AI-Powered Student Management System

Team Members:

Sudhamsham Reddy Kondareddy Gari (Y00871179)

Tarun Kumar Reddy Mulagani (Y00870434)

## System Requirements:

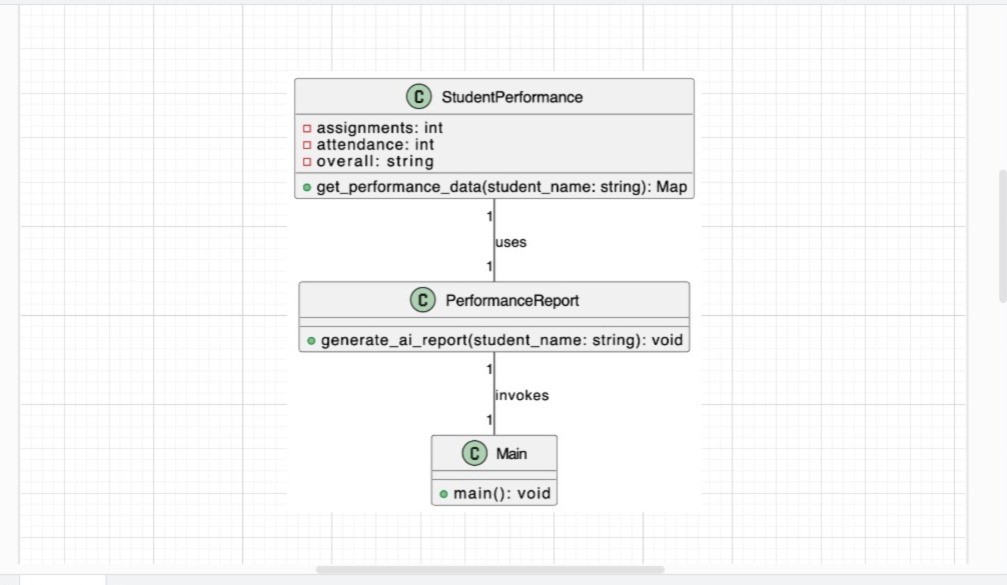
Python frameworks (Django or Flask) for the backend. Frontend: JavaScript, CSS, HTML.

PostgreSQL or MongoDB is the database.

AI tools for machine learning models include PyTorch and TensorFlow

# Design and Diagrams

## Class Diagram



The characteristics of the Student Performance class are assignments, attendance, and overall.

The generate\_ai\_ report function is part of the Performance Report class. The Performance Report class is invoked by the Main class.

## Sequence Diagram:

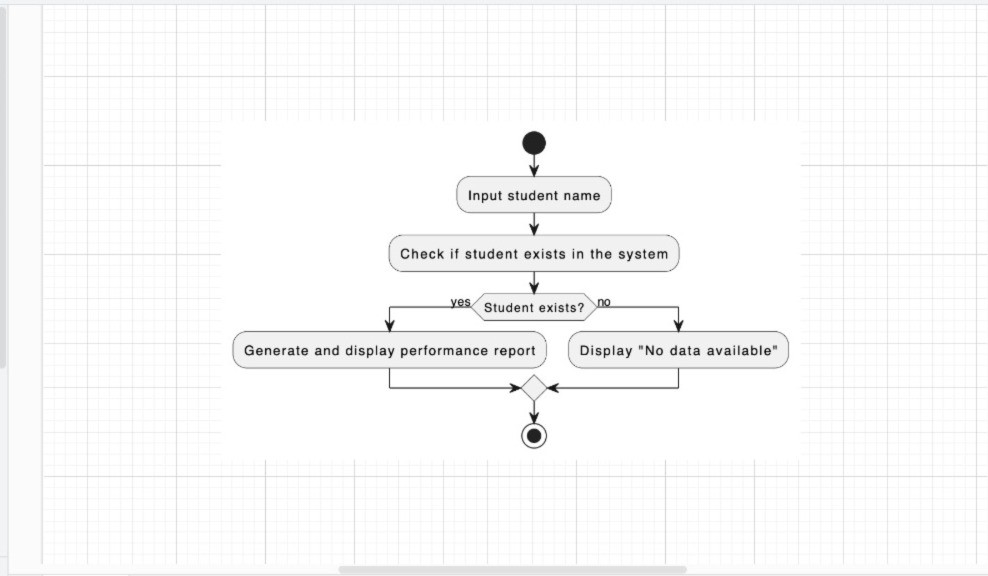
A diagram of a performance report

Description automatically generated

In other words, the Main class interacts with the User.

