



SYMBIOSIS
INSTITUTE OF TECHNOLOGY, PUNE



SERVICE LEARNING PRESENTATION

Third Year CSE 2022-26



GROUP MEMBERS

ABHISHEK RAJPUT 22070122007
ARMAAN JESWANI 22070122029
GAURI DEOGAHRE 22070122049
MALAY DOSHI 22070122059

TABLE OF CONTENTS

- ▶ INTRODUCTION
- ▶ PROBLEMS STATEMENT
- ▶ WORK ALLOTED
- ▶ OUR SOLUTION
- ▶ TECH STACK AND WORKFLOW
- ▶ PROGRESS OF THE PROJECT
- ▶ WORK LEFT AND CHALLENGES FACED
- ▶ IMAGES
- ▶ IMPORTANT LINKS & REFERENCES

ABOUT OUR NGO

NGO Name: Akshar Paul (Founded 2013)

Focus Area: Improving literacy and providing support to underprivileged children.

Location: Pune, India

Expansion: Extended reach to Nashik and Satara.

Mission: Making education accessible to children in need.

Variety of free services:

1. Including basic literacy and numeracy classes
2. Access to books and learning materials
3. Additional support to help children stay engaged and motivated in their studies
4. Creating a supportive and nurturing learning environment, Akshar Paul aims to give these children the tools they need to build a brighter future.

Target Group:

1. Children of pavement dwellers
2. Children from families living at construction sites

PROBLEM STATEMENT

Attendance is recorded manually, a process that is often prone to errors and inefficiencies. The NGO struggles with manually tracking student attendance, leading to inaccuracies. This is time-consuming and labour-intensive, diverting resources away from core educational activities.

Additionally, the student records from the past 10 years have been manually recorded in registers, which are both difficult to maintain and vulnerable to damage. To address this, these records have been transferred to a more suitable Excel format for easier management and preservation.

WORK ALLOTED

- ▶ Develop an efficient and user-friendly attendance management system.
- ▶ New system should leverage biometric tracking technology to ensure precise and consistent tracking of student attendance.
- ▶ Simplify the process for both the students and teachers by integrating Power BI reports.
- ▶ Digitilize the offline records.
- ▶ Teach students at NGO about english and basics of Computer software and hardware.

OUR SOLUTIONS

Akshar Paul faces challenges in efficiently tracking student attendance and maintaining student data that impacts their ability to support underprivileged children's education and covering larger areas

Solution 01

Biometric attendance management system

Solution 02

Integrating biometric attendance with PowerBI for robust data analytics for comprehensive reports

Solution 03

Digitize the old student records in database and excel

The integrations seeks to deliver improved educational outcomes by ensuring high levels of student engagement and delivering deep actionable insights for better intervention.

TECH STACK AND WORK FLOW

► Designing and Frontend:

- Figma
- HTML, CSS

► Other:

