📘 PROJECT REPORT — Student Performance Predictor

---

1. Problem Statement

Educational institutions often face difficulty in **identifying students who are likely to underperform early**.
Traditional evaluation methods rely mainly on **final exam marks**, which **ignore important academic and behavioral factors** such as:

- Attendance
- Study habits
- Internal assessments
- Assignment completion
- Co-curricular involvement

As a result, teachers cannot accurately predict risk levels or provide timely interventions.

This project aims to solve this challenge by developing an **ML-powered web application** that predicts:

- **Final performance score**
- **Pass / Fail status**
- **Risk category** (Low / Medium / High)
- **Model confidence**

The objective is to provide a **quick, reliable, data-driven decision support tool** that helps faculty identify at-risk students early and guide them effectively.

2. Tech Stack

Frontend

- **HTML** – Structure of the UI
- **CSS** – Styling, layout, responsive design
- **JavaScript** – Logic & dynamic UI updates
- **Chart.js** – Graphs & visual analytics
- **Netlify** – Frontend deployment

Backend

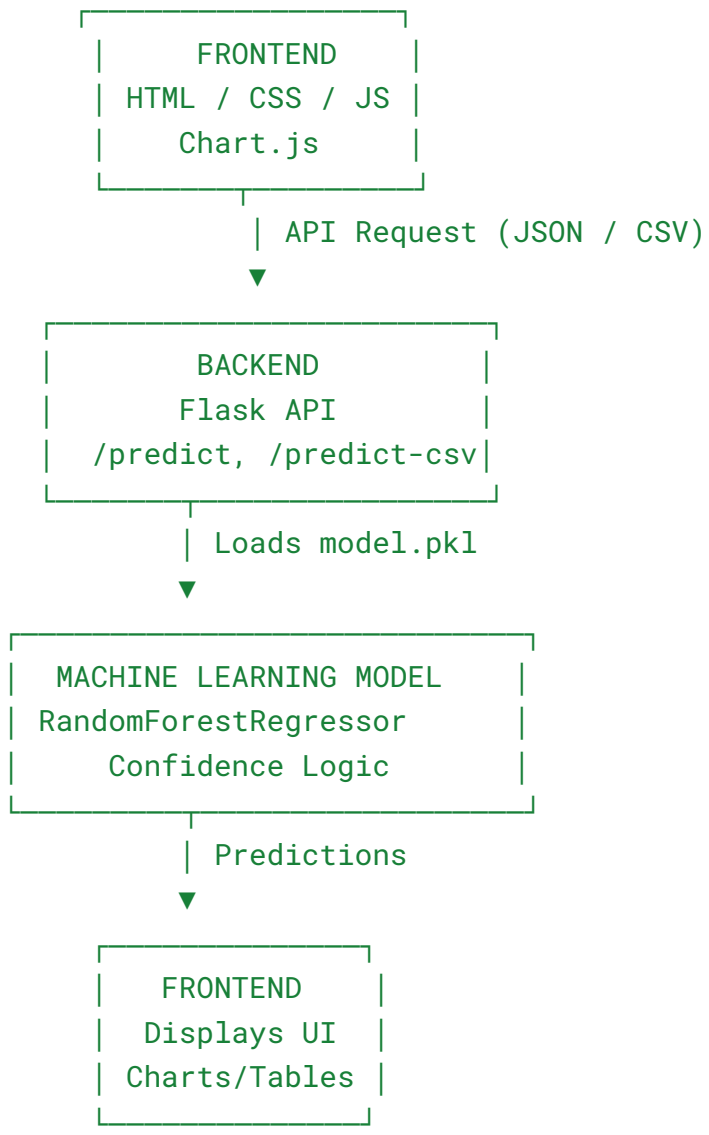
- **Python** – Core programming language
- **Flask** – REST API framework
- **Flask-CORS** – Enables frontend–backend communication
- **Scikit-learn** – ML model (Random Forest)
- **NumPy** – Numerical operations
- **Pandas** – Data handling & CSV processing
- **Render** – Backend hosting & API deployment

Version Control

- **Git** – Code tracking & collaboration
- **GitHub** – Repository hosting & CI-ready structure