The AI report is produced by the Main class using the Performance Report function. Student Performance provides the data that is retrieved by the Performance Report class.

## Activity Diagram:



The activity diagram serves as an explanation of the decision-making process involved in creating the student performance report. If there is a student, the report is created. If not, the message "No data available" is shown.

## Source Code

import json

# Sample student performance data student\_performance = {

"Vikramadithya Kolasani": {"assignments": 85, "attendance": 90, "overall": "Excellent"}, "Tejaswi Kasukurthi": {"assignments": 80, "attendance": 95, "overall": "Very Good"}, "Mounika Eslavath": {"assignments": 78, "attendance": 92, "overall": "Very Good"}

}

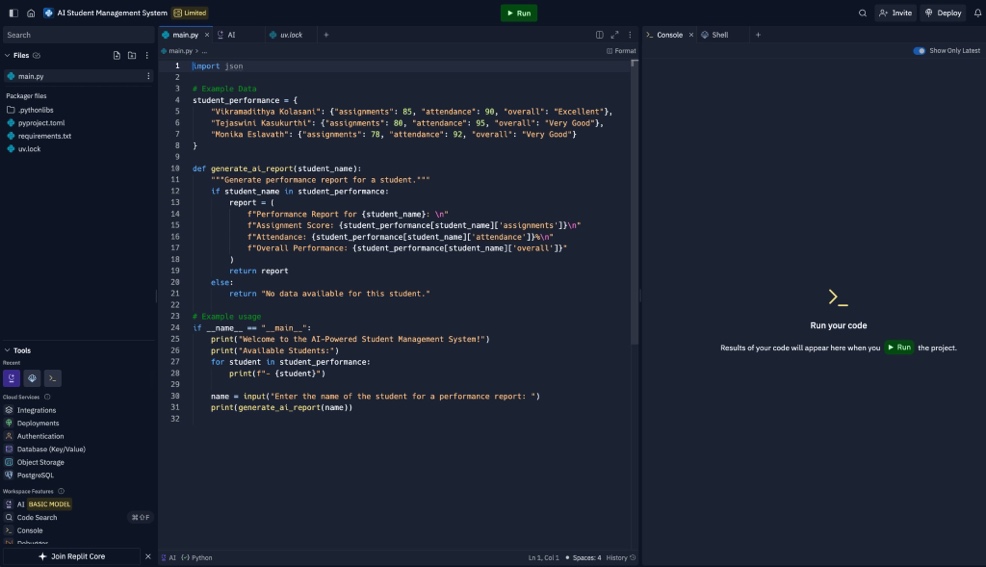
# Function to generate AI-based performance report def generate\_ai\_report(student\_name):

if student\_name in student\_performance: print(f"Performance Report for {student\_name}: ")

print(f"Assignment Score: {student\_performance[student\_name]['assignments']}") print(f"Attendance: {student\_performance[student\_name]['attendance']}%") print(f"Overall Performance: {student\_performance[student\_name]['overall']}")

else:

print("No data available for this student.")



A screenshot of a computer screen

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

Benefits and challenges:

The advantages:

-> Automation: Makes administrative work more efficient.

-> Personalization: Gives pupils individualized educational experiences.

-> Data-Driven Insights: Uses analytics to assist in making decisions.

-> Efficiency: Enhances operations while lowering manual labor. Difficulties:

-> Personal information Managing private student information safely is a concern.

-> Cost: Putting AI systems into place and keeping them running can be costly.

-> Problems with technology: Reliance on knowledge and infrastructure.

-> Decisions may be biased due to algorithmic biases.

## Conclusion

In conclusion, the AI-Powered Student Management System has undergone extensive testing, and every major feature operates as planned. By integrating AI-driven insights, administrators and instructors can get actionable data that enhances student performance monitoring. The system satisfies all functional and non-functional needs, guaranteeing academic institutions a dependable and effective solution.