in the generated_qrs folder.

For Attendance Tracking**:**

1. Run attendanceGUI.py.
2. Scan the QR codes using a connected camera or webcam.
3. Attendance will be logged automatically.

For Analytics:

1. Run attendance_analytics.py.
2. Processed analytics data will be available in the analytics folder.

8. Steps and Stages of the Mini-Project Work

Step 1: Requirement Analysis

- Gather requirements for attendance tracking and QR code-based systems.
- Understand the necessary features for GUI, QR code generation, attendance logging, and analytics.

Step 2: Design

- Design the overall system architecture.
- Plan the folder structure, file organization, and database (Excel file) structure.

Step 3: Development

1. **QR Code Generation:**
    - Create a script (generate.py) to generate unique QR codes for each student based on their university roll numbers.
2. **Attendance Tracking:**
    - Implement QR code scanning and attendance logging using OpenCV and Pyzbar libraries in attend.py.
3. **Graphical User Interface (GUI):**
    - Design intuitive GUIs for QR code generation (qr generater gui.py) and attendance tracking (attendanceGUI.py).
4. **Analytics:**
    - Implement the attendance_analytics.py script to process attendance logs and generate insights.

Step 4: Testing

- Test each component for accuracy and reliability:

- Validate QR code generation.
- Simulate attendance tracking and ensure logs are created correctly.
- Verify analytics results.

Step 5: Error Handling

- Address issues like missing or duplicate roll numbers, camera errors, and invalid Excel file formats.

Step 6: Deployment and Documentation

- Finalize the project setup.
- Create detailed documentation for the project, including setup instructions, error handling, and navigation flow.

9. Data Files

1. **students.xlsx:**
   - Columns include Name & University Roll No.
   - Acts as the source for QR code generation.
2. **cumulative_analytics.xlsx:**
   - Tracks cumulative attendance and other performance metrics.
   - Data is visualized using charts for better insights.

10. Screens and Pages

Below are snapshots of all key screens of the project:

```
[Running] python -u "c:\Users\Sandipan Sasmal\Desktop\python mini proj\generate.py"
c:\Users\Sandipan Sasmal\Desktop\python mini proj\generate.py:41: FutureWarning: Series.__getitem__ treating keys as positions is
deprecated. In a future version, integer keys will always be treated as labels (consistent with DataFrame behavior). To access a value by
position, use `ser.iloc[pos]`
  students_list = students[['name', 'university roll number']].apply(lambda x: [x[0].strip(), str(x[1]).strip()], axis=1).values.tolist()
  # List of (name, roll number) pairs
Student Data Loaded:
['Linika Agarwal', '10900223069']
['Misbah Rahaman', '10900223078']
['Prakriti Samanta', '10900223084']
['Priyanshu Roy', '10900223089']
['Rebanta Biswas', '10900223098']
['Sagnik Kumbhakar', '10900223107']
['Samaira Shaw', '10900223110']
['Sandipan Sasmal', '10900223111']

QR code already exists for the following students:
Linika Agarwal (University Roll Number: 10900223069)
Misbah Rahaman (University Roll Number: 10900223078)
Prakriti Samanta (University Roll Number: 10900223084)
Priyanshu Roy (University Roll Number: 10900223089)
Rebanta Biswas (University Roll Number: 10900223098)
Sagnik Kumbhakar (University Roll Number: 10900223107)
Samaira Shaw (University Roll Number: 10900223110)
Sandipan Sasmal (University Roll Number: 10900223111)
```

Default Loaded Student's Name List

```
Student Data Loaded:
               name  university roll number
0    Linika Agarwal             10900223069
1    Misbah Rahaman             10900223078
2  Prakriti Samanta             10900223084
3     Priyanshu Roy             10900223089
4    Rebanta Biswas             10900223098
5  Sagnik Kumbhakar             10900223107
6      Samaira Shaw             10900223110
7   Sandipan Sasmal             10900223111
line 16: mode: byte
QR code generated for: Sandipan Sasmal (University Roll Number: 10900223111) -> generated_qrs\Sandipan Sasmal_10900223111.png

QR code already exists for the following students:
Linika Agarwal (University Roll Number: 10900223069)
Misbah Rahaman (University Roll Number: 10900223078)
Prakriti Samanta (University Roll Number: 10900223084)
Priyanshu Roy (University Roll Number: 10900223089)
Rebanta Biswas (University Roll Number: 10900223098)
Sagnik Kumbhakar (University Roll Number: 10900223107)
Samaira Shaw (University Roll Number: 10900223110)
```

QR Code Generation for new student

```
Student Data Loaded:
['Linika Agarwal', '10900223069']
['Misbah Rahaman', '10900223078']
['Prakriti Samanta', '10900223084']
['Priyanshu Roy', '10900223089']
['Rebanta Biswas', '10900223098']
['Sagnik Kumbhakar', '10900223107']
['Samaira Shaw', '10900223110']
['Sandipan Sasmal', '10900223111']
line 16: mode: byte
QR code generated for: Linika Agarwal (University Roll Number: 10900223069) -> generated_qrs\Linika Agarwal_10900223069.png
line 16: mode: byte
QR code generated for: Misbah Rahaman (University Roll Number: 10900223078) -> generated_qrs\Misbah Rahaman_10900223078.png
line 16: mode: byte
QR code generated for: Prakriti Samanta (University Roll Number: 10900223084) -> generated_qrs\Prakriti Samanta_10900223084.png
line 16: mode: byte
QR code generated for: Priyanshu Roy (University Roll Number: 10900223089) -> generated_qrs\Priyanshu Roy_10900223089.png
line 16: mode: byte
QR code generated for: Rebanta Biswas (University Roll Number: 10900223098) -> generated_qrs\Rebanta Biswas_10900223098.png
line 16: mode: byte
QR code generated for: Sagnik Kumbhakar (University Roll Number: 10900223107) -> generated_qrs\Sagnik Kumbhakar_10900223107.png
line 16: mode: byte
QR code generated for: Samaira Shaw (University Roll Number: 10900223110) -> generated_qrs\Samaira Shaw_10900223110.png
line 16: mode: byte
QR code generated for: Sandipan Sasmal (University Roll Number: 10900223111) -> generated_qrs\Sandipan Sasmal_10900223111.png

All QR codes generated successfully.
QR codes saved in the folder: generated_qrs
```