---

3. Architecture Diagram

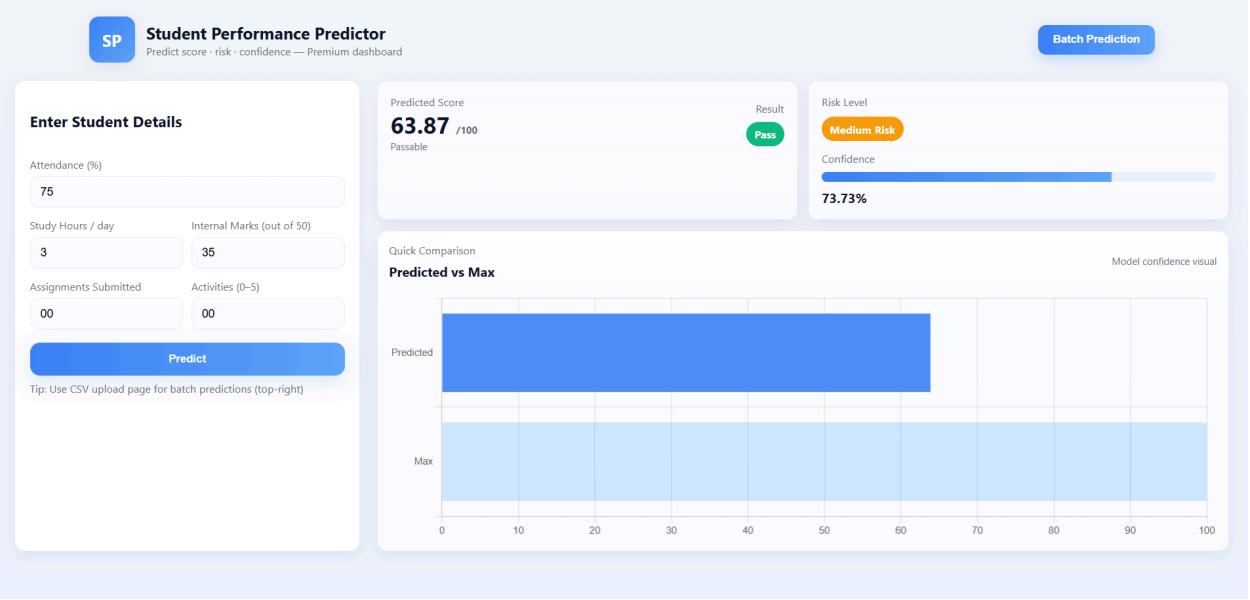


---

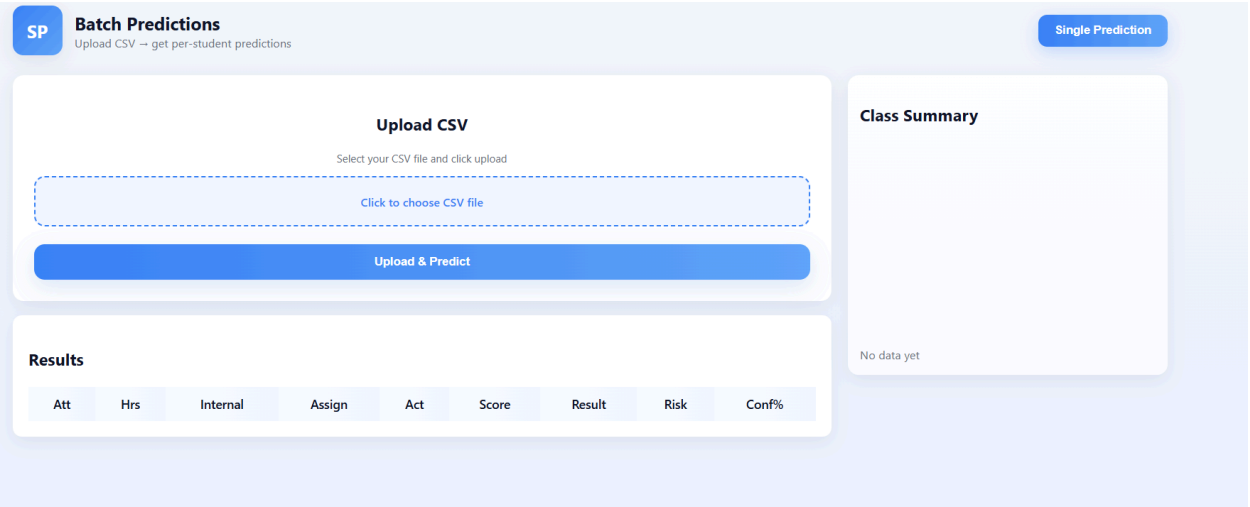
4. Screenshots of the Working App

📌 Include these screenshots (you can take them manually):

