UIT2739 – FULL STACK DEVELOPMENT

A PROJECT REPORT

on

Bus Management System

Submitted by
Sachin M A – 3122225002108
Srinivas S-3122225002135
Vansh Sharma-3122225002151

Department of Information Technology
Sri Sivasubramaniya Nadar College of Engineering
(An Autonomous Institution, Affiliated to Anna University)
Rajiv Gandhi Salai (OMR), Kalavakkam – 603 110

NOVEMBER 2025

SRI SIVASUBRAMANIYA NADAR COLLEGE OF ENGINEERING

Department of Information Technology

CERTIFICATE

Certified that this project titled **"Bus Management System"** is the bonafide work of **"Sachin M A(3122225002108), Srinivas S (3122225002135), Vansh Sharma(3122225002151)",** and is submitted for project review on **27 November 2025.**

Place : Kalavakkam

Date : **Internal Examiner**

TABLE OF CONTENTS

Project Overview

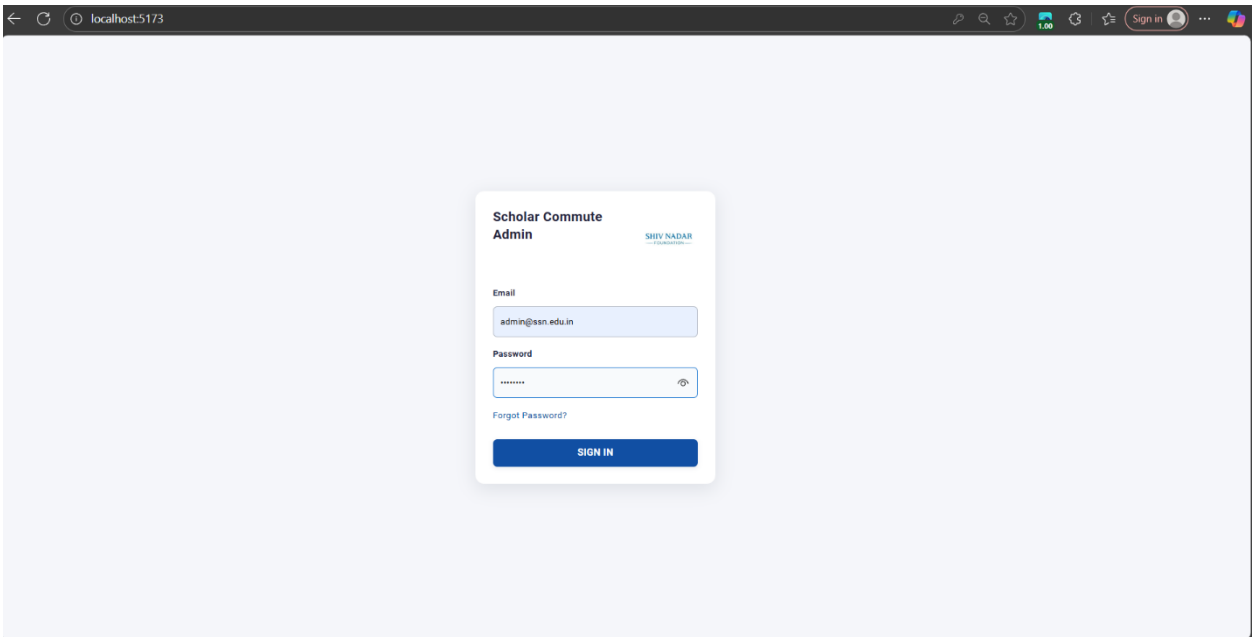
The Bus Management and Tracking System is designed to automate student transit monitoring and attendance across the college transportation network. The system integrates ML-based facial recognition using Raspberry Pi devices, GPS-based live bus tracking, route management, and role-specific web applications for both administrators and students. The solution eliminates manual attendance processes, improves real-time visibility, enhances operational efficiency, and provides students with accurate information on bus availability and movement.