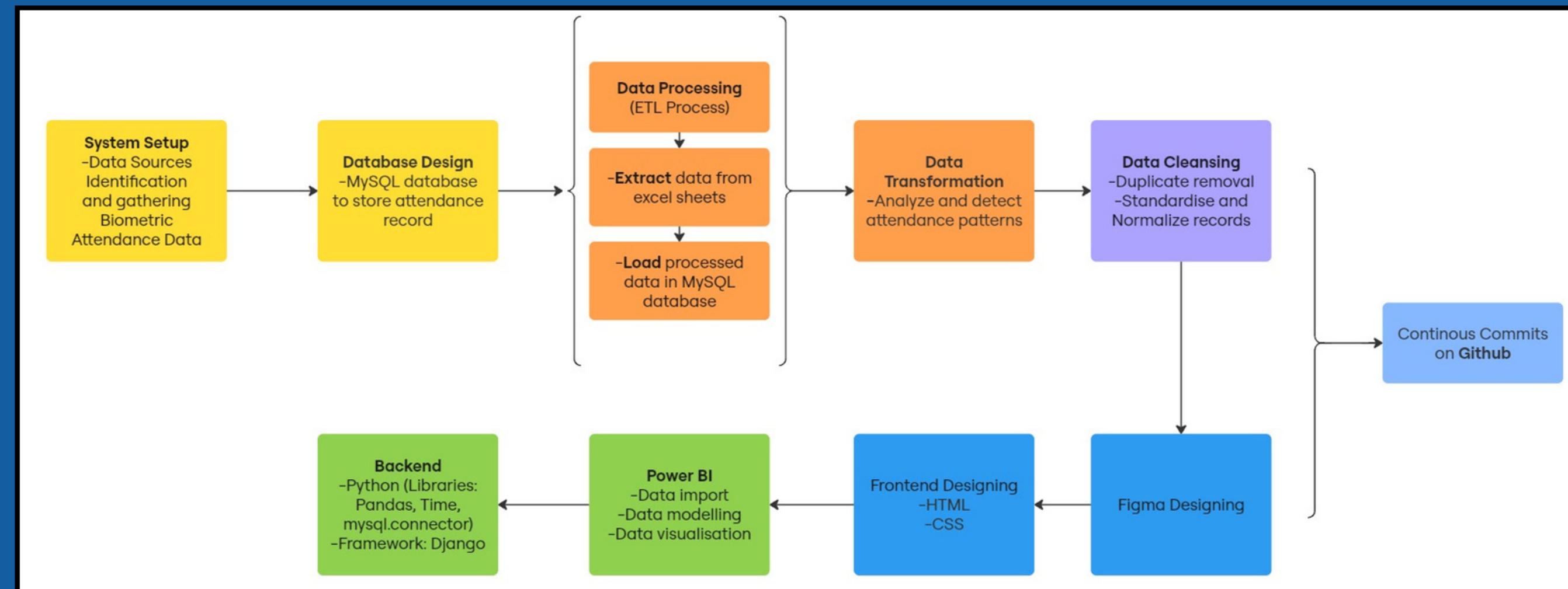
- Excel
- Power BI
- Git/Github

► Backend:

- Python
- Python Libraries: Pandas, time, mysql.connector

► Database:

- MySQL



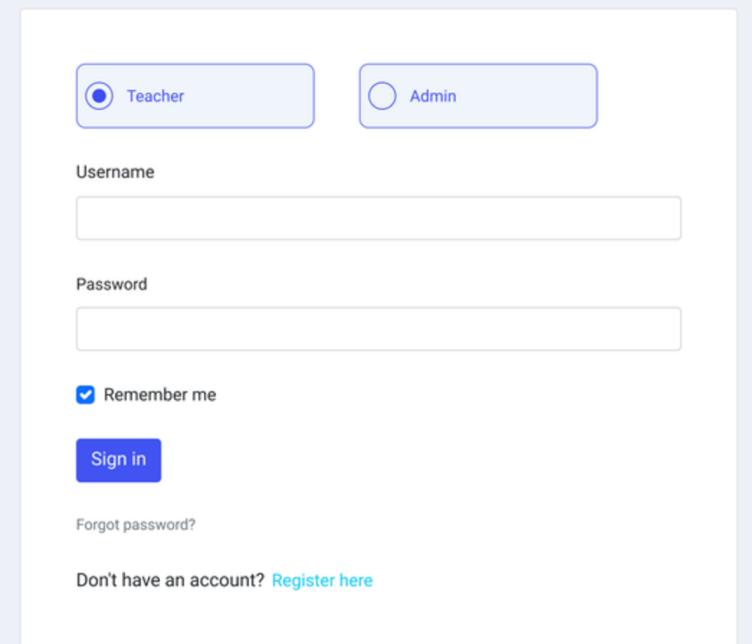
WORK PROGRESS

- ▶ **1.PLANNING AND REPORT:** Entire planning phase focused on system requirements, workflow, user needs and developed a report for the same.
- ▶ **2.DATA DIGITILIZATION:** Digitized manually entered 10 years old student data into the administrator system.
- ▶ **3.FIGMA PROTOTYPE:**A user interface prototype was created using Figma, visually outlining the design and user interactions for the biometric attendance management system.
- ▶ **4.DATABASE DESIGN IN MySQL:** Designed database in MySQL which includes tables for storing attendance records, user information, and biometric data.
- ▶ **5.POWERBI REPORT:** Currently working on report to visualise attendance trends and summaries.

