🎓 Student Dropout Risk Clustering App (Unsupervised)

📌 Goal

Allow educators to upload student performance CSV files and discover clusters of students (Low/Medium/High risk) using unsupervised learning.

✅ Features

- 📥 Upload 1 or more CSVs (e.g., performance + demographics)
- 🧹 Auto data preprocessing and feature engineering
- 📊 Unsupervised clustering using KMeans or DBSCAN
- 📉 Dimensionality reduction with PCA or t-SNE
- 📈 Visualize clusters and risk distribution
- 🧠 Interactive SQL Playground (DuckDB-based)
- 📤 Download filtered high-risk reports
- 🚀 Deploy on Streamlit + Docker + GitHub Actions

📂 Input Columns

Column	Description
student_id	Unique student identifier
quiz1/quiz2/quiz3	Quiz marks
assignment1/2/3	Assignment marks
midterm, final	Exam scores
total_lectures, present_lectures	Attendance tracking
lab_sessions, attended_labs	Lab participation
previous_gpa	Academic history
age, gender	Optional metadata
support_program, internet_access	Social factors

```
📂 student_risk_cluster_app/ (Project Root)
student_risk_cluster_app/
│
├── 📂 data/
│   └── sample_input.csv              # Sample user-uploadable dataset
│
```

```
├── 🗁 notebooks/
│   └── clustering_experiments.ipynb  # Jupyter notebook for model development
│
├── 🗁 app/
│   ├── __init__.py
│   ├── clustering.py                 # KMeans/DBSCAN logic + PCA/t-SNE
│   ├── preprocessing.py              # Cleaning, feature engineering
│   ├── sql_playground.py            # SQL querying logic using DuckDB
│   ├── utils.py                      # Helper functions (e.g. feature calc)
│   └── config.py                     # Feature config & settings
│
├── 🗁 streamlit_app/
│   ├── dashboard.py                  # Main Streamlit interface
│   ├── components/
│   │   ├── file_uploader.py
│   │   ├── risk_summary.py
│   │   ├── visualizations.py
│   │   └── sql_playground_ui.py
│
├── 🗁 docker/
│   └── Dockerfile
│
├── 🗁 .github/
│   └── workflows/
│       └── deploy.yml                # GitHub Actions CI/CD
│
├── requirements.txt
├── .gitignore
├── README.md
└── PROJECT_PLAN.md                  ✅ This file (below)
```

📊 Clustering Strategy

- Engineered Features:
  - `attendance_ratio`, `assignment_avg`, `quiz_avg`, `lab_participation`
- Dimensionality Reduction:
  - `PCA`, `t-SNE`
- Clustering Models:
  - `KMeans`, `DBSCAN`, `IsolationForest`
- Output:
  - Cluster labels → Interpreted into "Low", "Medium", "High Risk"

🖥 Dashboard Features (Streamlit)

- 📥 Upload CSVs
- 📊 Cluster summary cards (Low/Med/High counts)
- 📉 Scatter plots, bar charts by cluster
- 📋 Risk-labeled student table

- 🧪 SQL Playground
  - Query across multiple uploaded files
  - Join tables and run advanced filters
- 📤 Export reports per cluster

⚙️ Tech Stack

Layer	Tools
ML & Preprocessing	pandas, scikit-learn, DuckDB
Clustering	KMeans, DBSCAN, PCA
Visualization	Streamlit
Backend/API (optional)	FastAPI
Deployment	Docker, GitHub Actions

🛠️ How It Works

1. **User uploads 1+ CSVs**
2. **App cleans and processes data**
3. **Model clusters students** into groups
4. **Streamlit shows insights** + SQL Playground
5. **Teacher interprets risk levels**, downloads student reports

---

📦 Deployment

- 📦 Dockerized for local/cloud use
- 🌐 Streamlit share or Render for free deployment
- 🔁 GitHub Actions to auto-rebuild on push

🔮 Future Enhancements

- 📎 LMS API integration (for auto data pull)
- 🧠 Add NLP: feedback sentiment → features
- 🧪 Add active learning: manually label a few, then classify
- 🔔 Alert system (email/WhatsApp for high-risk students)
- 🔍 Live retraining pipeline using Airflow or Prefect

🚀 Outcome

A complete **industry-level ML project** solving a real-world education problem with:

- Unsupervised learning ✅
- CI/CD, Docker ✅

- SQL integration ✅
- Clean UX ✅