Project Requirements

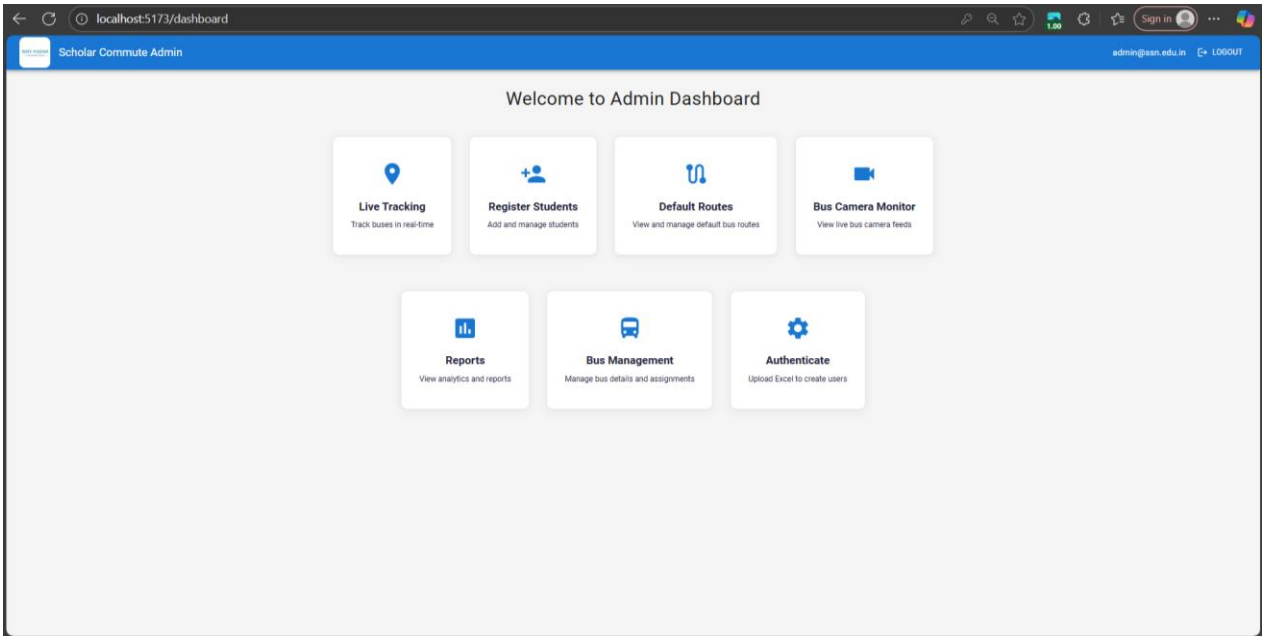
2.1 Wireframes / Screenshots

1) Admin Interface

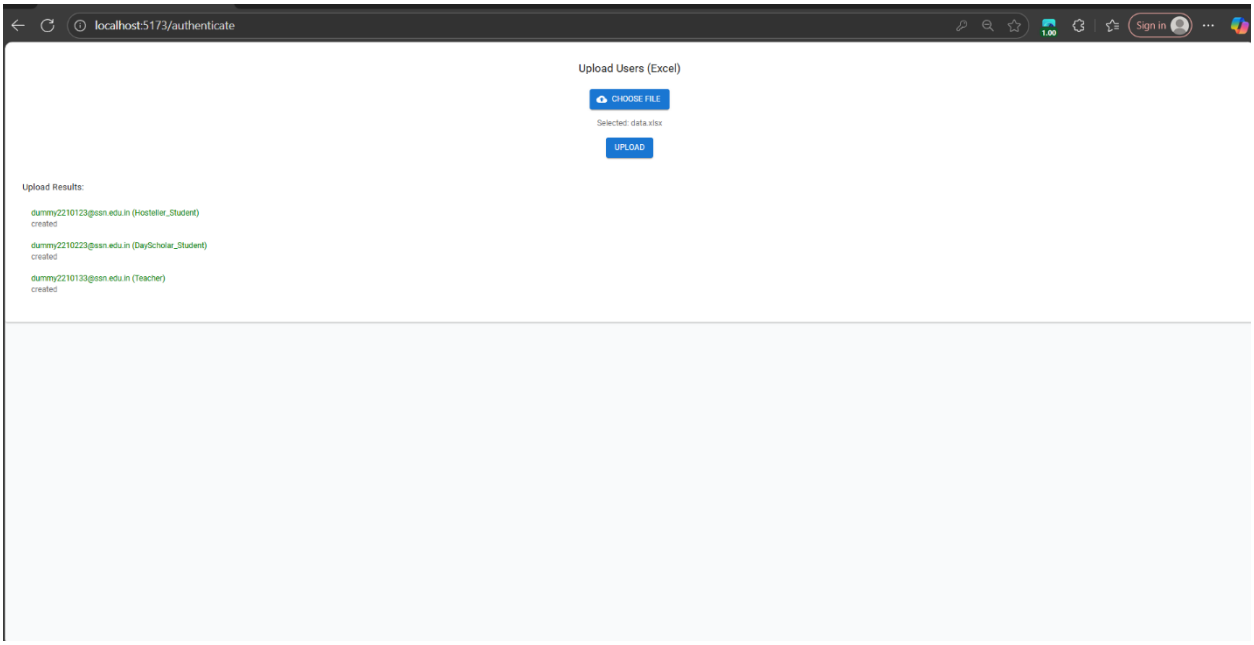
a. Login Page



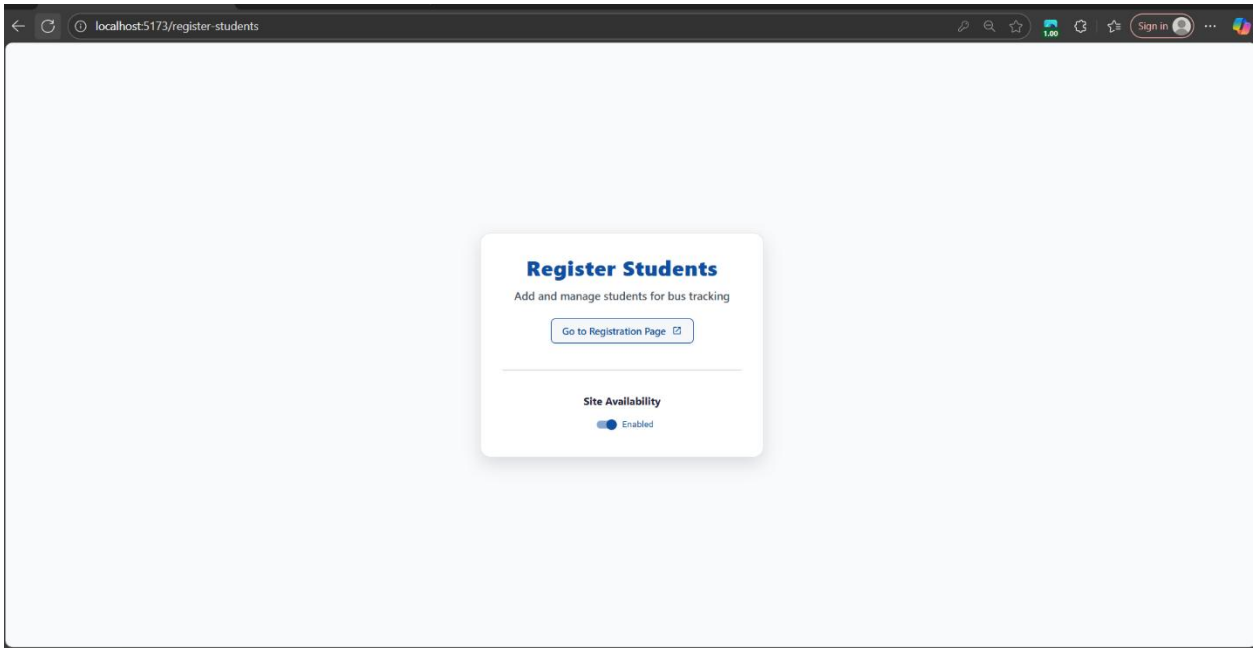
b. Dashboard



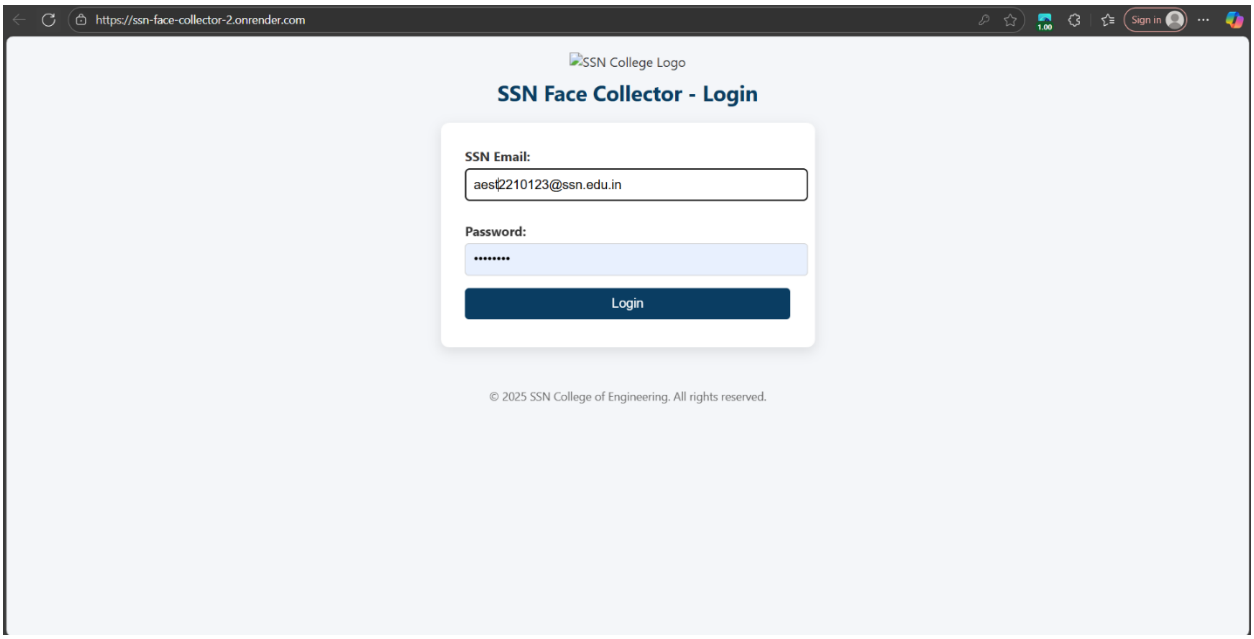
c. Authenticate Students



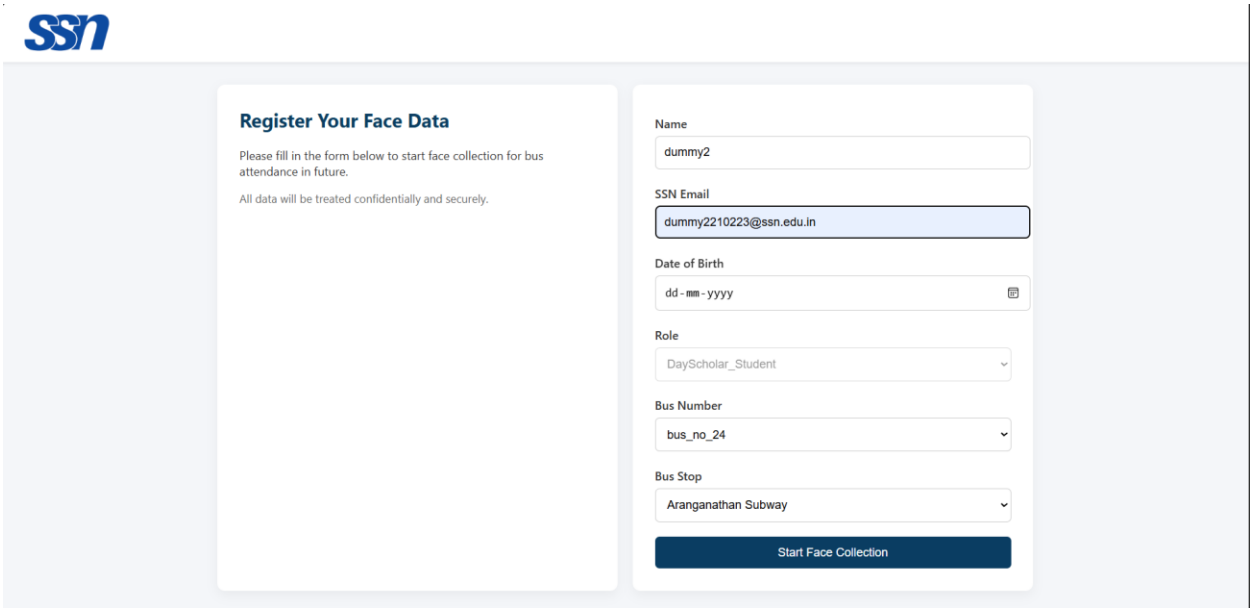
d. Registration Page

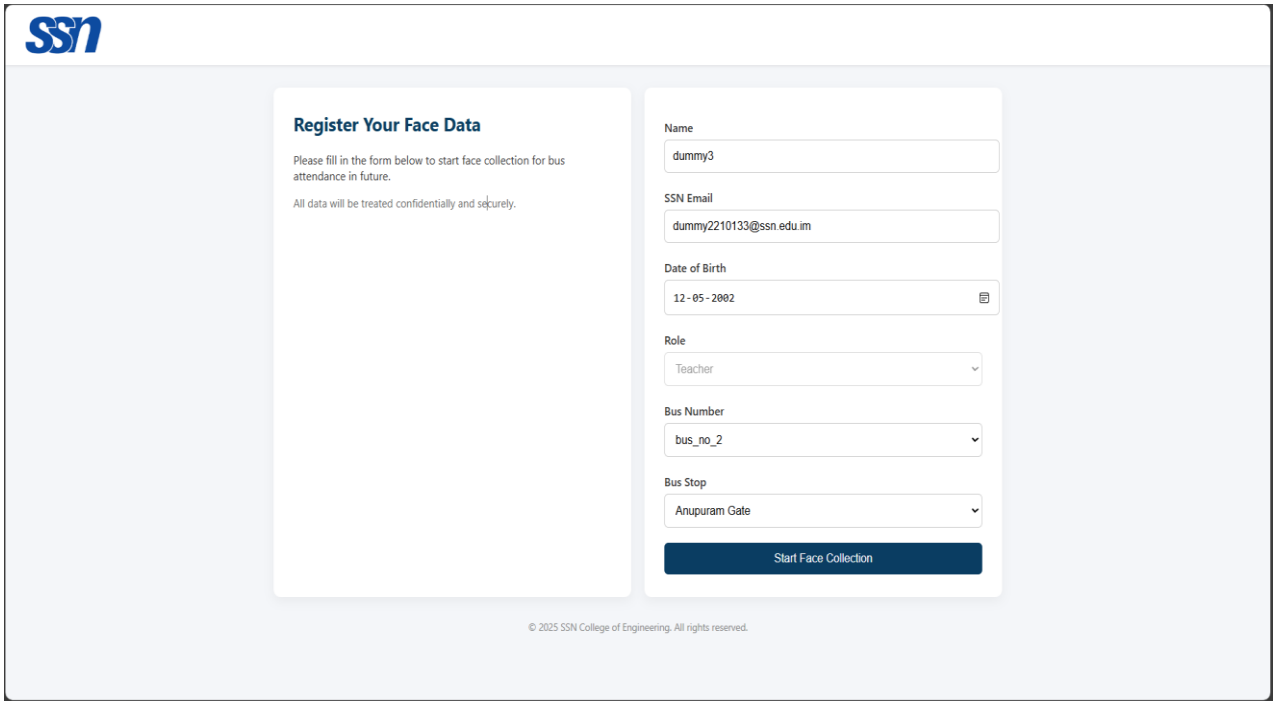