Attendance

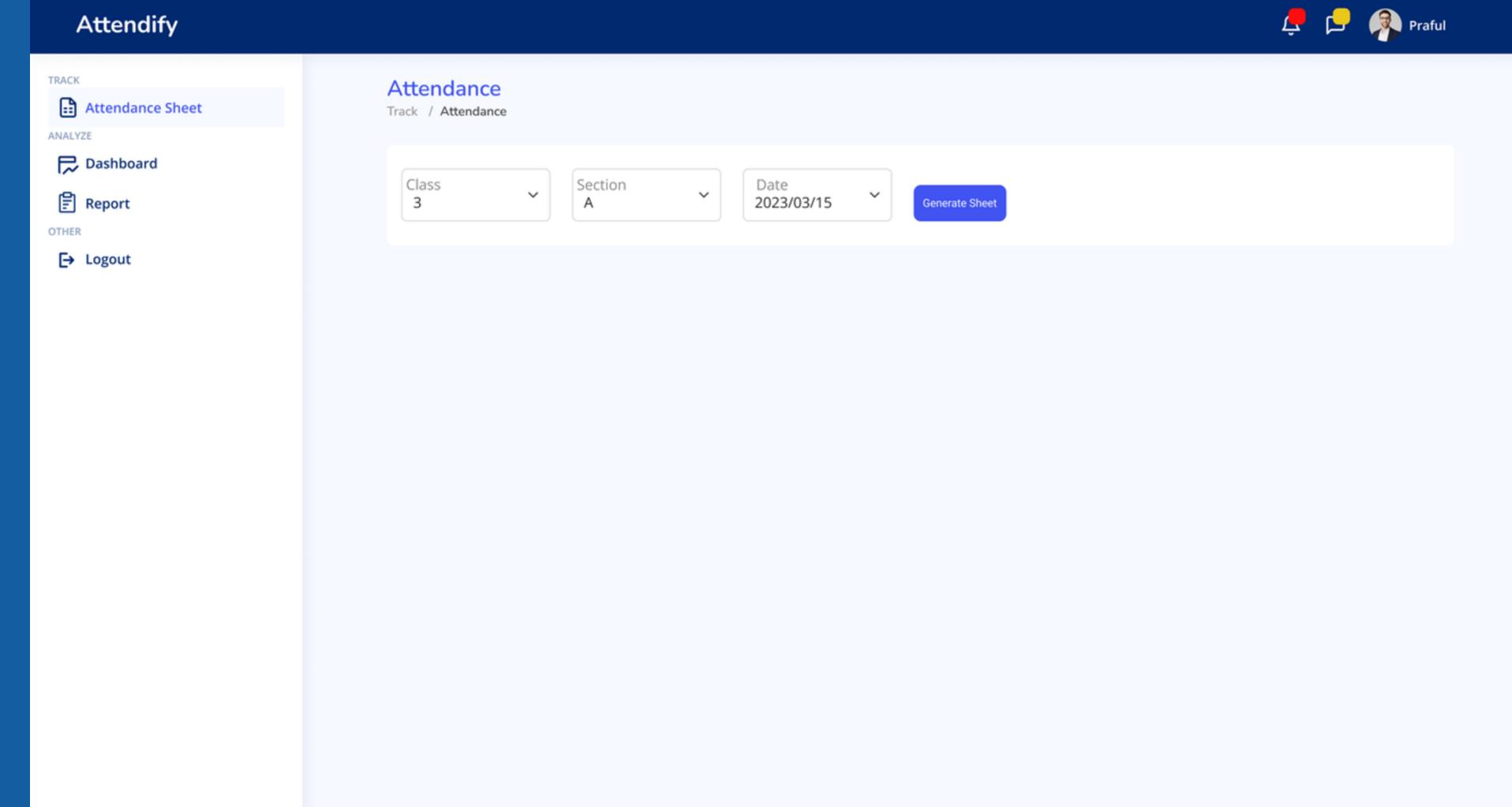
Akshar Paaul

Every child's education should start at his correct age. Bridging the gap between underprivileged children and the main stream of education.



The image shows a login form for the Attendify application. It features two radio button options: "Teacher" (selected) and "Admin". Below these are input fields for "Username" and "Password". A "Remember me" checkbox is checked. A "Sign in" button is present, along with links for "Forgot password?" and "Don't have an account? Register here".

FIGMA DESIGN



The image displays the Attendify application's dashboard. The top navigation bar includes icons for notifications, messages, and user profile (Praful). The left sidebar contains a navigation menu with "Attendance Sheet" (highlighted in blue), "Dashboard", "Report", and "Logout". The main content area is titled "Attendance" and shows filters for "Class 3", "Section A", and "Date 2023/03/15", with a "Generate Sheet" button.

TRACK
[Attendance Sheet](#)

ANALYZE
[Dashboard](#)

Report

OTHER
[Logout](#)

Attendance

Track / Attendance

Teachers

Date
2023/03/15

[Generate Sheet](#)

Students

Class
3

Section
A

Date
2023/03/15

[Generate Sheet](#)

FIGMA DESIGN

TRACK
[Attendance Sheet](#)

ANALYZE
[Dashboard](#)

Report

OTHER
[Logout](#)

Dashboard

Analyze / Dashboard

Present | Today

145
12% increase

Absent | Today

145
12% increase

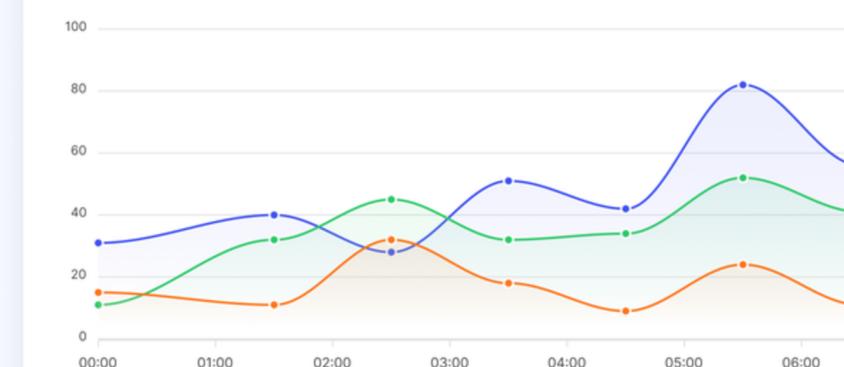
Attendance | This Month

145
12% increase

[Generate Report](#)