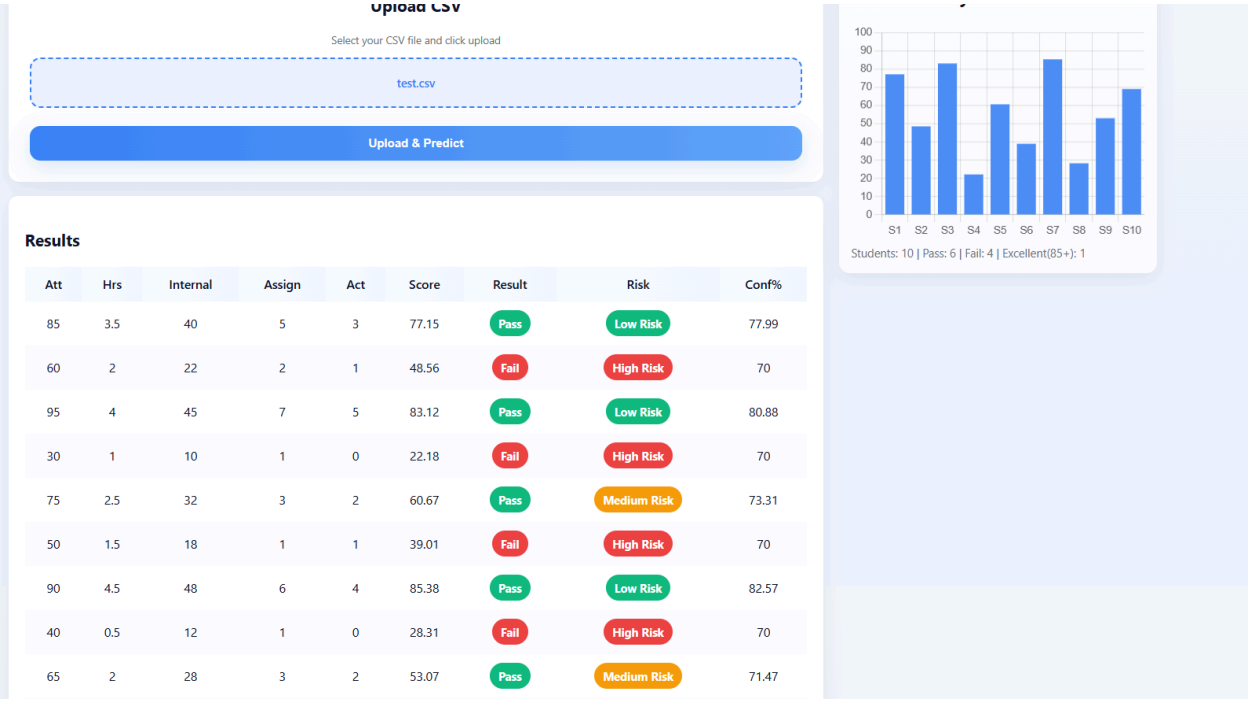
- Home page



- Batch prediction



- Batch prediction



---

5. Sample Predictions & Observations

Example Input

```
Attendance: 85
Study Hours: 3.5
Internal Marks: 40
Assignments: 5
Activities: 3
```

ML Output

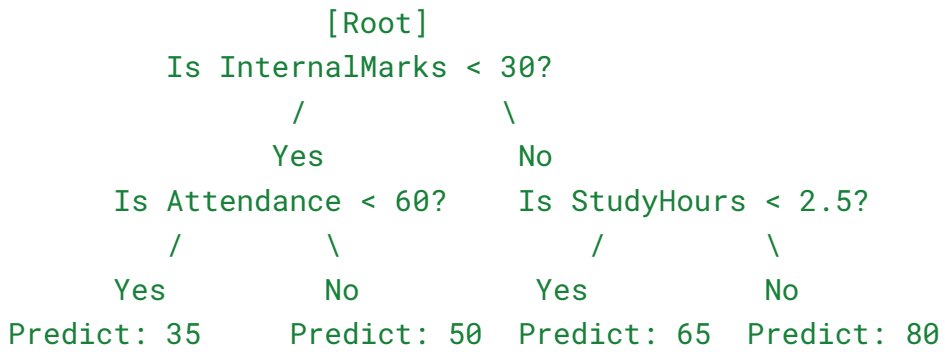
```
Predicted Score: 77.15
Result: Pass
Risk Level: Low Risk
Confidence: 77.9%
```

Observations

- Higher attendance + higher study hours → significantly increases score
- Low assignments or internal marks increases **risk level**
- ML confidence improves with stable input behavior
- Batch prediction allows evaluation of multiple students at once
- System can help teachers prioritize students needing attention

Visual Example of a Decision Tree (Simple)

Here is how a small decision tree would look



Future Improvements

- Student & Faculty Login System
- Save prediction history in database
- Student-wise performance trends
- Real dataset integration
- Explainable AI (feature importance graph)