ATTENDANCE MANAGEMENT SYSTEM WORKFLOW REPORT

NGO: Akshar Paaul

PRN	Name
22070122007	Abhishek Rajput
22070122029	Armaan Jeswani
22070122049	Gauri Deoghare
22070122059	Malay Doshi

CONTENT

1. Introduction
2. Requirement Gathering
3. Tech Stack
4. Relational Schema
5. Workflow
6. Flowchart
7. Conclusion

INTRODUCTION

Akshar Paul NGO is committed to improving literacy and providing support to underprivileged children. Akshar Paul targets children of pavement dwellers and those from families living at construction sites, who often miss out on formal education due to their transient lifestyles and lack of resource.Akshar Paul NGO faces challenges in efficiently tracking student attendance, impacting their ability to support underprivileged children's education. To address this, the NGO plans to implement a biometric attendance management system for accurate monitoring and use Power BI for comprehensive attendance reports. This integration aims to enhance educational outcomes by ensuring consistent student participation and providing actionable insights for better support.

REQUIREMENT GATHERING

The system will gather attendance data from biometric devices, exporting it as Excel files. Python will handle preprocessing tasks, such as cleaning and formatting the data, before it is stored in a MySQL database. Finally, Power BI will be used to generate reports, providing insights into attendance patterns and enabling data-driven decision-making.

TECH STACK

1. **Excel:** For data collection from a biometric machine.
2. **Python:** For data preprocessing.
3. **MySQL:** For data storing.
4. **Power BI:** For data visualisation, representation and report making.

RELATIONAL SCHEMA

The below are the normalised tables according to the data from the excel sheets provided:

1. Student Table

This table holds the basic information about students.

COLUMN NAME	DATA TYPE	DESCRIPTION
student_id	INT	Primary key
name	VARCHAR	Name of the student
class	VARCHAR	Class
div	VARCHAR	Division

2.Attendance Table

This table records the attendance information for each student.

Column name	Data type	Description
attendance_id	INT	Primary key
student_id	INT	Foreign key referencing 'Student Table'
date	DATE	Date of attendance
status	VARCHAR	Attendance status(Present,Absent)
late_minutes	INT	Minutes late(if any)
early_minutes	INT	Minutes early(if any)

3.Teacher Table

This table holds the basic information about teachers

Column_name	Data type	Description
teacher_id	INT	Primary Key
name	VARCHAR	Name of teacher
clss	VARCHAR	Class they are assigned to(optional)

4.Teacher attendance table

This table records the attendance information for each teacher.

Column_name	Data Type	Description
attendance_id	INT	Primary key
teacher_id	INT	Foreign key referencing 'Teacher Table'
date	DATE	Date of attendance
status	VARCHAR	Attendance status(Present,Absent)
late_minutes	INT	Minutes late(if any)
early_minutes	INT	Minutes early(if any)

5.Teacher Shift table

This table manages the shifts assigned to each teacher.

Column Name	Data Type	Description
shift_id	INT	Primary Key
teacher_id	INT	Foreign key referencing 'teacher table'
shift_number	INT	Shift number(e.g. 1,2)

6.Log Table

This table records every entry and exit of students.

Column Name	Data type	Description
log_id	INT	Primary Key
student_id	INT	Foreign key referencing "Student table"
datetime	DATETIME	Date and time of the entry/exit
mode	VARCHAR	Mode of entry(In/Out)

WORKFLOW

Below are the workflow steps for our attendance report generation

1. System Setup

- **Data Sources Identification:**

  Attendance Data Sources:

  1. **Biometric Devices**: Attendance data is captured through biometric devices i.e fingerprint scanners.

  Data Update Frequency:

  1. **Monthly Updates**: Attendance data from biometric devices will be synced and updated in the database monthly.

2. Database Design in MySQL:

- **MySQL Database Setup:**
  1. Set up a **MySQL** database to store attendance records, student and teacher information, and shift details.

- **Monthly Data Update Process:**
  1. At the end of each month, biometric data from devices is synced with the **MySQL** database.
  2. Attendance records for both students and teachers are inserted/updated in the respective tables.

- **Indexes and Performance:**
  1. Add indexes on `date`, `student_id`, and `teacher_id` to improve query performance.
  2. Ensure that foreign key constraints maintain data integrity across tables.

- **Backup and Security:**
  1. Implement daily backups of the MySQL database to avoid data loss.
  2. Ensure that sensitive data, such as biometric identifiers, is securely stored or encrypted if applicable.

3. Data Processing

- **ETL Process Overview:**

  Extract: Collect data from Excel sheets generated by the machine.

  Transform: Clean, format, and preprocess the data for further analysis and report.