Reports

/Today



WORK PROGRESS IN PYTHON

9/14/24, 11:52 PM attendance.py

```
attendance.py

1 import pandas as pd
2 import mysql.connector
3 from mysql.connector import errorcode
4 from datetime import datetime
5
6 # Establish the database connection
7 db_config = {
8     'user': 'root',
9     'password': 'armaan',
10    'host': 'localhost',
11    'database': 'attendance_db'
12 }
13
14 try:
15     conn = mysql.connector.connect(**db_config)
16     cursor = conn.cursor()
17
18     # Read the Excel files
19     student_df = pd.read_excel(r"D:\service learning\student attendance.xlsx")
20     teacher_df = pd.read_excel(r"D:\service learning\teacher attendance.xlsx")
21
22     # Debug: Check if data is being read correctly
23     print("Student DataFrame head:\n", student_df.head())
24     print("\nTeacher DataFrame head:\n", teacher_df.head())
25
26     # Inserting into Class Table
27     class_data = student_df[['ClassName']].drop_duplicates()
28     print("\nInserting into Class table:")
29     for _, row in class_data.iterrows():
30         print(f"Inserting: {row['ClassName']}")
31         cursor.execute(
32             "INSERT INTO Class (ClassName, Year) VALUES (%s, '2024') ON DUPLICATE KEY UPDATE
ClassNames=VALUES(ClassName);",
33             (row['ClassName']),
34         )
35     conn.commit()
36
37     # Inserting into Student Table
38     print("\nInserting into Student table:")
39     for _, row in student_df.iterrows():
40         if pd.isna(row['StudentID']) or pd.isna(row['ClassName']):
41             print(f"Skipping student record due to missing data: {row}")
42             continue
43         print(f"Inserting Student: {row['StudentName']} in Class: {row['ClassName']}")
44         cursor.execute(
45             "INSERT INTO Student (StudentID, StudentName, ClassID) VALUES (%s, %s, (SELECT
ClassID FROM Class WHERE ClassName=%s)) ON DUPLICATE KEY UPDATE
StudentName=VALUES(StudentName);",
46             (row['StudentID'], row['StudentName'], row['ClassName'])
47         )
48     conn.commit()
49 
```

localhost:53894/75f03eb1-70c3-42fc-806e-ae6dabd4c614/ 1/3

Database Connection

9/14/24, 11:52 PM attendance.py

```
attendance.py

50     # Inserting into Teacher Table
51     print("\nInserting into Teacher table:")
52     for _, row in teacher_df.iterrows():
53         if pd.isna(row['TeacherID']):
54             print(f"Skipping teacher record due to missing data: {row}")
55             continue
56         print(f"Inserting Teacher: {row['TeacherName']}")
57         cursor.execute(
58             "INSERT INTO Teacher (TeacherID, TeacherName) VALUES (%s, %s) ON DUPLICATE KEY
UPDATE TeacherName=VALUES(TeacherName);",
59             (row['TeacherID'], row['TeacherName'])
60         )
61     conn.commit()
62
63     # Insert into DateDimension Table
64     all_dates = pd.concat([student_df['Date'],
teacher_df['Date']]).drop_duplicates().sort_values()
65     print("\nInserting into DateDimension table:")
66     for date in all_dates:
67         date_obj = pd.to_datetime(date).date()
68         print(f"Inserting Date: {date_obj}")
69         cursor.execute(
70             "INSERT INTO DateDimension (Date, Year, Month, Day, MonthName, WeekdayName)
VALUES (%s, %s, %s, %s, %s, %s) ON DUPLICATE KEY UPDATE Date=VALUES(Date);",
71             (date_obj, date_obj.year, date_obj.month, date_obj.day, date_obj.strftime('%B'),
date_obj.strftime('%A'))
72         )
73     conn.commit()
74
75     # Inserting into StudentAttendance Table
76     print("\nInserting into StudentAttendance table:")
77     for _, row in student_df.iterrows():
78         if pd.isna(row['StudentID']) or pd.isna(row['Date']):
79             print(f"Skipping student attendance record due to missing data: {row}")
80             continue
81         date_obj = pd.to_datetime(row['Date']).date()
82
83         # Handle missing InTime for absent students
84         in_time = row['InTime'] if row['Status'] == 'Present' and pd.notna(row['InTime'])
else None
85
86         # Get DateID for the current date
87         cursor.execute("SELECT DateID FROM DateDimension WHERE Date=%s", (date_obj,))
88         date_id = cursor.fetchone()[0]
89
90         print(f"Inserting StudentAttendance for StudentID: {row['StudentID']} on Date:
{date_obj}")
91         cursor.execute(
92             """
93             INSERT INTO StudentAttendance (StudentID, DateID, Status, InTime)
VALUES (%s, %s, %s, %s)
94             ON DUPLICATE KEY UPDATE Status=VALUES(Status), InTime=VALUES(InTime);
95             """,
96             (row['StudentID'], date_id, row['Status'], in_time)
97         ) 
```