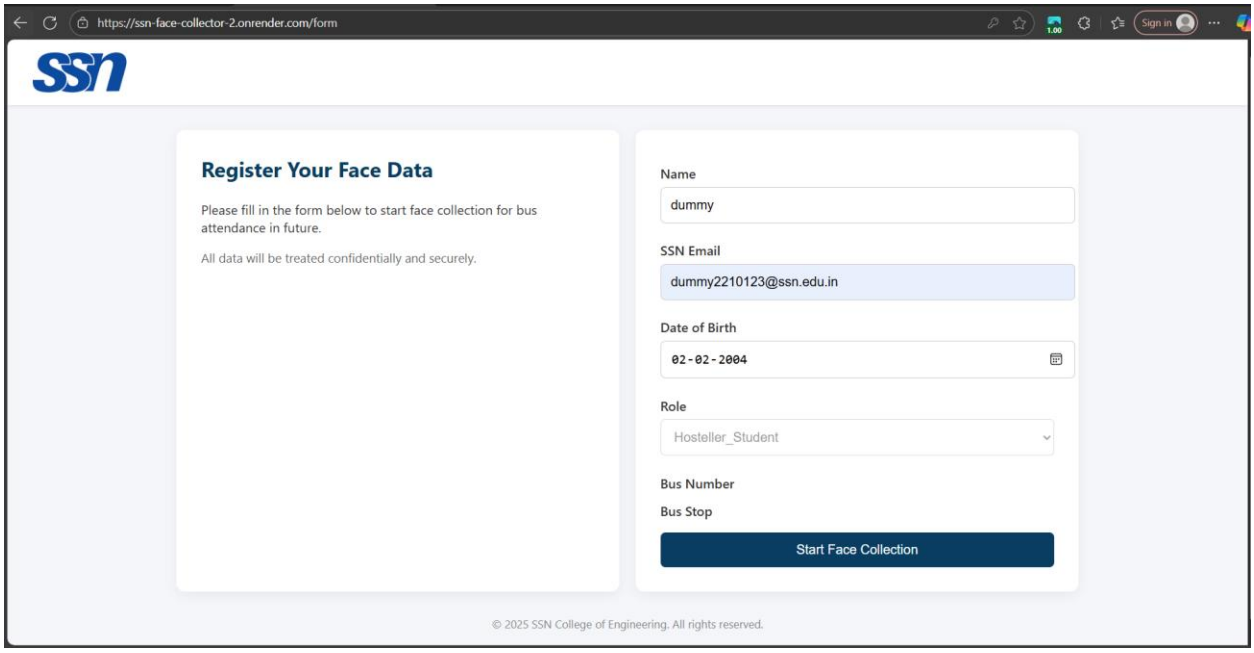
i. Login of Registration Page

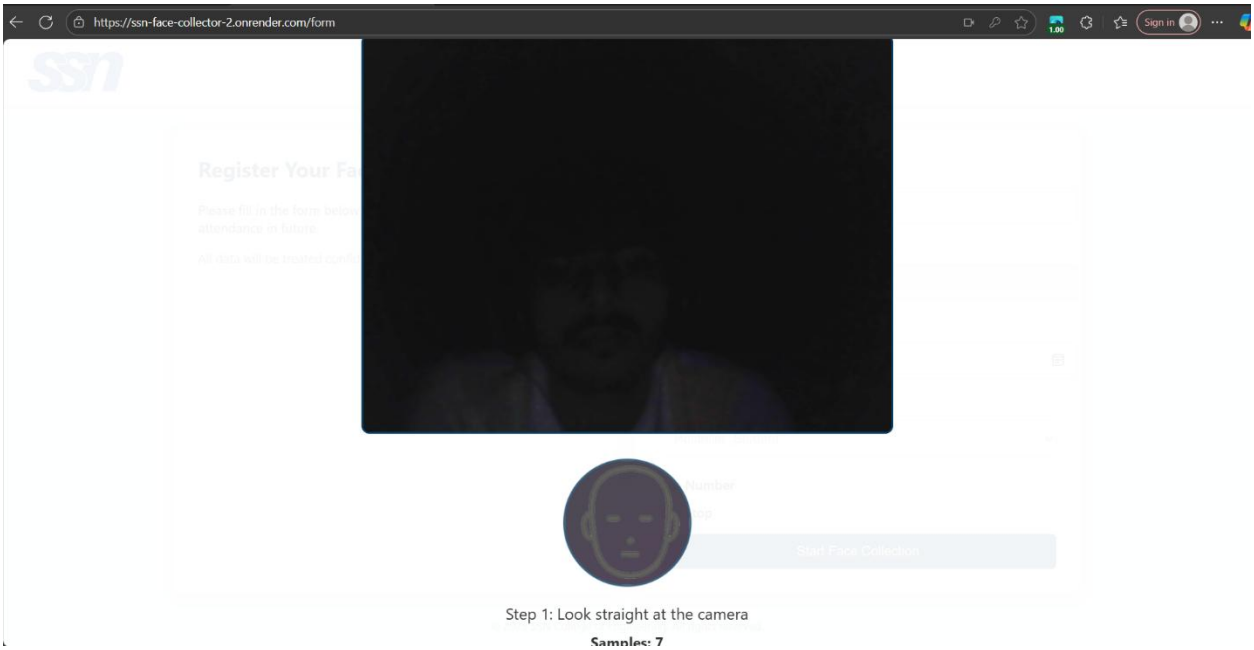


ii. Student Detail collection Page

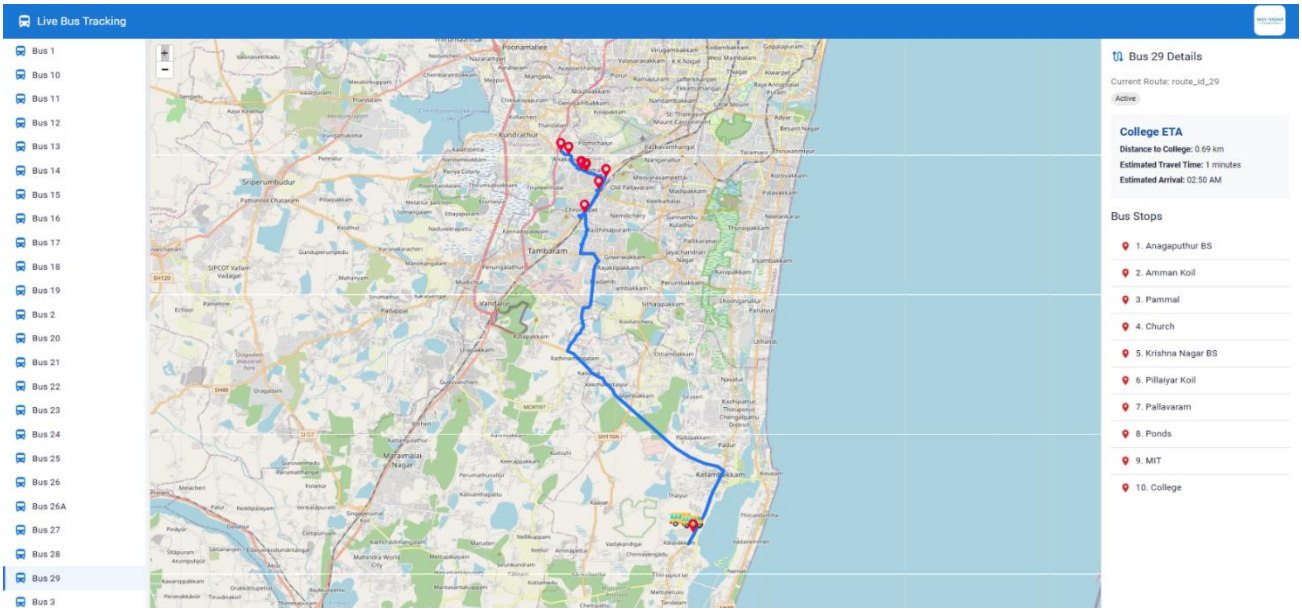





https://ssn-face-collector-2.onrender.com/form
Sign in
Register Your Face Data
Please fill in the form below to start face collection for bus attendance in future.