  Load: Insert the processed data into the MySQL database for storage .

  1. Use Python scripts with libraries like `pandas` and `openpyxl` or tools like MySQL Workbench's Import Wizard to load the Excel data into the MySQL tables.
  2. The data will be mapped to the appropriate fields in the `Student_Attendance` and `Teacher_Attendance` tables.

4. Data Transformation

- **Attendance Patterns and Anomalies:**

  Analyse attendance records to detect patterns such as regular absenteeism or frequent late arrivals.

  Identify anomalies, such as unusually long or short working hours, or missing attendance records.

5. ETL Automation

- **Automate the ETL Process:** Set up scheduled tasks or cron jobs to automate the extraction of data from the Excel sheets and the transformation and loading processes.
- **Error Handling:** Implement error handling mechanisms to manage issues during data import or transformation, such as missing data or incorrect formats.

6. Data Cleansing

- **Duplicate Removal:** Ensure there are no duplicate records in the attendance tables.
- **Standardisation:** Ensure that data formats (e.g., date formats, time formats) are standardised across all records.

7. Backup and Security

- Backup the transformed and validated data to ensure that no data is lost during the ETL process.
- Ensure that sensitive data, such as names and attendance details, is securely stored in the MySQL database.

8. Power BI Integration

- **Data Import:** Connect Power BI to the database or use Power Query to fetch the data. Schedule data refreshes in Power BI to keep the reports up-to-date.
- **Data Modeling:** Create relationships between tables in Power BI (e.g., Teachers and Students). Create calculated columns or measures (e.g., total hours worked, total days present).
- **Visualisation:**

Dashboard Design:

1. **Overall Attendance Dashboard**:
   - **KPIs**: Display key performance indicators such as total present, absent, late entries, and early exits.
   - **Bar Charts**: Show daily or weekly attendance trends for students and teachers.
   - **Pie Charts**: Illustrate the distribution of attendance statuses (e.g., Present, Absent, Late, Early) for both students and teachers.

Class-wise Attendance:

- **Heat Maps/Bar Charts**: Visualise attendance trends across different classes or shifts. This helps understand how different groups perform in terms of attendance.

Teacher/Student Attendance Reports:

- **Detailed Reports**: Provide detailed reports showing individual attendance records for each teacher and student, with filters to drill down into specific individuals or date ranges.
- **Table Views**: Show daily attendance data, including late entries and early exits.

Monthly Trends:

- **Line Charts**: Display trends in attendance, late entries, and absences over several months. This helps identify seasonal patterns or recurring issues.

Leave Analysis:

- **Pie Charts/Donut Charts**: Illustrate the types of leaves taken by teachers, such as sick leave, personal leave, or other authorised absences.
- **Comparison Charts**: Compare leave statistics between departments or classes.

Gender-wise Attendance Visualization:

- **Bar or Pie Chart**: Create a visualisation that compares the total number of girls and boys present on any given day or over a specific period.
- **Interactive Slicers**: Allow users to filter the data by class or date range to analyse attendance patterns by gender.

Report Publishing:

- **Publishing Reports**: After designing the reports in Power BI Desktop, publish them to the Power BI service to make them accessible to others within the school.
- **Set up Dashboard Access**: Assign access permissions to ensure that only authorised personnel (e.g., school administrators, teachers) can view and interact with the reports.