localhost:53894/75f03eb1-70c3-42fc-806e-ae6dabd4c614/ 2/3

Data Insertion

9/14/24, 11:52 PM attendance.py

```
attendance.py

98     )
99     conn.commit()
100
101    # Inserting into TeacherAttendance Table
102    print("\nInserting into TeacherAttendance table:")
103    for _, row in teacher_df.iterrows():
104        if pd.isna(row['TeacherID']) or pd.isna(row['Date']):
105            print(f"Skipping teacher attendance record due to missing data: {row}")
106            continue
107        date_obj = pd.to_datetime(row['Date']).date()
108
109        # Handle missing InTime and OutTime for absent teachers
110        in_time = row['InTime'] if row['Status'] == 'Present' and pd.notna(row['InTime'])
else None
111        out_time = row['OutTime'] if row['Status'] == 'Present' and pd.notna(row['OutTime'])
else None
112
113        # Get DateID for the current date
114        cursor.execute("SELECT DateID FROM DateDimension WHERE Date=%s", (date_obj,))
115        date_id = cursor.fetchone()[0]
116
117        print(f"Inserting TeacherAttendance for TeacherID: {row['TeacherID']} on Date:
{date_obj}")
118        cursor.execute(
119            """
120            INSERT INTO TeacherAttendance (TeacherID, DateID, Status, InTime, OutTime)
VALUES (%s, %s, %s, %s, %s)
121            ON DUPLICATE KEY UPDATE Status=VALUES(Status), InTime=VALUES(InTime),
OutTime=VALUES(OutTime);
122            """,
123            (row['TeacherID'], date_id, row['Status'], in_time, out_time)
124        )
125    conn.commit()
126
127    print("Data inserted/updated successfully!")
128
129 except mysql.connector.Error as err:
130     if err.errno == errorcode.ER_ACCESS_DENIED_ERROR:
131         print("Something is wrong with your user name or password")
132     elif err.errno == errorcode.ER_BAD_DB_ERROR:
133         print("Database does not exist")
134     else:
135         print(err)
136 finally:
137     cursor.close()
138     conn.close()
139
140 
```

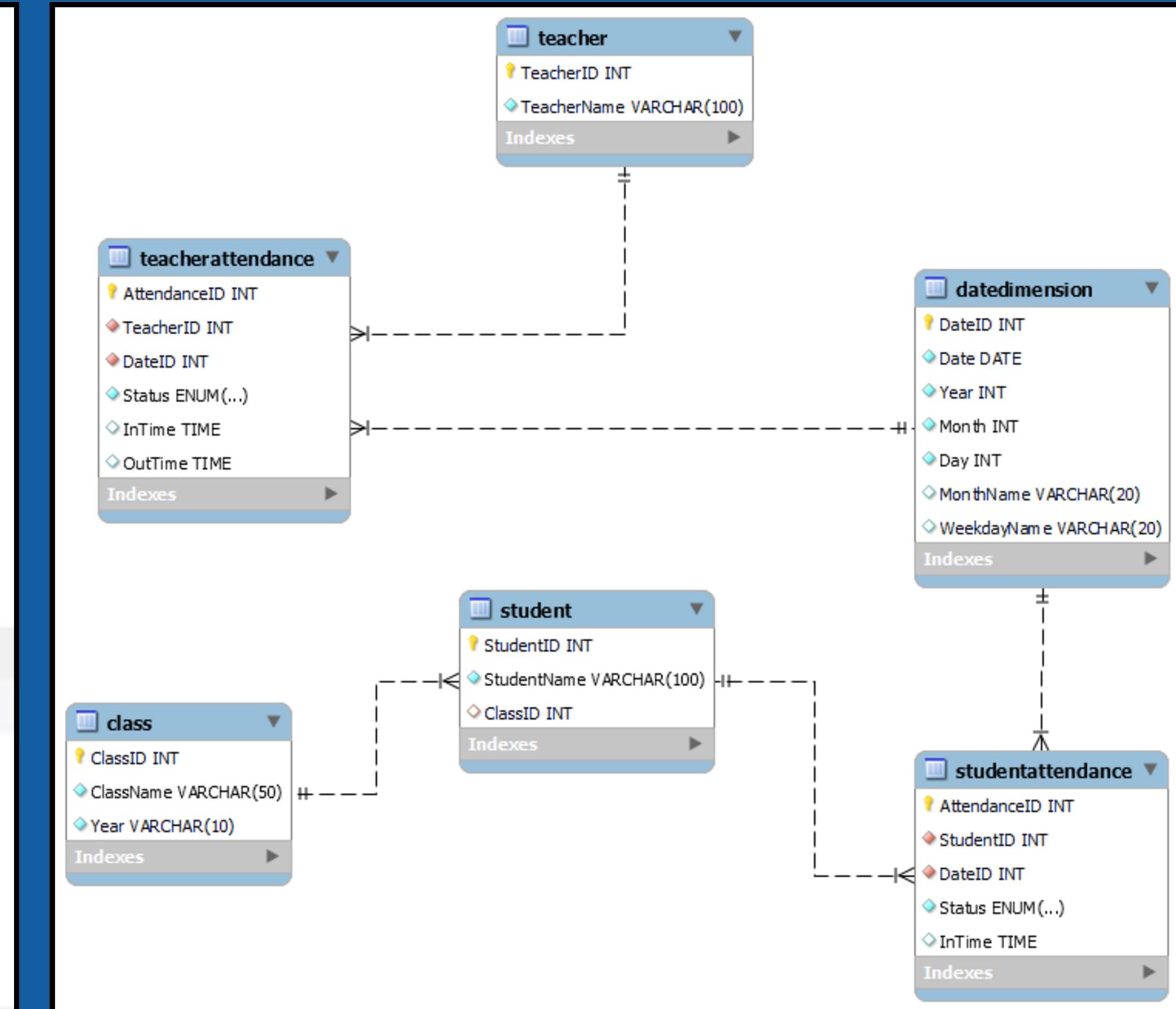
localhost:53894/75f03eb1-70c3-42fc-806e-ae6dabd4c614/ 3/3