All data will be treated confidentially and securely.
Name
dummy
SSN Email
dummy2210123@ssn.edu.in
Date of Birth
02-02-2004
Role
Hosteller_Student
Bus Number
Bus Stop
Start Face Collection
© 2025 SSN College of Engineering. All rights reserved.

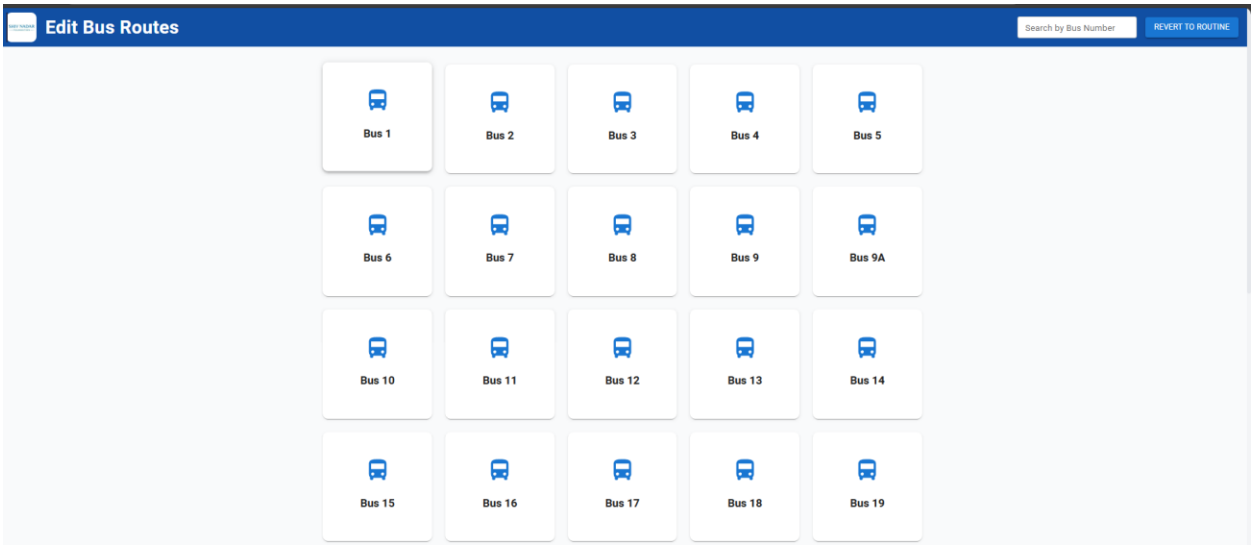

https://ssn-face-collector-2.onrender.com/form
Sign in
Start Face Collection
Step 1: Look straight at the camera
Samples: 7