9. Training Plan

- **Introduction to Power BI**

  Overview: Explain what Power BI is and how it can be used for reporting and data visualisation.
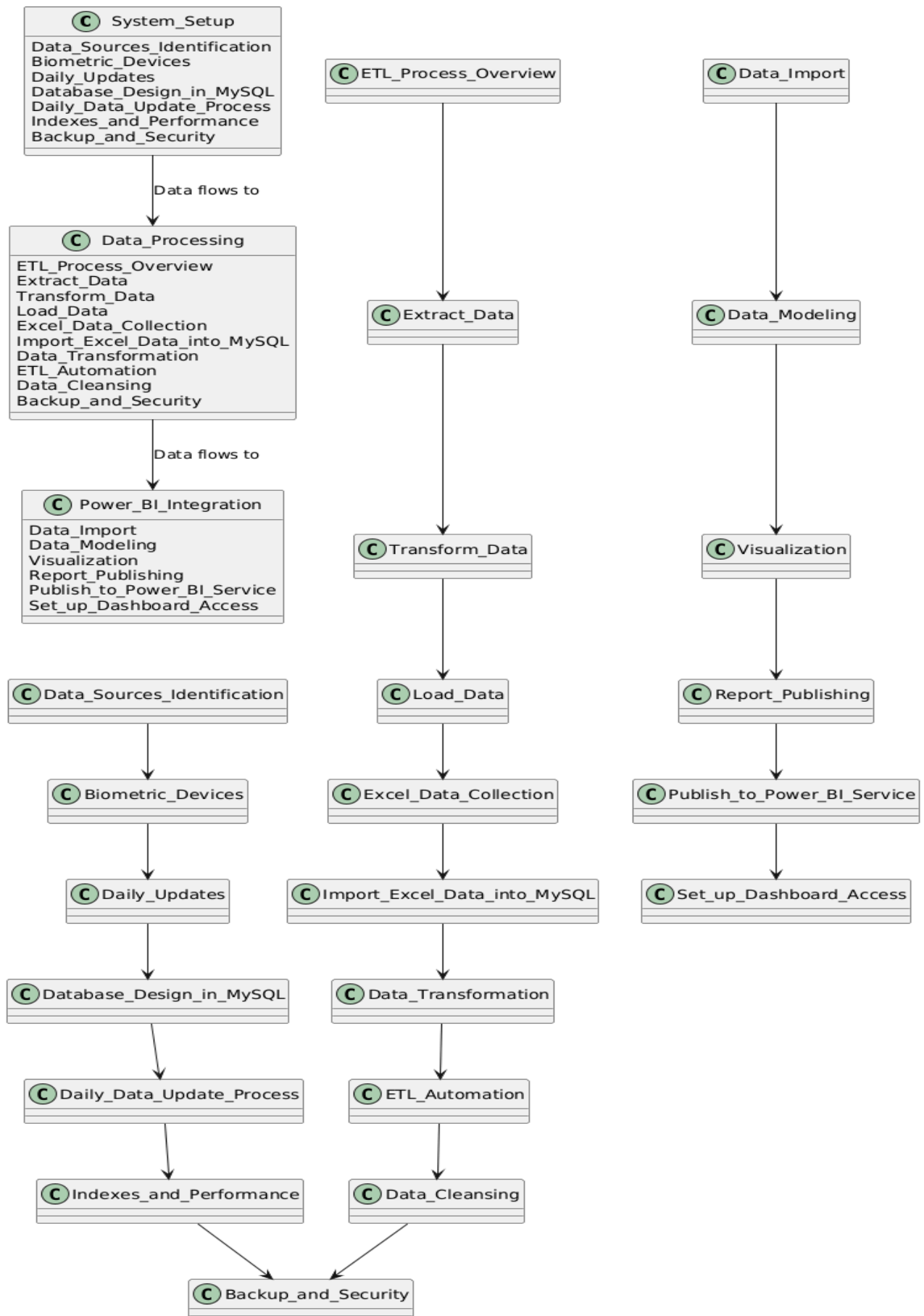
  Navigation: Show how to navigate the Power BI interface, including the workspace, reports, and dashboards.

- **Understanding Reports and Dashboards**

  Reports vs. Dashboards: Explain the difference between reports and dashboards and their use cases.

  Accessing Reports: Demonstrate how to access and open Power BI reports and dashboards.

FLOWCHART

System_Setup
- Data_Sources_Identification
- Biometric_Devices
- Daily_Updates
- Database_Design_in_MySQL
- Daily_Data_Update_Process
- Indexes_and_Performance
- Backup_and_Security

Data flows to

Data_Processing
- ETL_Process_Overview
- Extract_Data
- Transform_Data
- Load_Data
- Excel_Data_Collection
- Import_Excel_Data_into_MySQL
- Data_Transformation
- ETL_Automation
- Data_Cleansing
- Backup_and_Security

Data flows to

Power_BI_Integration
- Data_Import
- Data_Modeling
- Visualization
- Report_Publishing
- Publish_to_Power_BI_Service
- Set_up_Dashboard_Access

ETL_Process_Overview

Extract_Data

Transform_Data

Load_Data

Data_Import

Data_Modeling

Visualization

Report_Publishing

Data_Sources_Identification

Biometric_Devices

Daily_Updates

Database_Design_in_MySQL

Daily_Data_Update_Process

Indexes_and_Performance

Excel_Data_Collection

Import_Excel_Data_into_MySQL

Data_Transformation

ETL_Automation

Data_Cleansing

Publish_to_Power_BI_Service

Set_up_Dashboard_Access

Backup_and_Security

CONCLUSION

The attendance report generation system streamlines the entire process of collecting, processing, storing, and analysing attendance data. By integrating biometric devices, Python for preprocessing, MySQL for storage, and Power BI for reporting, the system ensures accurate, secure, and efficient management of attendance records. This end-to-end solution not only automates data handling but also provides valuable insights through intuitive Power BI dashboards, enabling informed decision-making for school administrators and enhancing overall operational efficiency.