Error Handling

WORK PROGRESS IN DATABASE

```
3      s.StudentName,  
4      d.Date  
5  FROM  
6      StudentAttendance sa  
7  JOIN  
8      Student s ON sa.StudentID = s.StudentID  
9  JOIN  
10     DateDimension d ON sa.DateID = d.DateID  
11 WHERE  
12     d.Date = '2024-09-02' -- Filter for 2nd September  
13     AND sa.Status = 'Absent';  
14
```

Result Grid		
StudentID	StudentName	Date
212	Sanya Sharma	2024-09-02
215	Ashwin Nair	2024-09-02
241	Shreya Gupta	2024-09-02
245	Swati Desai	2024-09-02
248	Pritam Patel	2024-09-02



POWER BI REPORT

 Akshar Paaul Teachers Report

Apr 22 May 22 Jun 22 09-04-2022 17-06-2022

89.7% **1.7%**

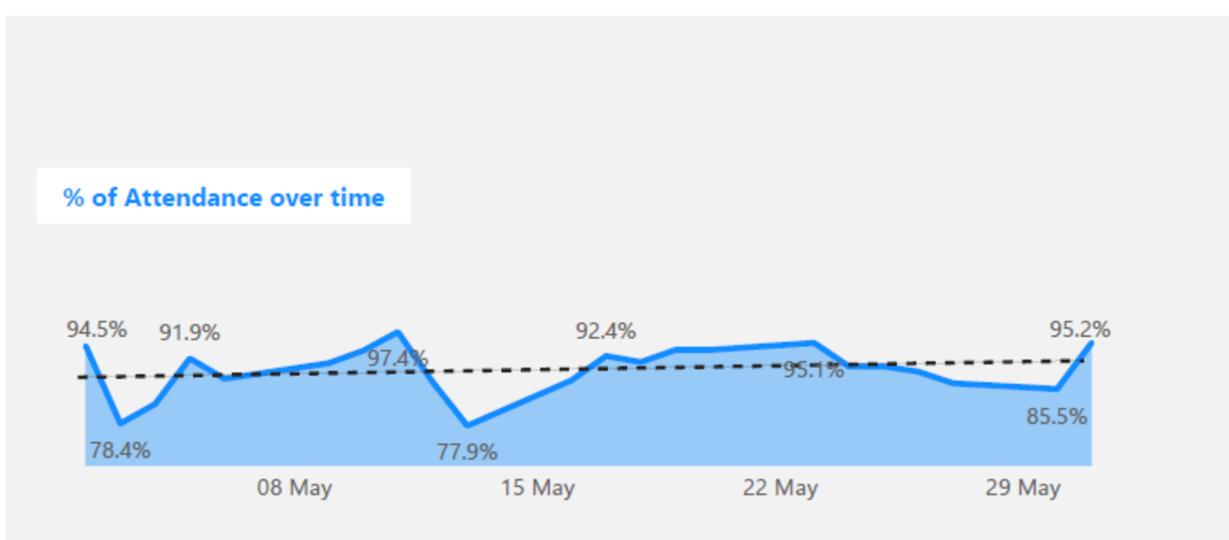
Attendance % SL %

Employee ID	Attendance %	Present Days
Atq -419	59.1%	26
Atq -452	59.1%	13
Atq -385	75.0%	33
Atq -349	77.3%	17
Atq -387	77.3%	17
Atq -408	77.3%	17
Atq -421	77.3%	17
Atq -343	81.8%	18
Atq -393	81.8%	18
Total	89.7%	1548