e. Live Tracking Page

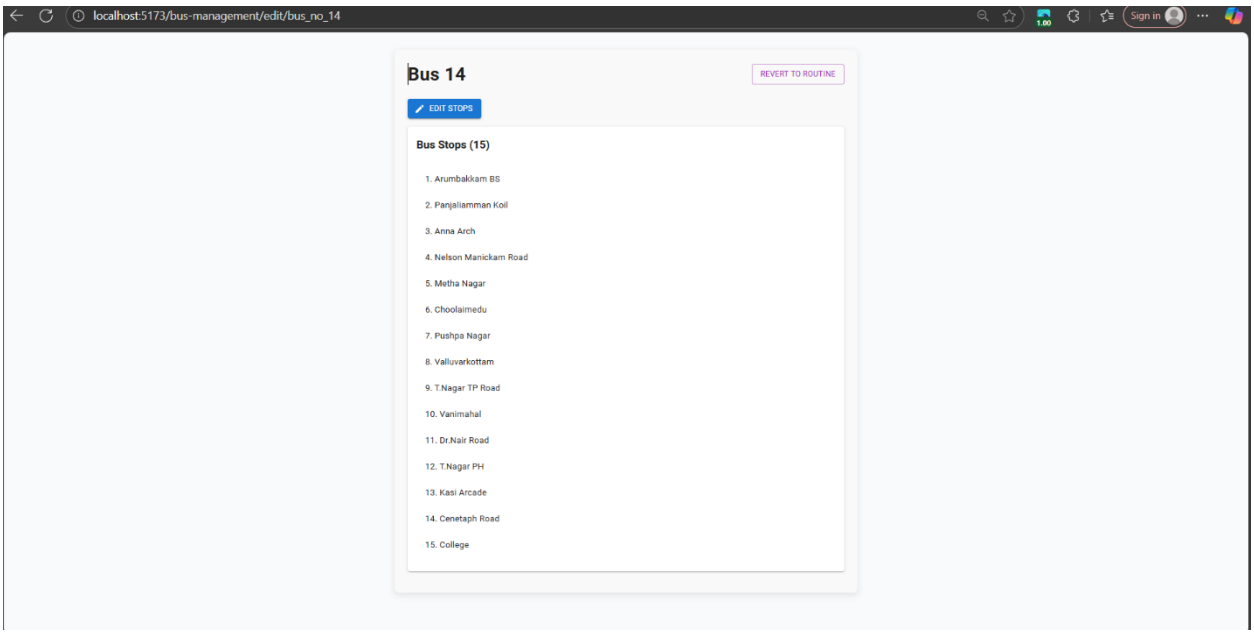


f. Bus Management Page

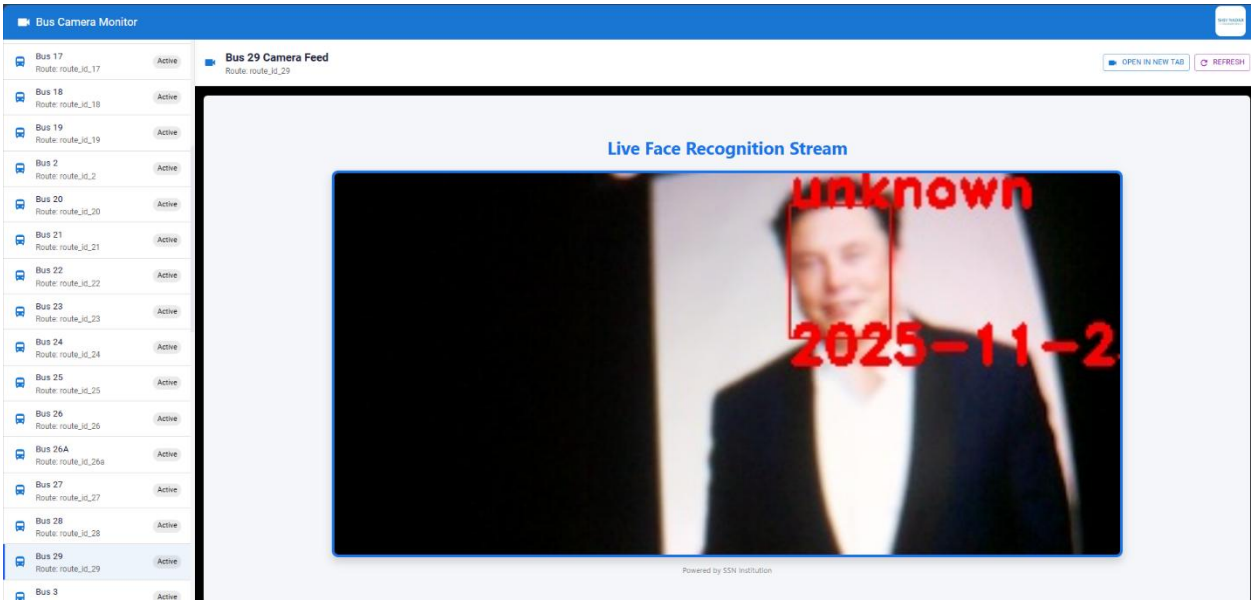
i. View Buses



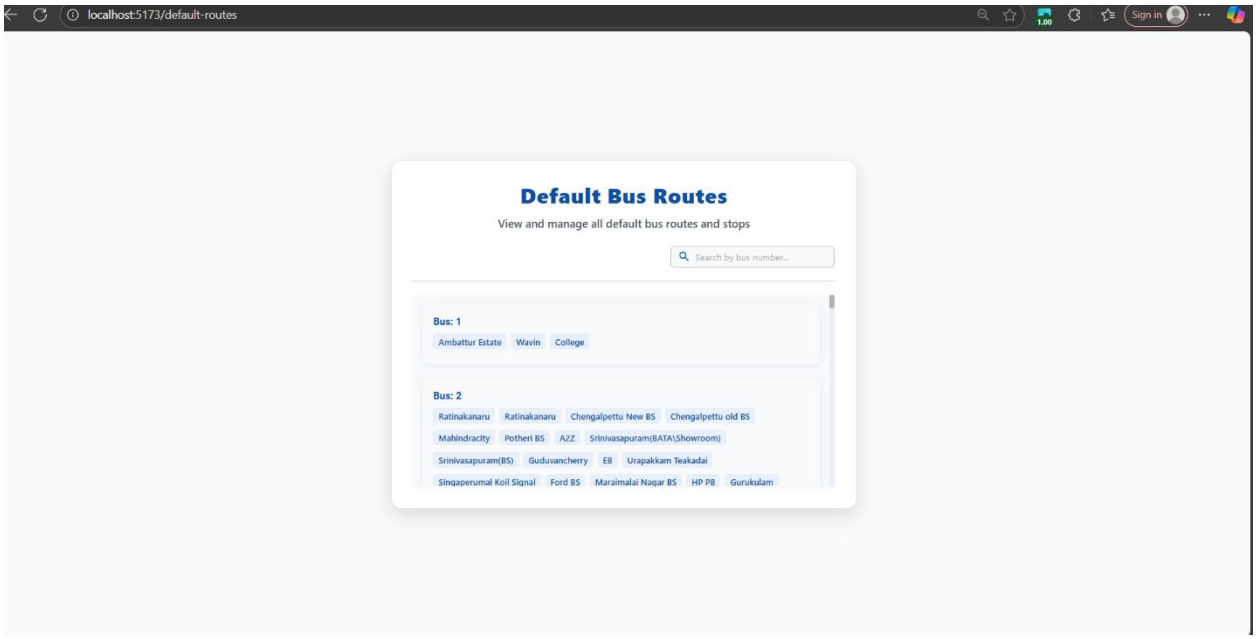
ii. Edit Bus Routes



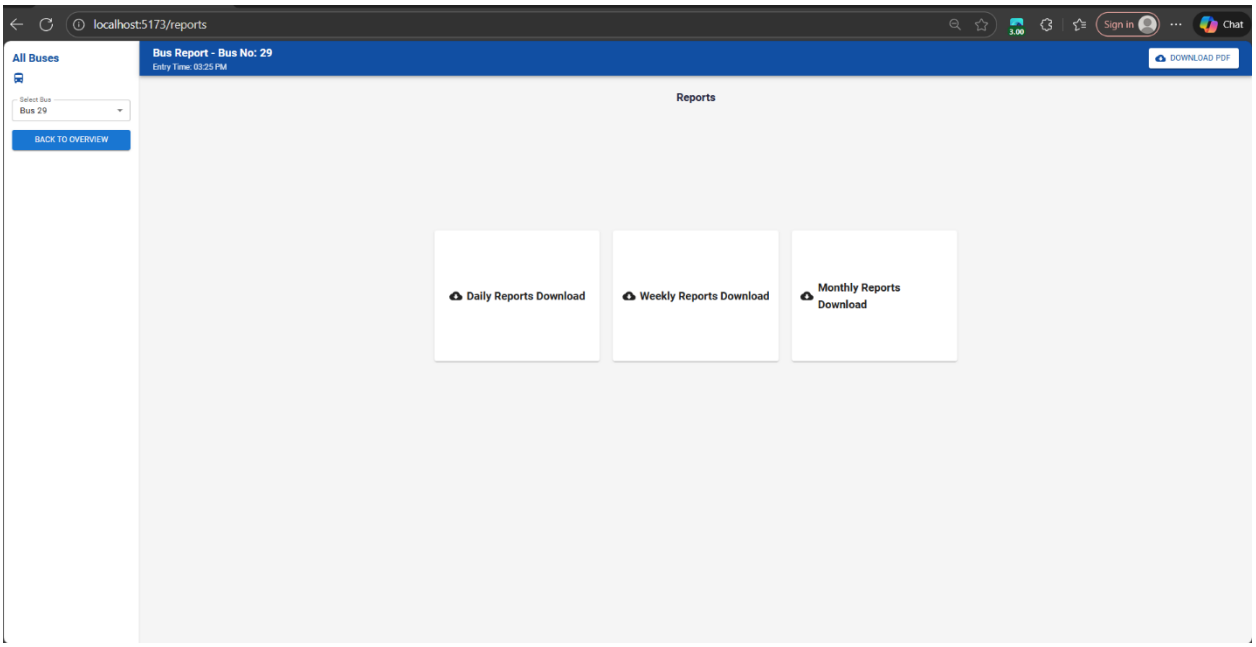
g. Bus Monitor Page

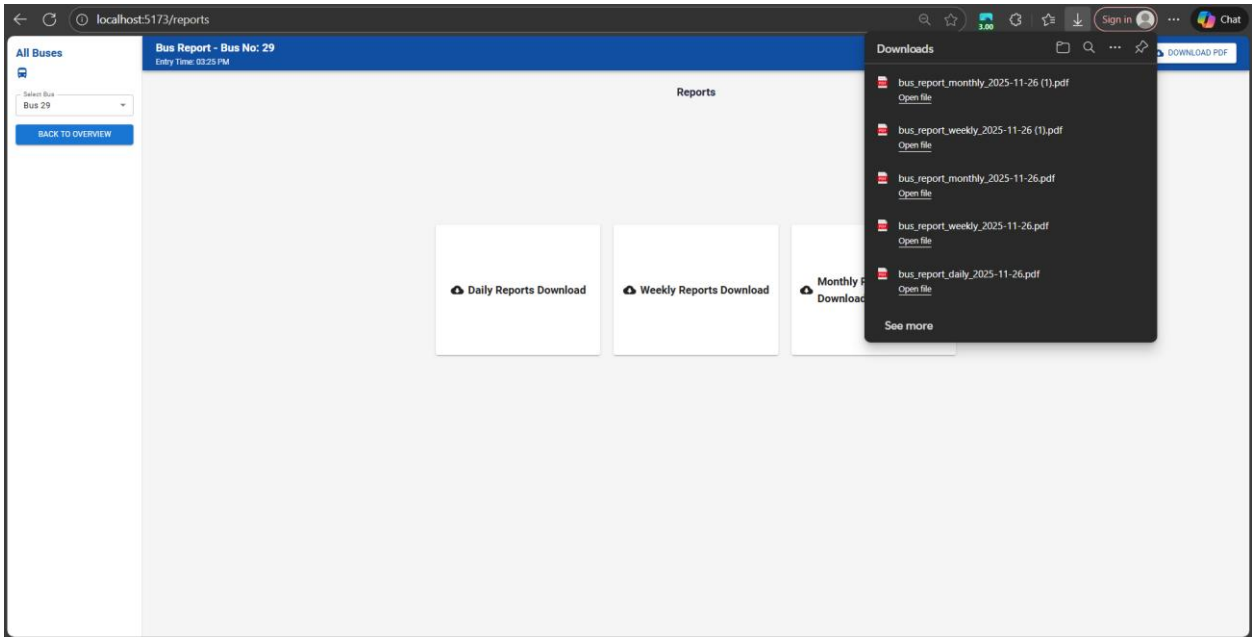


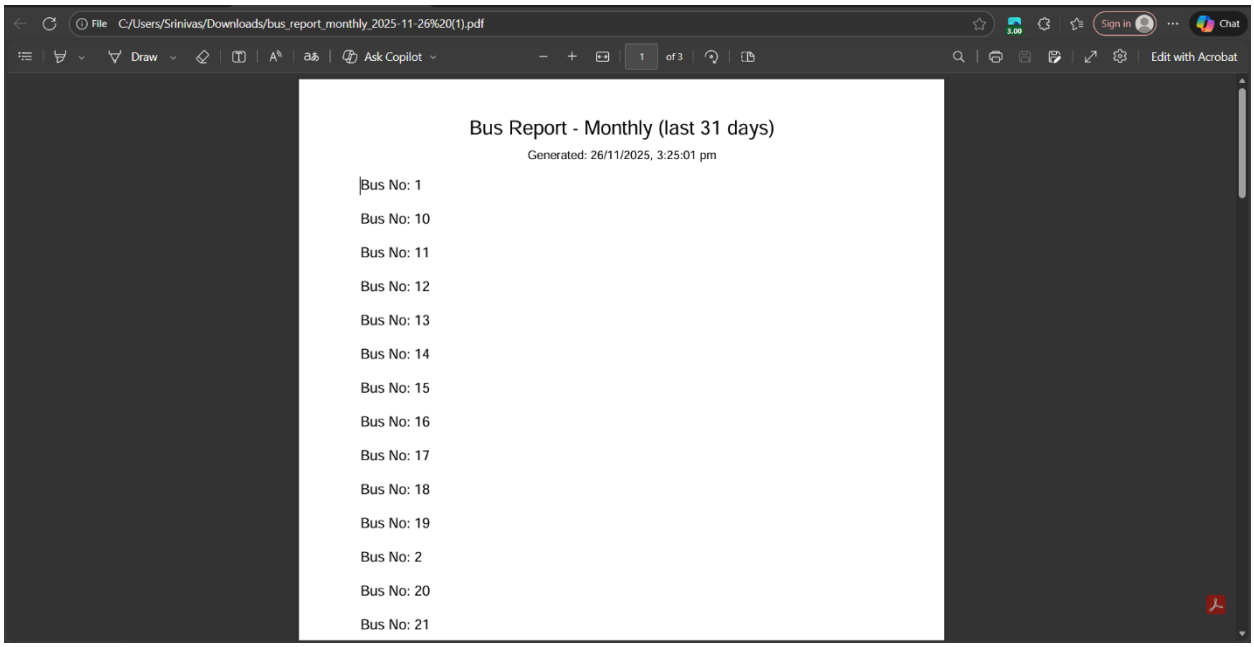
h. Default Routes

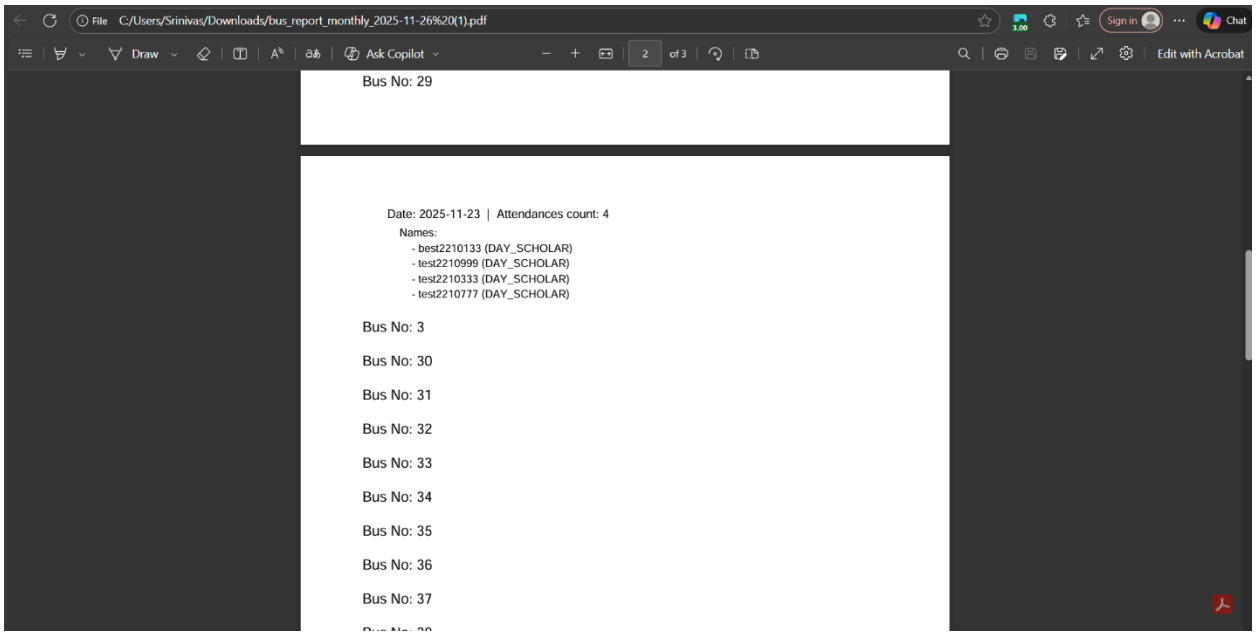


i. Reports Page

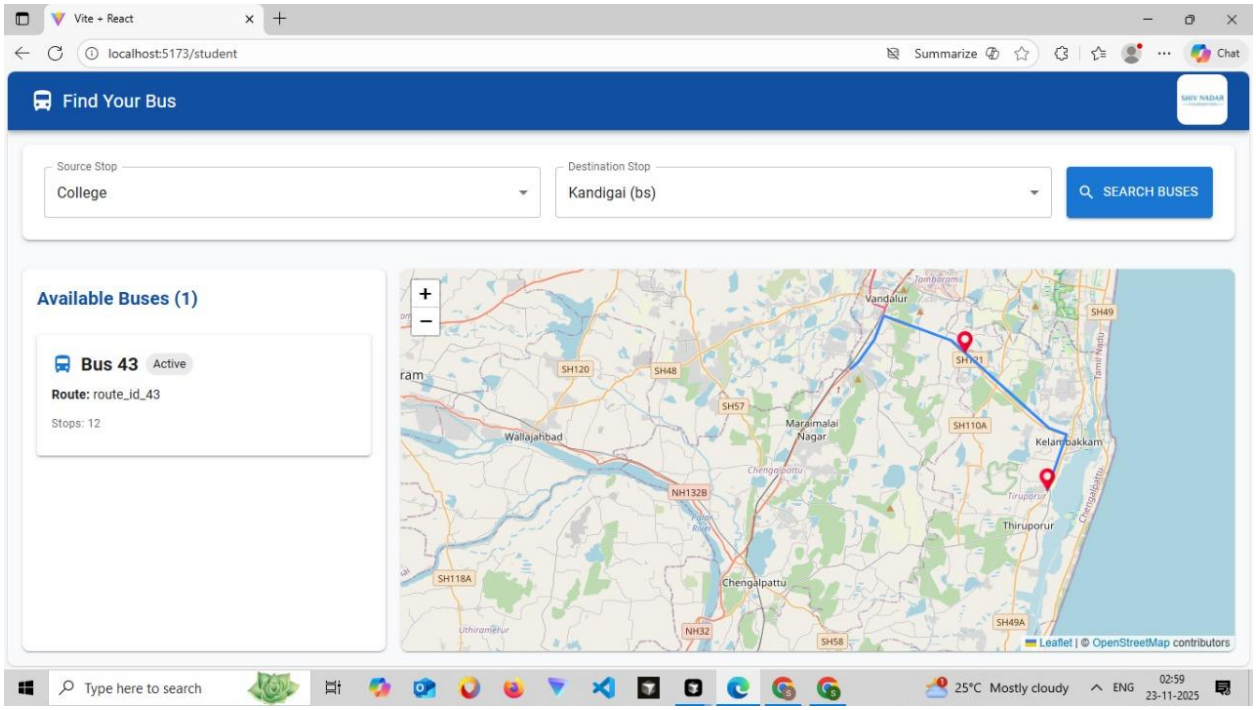



localhost:5173/reports
All Buses
Select Bus
Bus 29
BACK TO OVERVIEW
Bus Report - Bus No: 29
Entry Time: 03:25 PM
DOWNLOAD PDF
Reports
Daily Reports Download
Weekly Reports Download
Downloads
bus_report_monthly_2025-11-26 (1).pdf
Open file
bus_report_weekly_2025-11-26 (1).pdf
Open file
bus_report_monthly_2025-11-26.pdf
Open file
bus_report_weekly_2025-11-26.pdf
Open file
bus_report_daily_2025-11-26.pdf
Open file
See more


File C:/Users/Srinivas/Downloads/bus_report_monthly_2025-11-26%20(1).pdf
Draw
Ask Copilot
1 of 3
Edit with Acrobat
Bus Report - Monthly (last 31 days)
Generated: 26/11/2025, 3:25:01 pm
Bus No: 1
Bus No: 10
Bus No: 11
Bus No: 12
Bus No: 13
Bus No: 14
Bus No: 15
Bus No: 16