QR Code Generation for all students

```
[Running] python -u "c:\Users\Sandipan Sasmal\Desktop\python mini proj\generate.py"
Error: Some students are missing university roll numbers: ['New student']
```

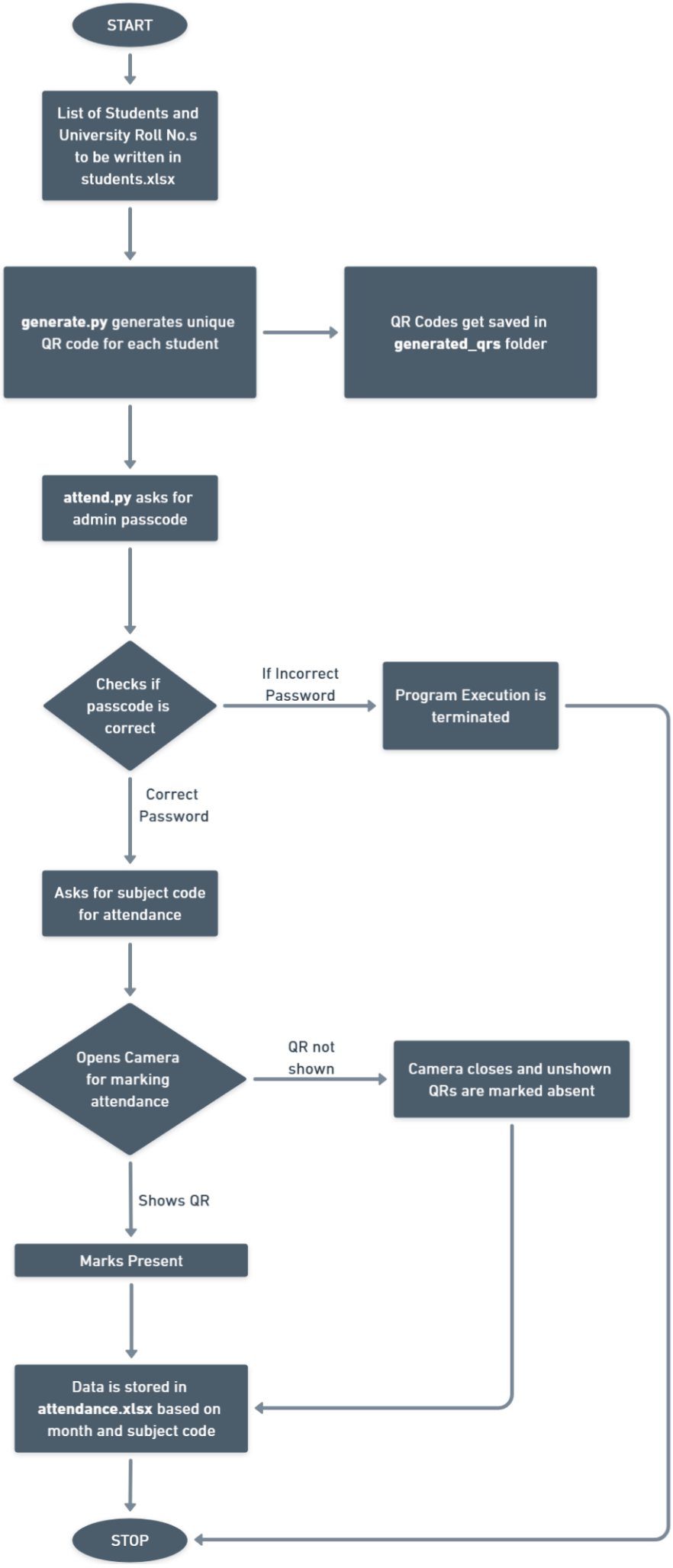
Error Handling during Missing Roll No.s

```
[Running] python -u "c:\Users\Sandipan Sasmal\Desktop\python mini proj\generate.py"
Error: Duplicate university roll numbers found for the following students: [['Sandipan Sasmal', 10900223111], ['New student', 10900223111]]
```

Error Handling during Duplicate Student entry

11. Navigation/Flow Diagram

The following flow diagram represents the navigation and flow of the application:


START
List of Students and University Roll No.s to be written in students.xlsx
generate.py generates unique QR code for each student
QR Codes get saved in generated_qrs folder
attend.py asks for admin passcode
Checks if passcode is correct
If Incorrect Password
Program Execution is terminated
Correct Password
Asks for subject code for attendance
Opens Camera for marking attendance
QR not shown
Camera closes and unshown QRs are marked absent
Shows QR
Marks Present
Data is stored in attendance.xlsx based on month and subject code
STOP

12. GitHub Repository Link

The entire project is hosted on GitHub.
Repository Link: https://github.com/SandipanSasmal/Attendance-Tracker-Python-Project

13. Future Improvements

- **Database Integration:**
  - Replace Excel sheets with a relational database for scalability.
- **Enhanced Analytics:**
  - Add graphical reports and visual summaries.
- **Mobile App:**
  - Develop a mobile application for QR code scanning and data access.
- **Customization:**
  - Allow customization of QR code content and appearance.