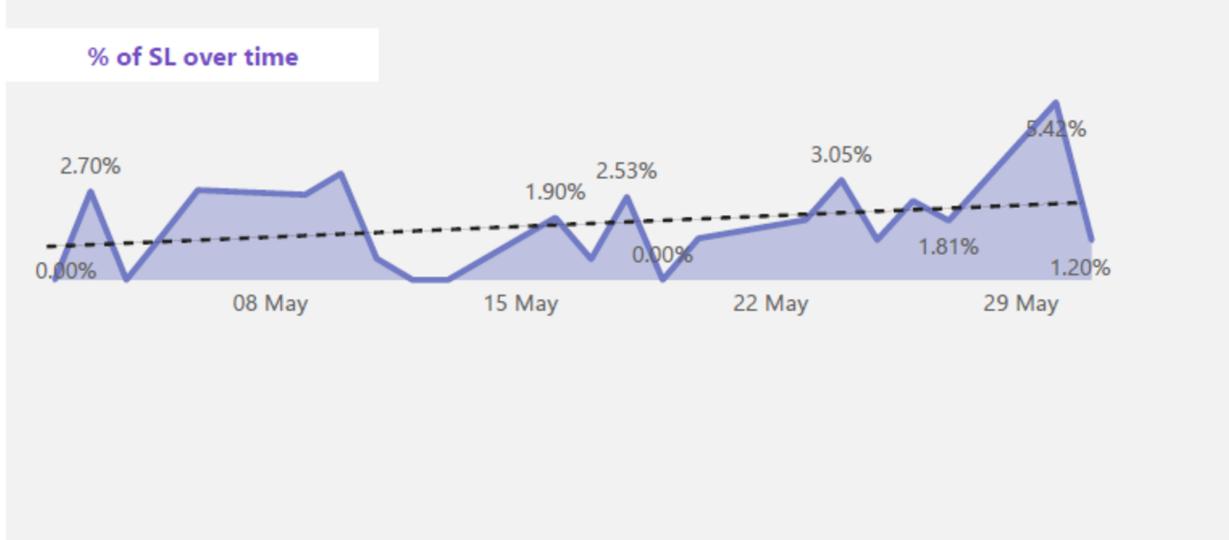
Employee ID	02 May 2022	03 May 2022	04 May 2022	05 M
Atq -331	P	P	WFH	P
Atq -334	P	P	P	P
Atq -335	P	P	P	P
Atq -343	P	P	P	P
Atq -348	P	P	PL	PL
Atq -349	P	P	P	P
Atq -362				
Atq -364				
Atq -366	P	P	P	P
Atq -367	HPL	FFL	PI	WLF
Total	HPL	FFL	HPL	LWF

SL - Sick Leave

% of Attendance over time



% of SL over time



day of week	Attendance %
Mon	90.6%
Thu	90.4%
Wed	90.4%
Tue	90.1%
Fri	86.5%
Total	89.7%

day of week	SL %
Mon	2.41%
Tue	2.15%
Fri	1.44%
Wed	1.12%
Thu	0.96%
Total	1.68%

OLD & NEW RECORDS

शक्षण मडळ									
नमुना नंबर १ (भाग १ नियम ७, पाहा)									
कोणत्या शाळेन व कोणत्या इयत्तेतून आले (नवीन असल्यास तसे लिहावे)	शाळेत नाव दाखल केल्याची तारीख	कोणत्या इयत्तेतून व कोणत्या वर्गात नाव दाखल केले	मातृभाषा	शाळा सोडल्याची तारीख	कोणत्या इयत्तेतून व कोणत्या वर्गात असला शाळा सोडली	आप्यास	वर्तणूक	शाळा सोडल्याचे कारण याची वर्गीकरणी	सर्टिफिकेट प्रावल्यावदल सही व तारीख
New OC	15/6/10	I st	marathi	10/10/12	III rd	Good	Good	Transfer	J-mundi
New	15/6/10	I st	marathi	15/6/17	VII	Good	Good for further education	14/6/17	16/6/17
New	15/6/10	I st	marathi	15/6/17	VII	Good	Good for further education	30/6/17	19/6/17
New	15/6/10	I st	marathi	15/6/17	VII	Good	Good for further education	30/6/17	19/6/17
New	15/6/10	I st	marathi	23/7/15	VI	Good	Good Parental Request	24/7/15	24/7/15