Bus No: 17
Bus No: 18
Bus No: 19
Bus No: 2
Bus No: 20
Bus No: 21


File C:/Users/Srinivas/Downloads/bus_report_monthly_2025-11-26%20(1).pdf
Draw
Ask Copilot
2 of 3
Sign in
Chat
Edit with Acrobat
Bus No: 29
Date: 2025-11-23 | Attendances count: 4
Names:
- best2210133 (DAY_SCHOLAR)
- test2210999 (DAY_SCHOLAR)
- test2210333 (DAY_SCHOLAR)
- test2210777 (DAY_SCHOLAR)
Bus No: 3
Bus No: 30
Bus No: 31
Bus No: 32
Bus No: 33
Bus No: 34
Bus No: 35
Bus No: 36
Bus No: 37

2) Student Interface



2.2 Functional Requirements and Use Cases

Admin Use Cases

1. Student Account Setup & Authentication
    - Admin verifies student details and sends system-generated login credentials to their registered email.
2. Student Registration with Facial Data
    - Admin collects face data and student information using a dedicated registration microservice.
    - The data is securely stored in Firebase for attendance processing.
3. Live Bus Tracking
    - Admin can monitor the real-time location of each bus using GPS coordinates streamed from Raspberry Pi devices.
4. Route Management
    - Admin can add, edit, delete, and review the default and updated routes of all buses, including stop-wise details.

5. Bus Monitoring (Raspberry Pi Integration)
    - Each bus is equipped with a Raspberry Pi + camera module that streams attendance and camera feed via ngrok tunneling.
    - Admin can view the live feed and attendance logs from the dashboard.
6. Attendance Reports
    - Admin can download individual student attendance reports or consolidated bus-wise reports for audits and analysis.

Student Use Cases

1. Bus Discovery Based on Stops
    - Students select their source and destination stops.
    - The system displays all buses available for the chosen route with timings and route details.
2. Live Bus Tracking
    - Students can track buses in real time on the map, ensuring accurate trip planning.

Technical Details

3.1 Tech Stack

Frontend

- React.js for the admin dashboard and student portal.

Backend

- Node.js + Express.js for core backend services (authentication, bus management, route management, tracking).
- Flask microservice for student registration and facial data capture.

Database & Authentication

- Firebase Firestore for structured data storage.
- Firebase Authentication for secure access control for admin and students.

IoT & ML Components

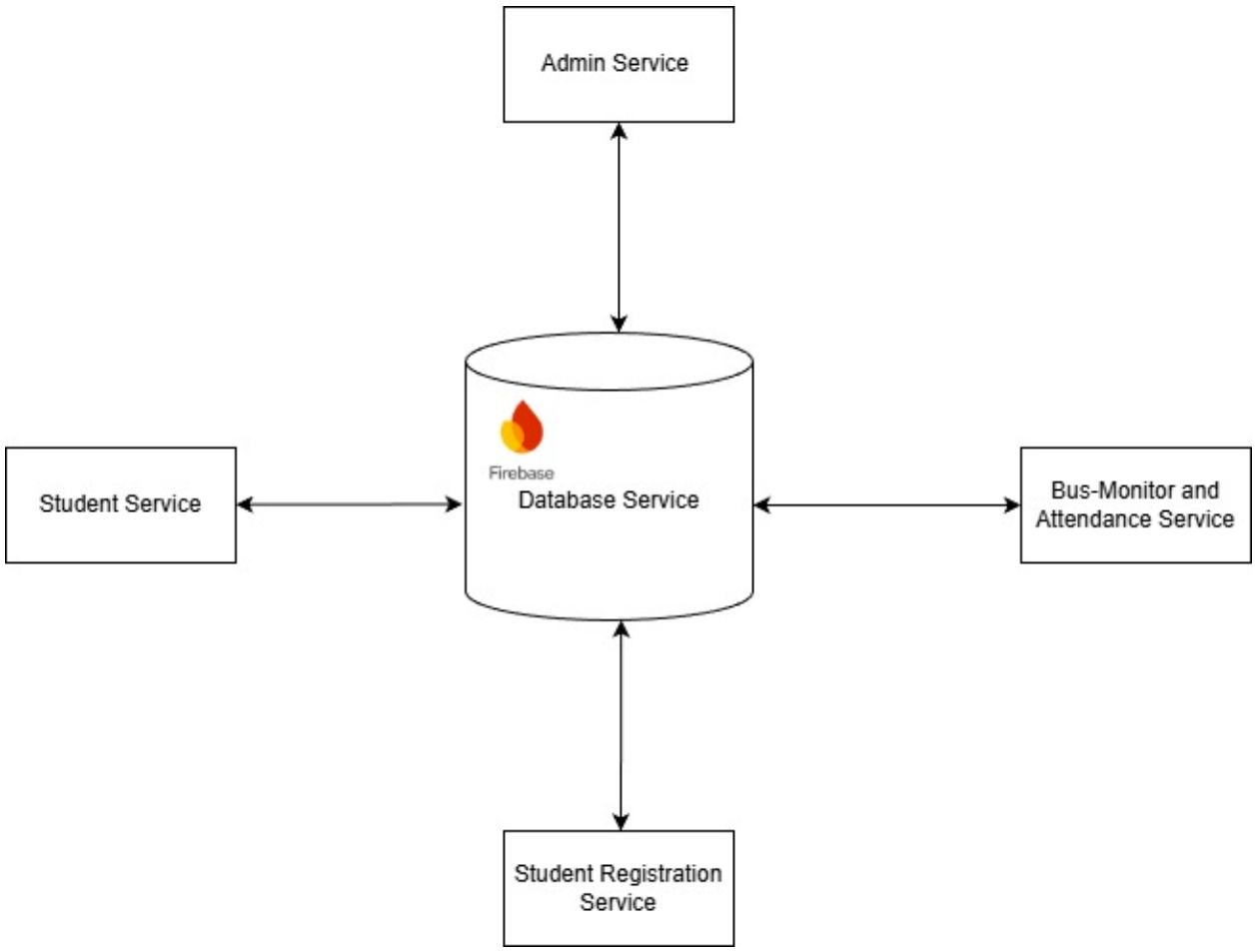
- Raspberry Pi + Camera Module for attendance capture using facial recognition.
- ngrok for secure tunneling and streaming camera outputs to the cloud.

Deployment & DevOps

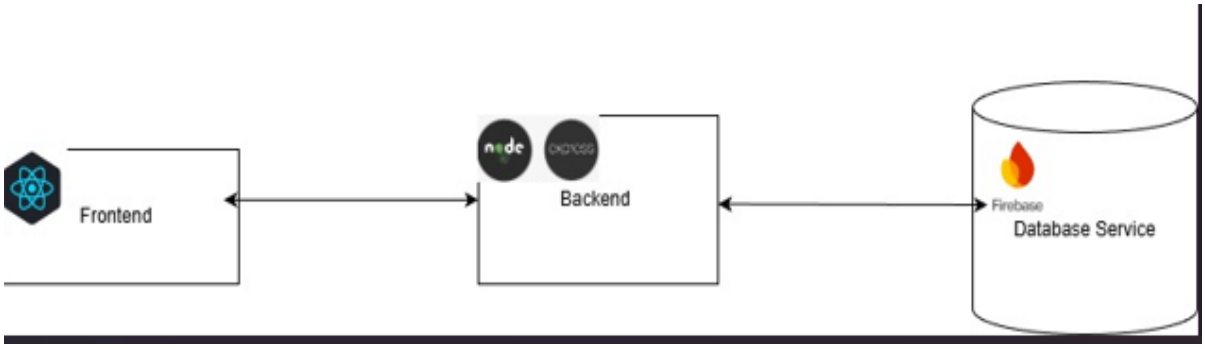
- Render for deploying the registration microservice.
- Git for version control and project collaboration.

3.2 Architecture Overview

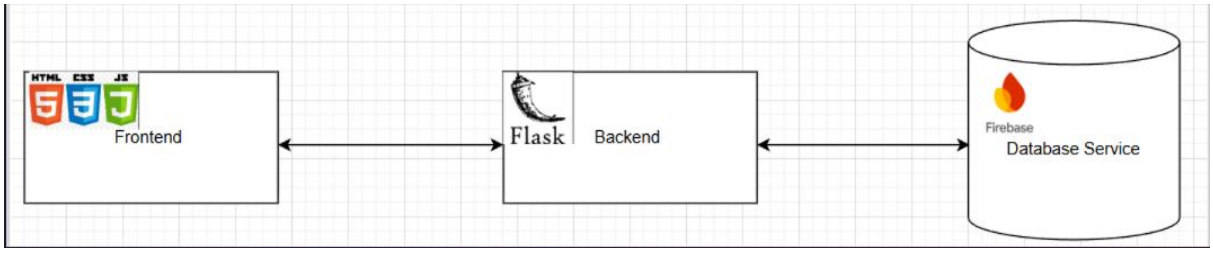
The system follows a modular, microservice-assisted architecture. The main backend (Node.js) handles authentication, bus route configuration, and real-time tracking. A separate Flask microservice manages face data registration. Raspberry Pi devices installed in buses capture facial images, recognize student entry, and push attendance + GPS details to Firebase via ngrok tunnels. The React frontends (admin and student portals) fetch data from Firebase and backend APIs to render dashboards, maps, and attendance information.



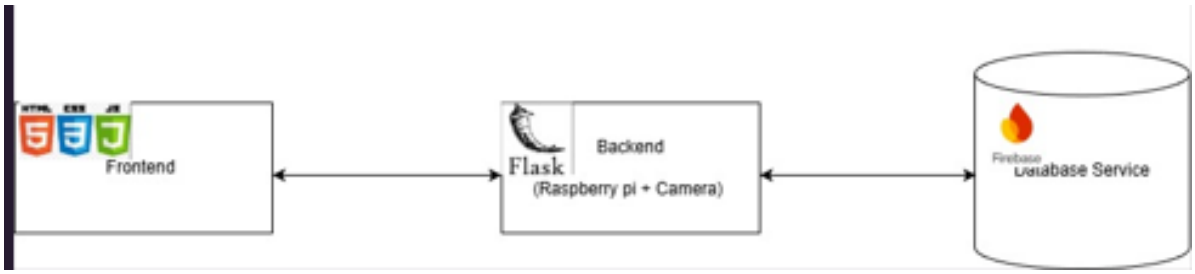
(i) Student and Admin Services



(ii) Student Registration Service



(iii) Bus-Monitor and Attendance Service



3.3 Design Patterns Used

1. Microservices Pattern
    - The facial data registration and attendance capture module is built as an independent Flask microservice, allowing modular updates without affecting core backend logic.
2. MVC Pattern (Model–View–Controller)
    - Applied in the Node.js backend for organizing routes, controllers, and data models cleanly.
3. Observer Pattern
    - Real-time updates are pushed to Firebase, and all connected clients automatically receive updated values.
4. Client–Server Pattern
    - Both student and admin interfaces operate as clients that interact with REST APIs and Firebase for dynamic data flow.