* मुलाने शाळा सोडतेवेळी भरावयाचे

AKSHAR PAAUL STUDENT DATA CLASS 3rd 2022-23

Admission No.	Name	Gender	DOB	Date of Admission	Religion	Category	Mother Tongue	Guardian's Name	Phone No. 1	Phone No. 2	Aadhaar No.
4112	Aarav Deshmukh	M	05-02-2015	10-07-2020	Hindu	Gen	Marathi	Rohan Deshmukh	9823456712	9734567812	7890 1234 5678 9123
4113	Vedika Kulkarni	F	22-08-2014	15-06-2020	Hindu	Gen	Marathi	Swati Kulkarni	9897654321	9854678934	6789 2345 8901 3456
4114	Rohan Sharma	M	30-11-2013	01-06-2021	Hindu	Gen	Hindi	Vivek Sharma	9823678912	9876543210	5678 3456 1234 8765
4115	Simran Kaur	F	10-05-2015	12-07-2020	Sikh	OBC	Punjabi	Harjit Kaur	9876543212	9867543211	2345 6789 9012 5678
4116	Aniket Jadhav	M	20-12-2013	09-06-2020	Hindu	OBC	Marathi	Milind Jadhav	9998765432	9871234567	8901 2345 6789 1234
4117	Priya Rane	F	18-03-2015	18-06-2020	Hindu	OBC	Marathi	Shraddha Rane	9876543214	9812345678	7654 3210 2345 6789
4118	Ishaan Bhatt	M	22-01-2014	05-07-2020	Hindu	Gen	Gujarati	Jignesh Bhatt	9834567812	9821678932	5432 8765 1234 0987
4119	Tanvi Joshi	F	05-09-2013	25-06-2021	Hindu	Gen	Marathi	Neha Joshi	9887654321	9745678910	8901 8765 4321 5678
4120	Kabir Khan	M	13-04-2014	15-06-2020	Muslim	Gen	Urdu	Akhtar Khan	9123456789	9734567891	3456 5678 1234 2345
4121	Zoya Sheikh	F	27-07-2013	01-07-2021	Muslim	OBC	Urdu	Salman Sheikh	9876549876	9867543212	4567 8901 2345 6789
4122	Yash Patil	M	02-02-2014	08-06-2020	Hindu	OBC	Marathi	Vishal Patil	9823456712	9923456781	6789 2345 7890 3456
4123	Ananya Thakur	F	29-09-2015	20-07-2020	Hindu	Gen	Hindi	Pooja Thakur	9867543214	9876543201	7890 3456 5678 1234
4124	Aditya Gokhale	M	19-03-2014	13-06-2020	Hindu	Gen	Marathi	Sanjay Gokhale	9898765432	9812345670	8901 3456 8765 4321
4125	Sanya Desai	F	08-08-2013	10-07-2021	Hindu	Gen	Gujarati	Smita Desai	9876543298	9734567898	5678 1234 8901 5678
4126	Arnav Bhosale	M	17-06-2015	19-06-2020	Hindu	OBC	Marathi	Shrikant Bhosale	9898765431	9734567810	3456 7890 2345 6789
4127	Kavya Nair	F	14-02-2014	12-06-2021	Hindu	Gen	Malayalam	Vinod Nair	9123456780	9745678912	8901 4321 6789 1234
4128	Dhruv Gaikwad	M	09-05-2015	08-07-2020	Hindu	OBC	Marathi	Mahesh Gaikwad	9734567890	9823456713	7654 2345 0987 3456
4129	Shreya Menon	F	27-10-2013	20-07-2021	Hindu	Gen	Malayalam	Anjali Menon	9765432109	9834567812	4321 7654 1234 0987
4130	Omkar Shinde	M	02-12-2014	25-06-2020	Hindu	OBC	Marathi	Sunil Shinde	9876543219	9745678912	6543 0987 2345 6789
4131	Maya Kapoor	F	23-01-2015	17-06-2020	Hindu	Gen	Hindi	Suman Kapoor	9723456710	9812345672	7890 1234 0987 5678

WORK LEFT

- Currently working on developing a front-end to create an efficient and user-friendly interface.
- Integrating the MySQL Database, Python frontend and Django Backend.

CHALLENGES FACED

- One of the significant challenges faced was continuously changing requirements of the NGO where the system had to be re-evaluated and modified several times to accommodate the NGO's requests.
- Integrating different components of the system became more complex.
- These evolving requirements not only impacted the initial project timeline but also created a need for redesigning certain features to align with the updated scope.

IMAGES



IMAGES



IMPORTANT LINKS

- **FIGMA:**
[https://www.figma.com/design/x7W0kX0EWrjbKygvPwo8ev/Attendance-management-system-\(Community\)?t=NjrfwPXmuKMJVyL6-0](https://www.figma.com/design/x7W0kX0EWrjbKygvPwo8ev/Attendance-management-system-(Community)?t=NjrfwPXmuKMJVyL6-0)
- **GITHUB:**
https://github.com/Abhishek-2502/Attendance_Management_System_NGO
- **FLOWCHART:**
<https://miro.com/app/board/uXjVLegAC9M=/>

REFERENCES

- <https://github.com/pnavin9/HR-Analytics>
- <https://github.com/topics/power-bi-dashboard?l=css>

THANK YOU!