STUDENT ACADEMIC PERFORMANCE PREDICTOR

📘 Chapter 1: Project Description and Outline

- **Project Name**: Student Academic Performance Predictor
- **Objective**: To develop a predictive system that estimates students' performance indices based on key factors such as hours studied, sleep hours, previous scores, extracurricular activity involvement, and practice frequency.
- **Motivation**: Educational institutions can use this tool to support students proactively by identifying those who may be underperforming.
- **Scope**:
    - Predict performance using linear regression.
    - Scale and preprocess input data for consistency.
    - Build a user-friendly input and prediction pipeline.

📗 Chapter 2: Related Work Investigation

- **Research Review**:
    - Similar systems have been developed using Decision Trees, SVM, and Neural Networks.
    - This project focuses on explainability through Linear Regression.
- **Gaps Identified**:
    - Many models don’t include user input scalability or interactive interfaces.
    - Often miss out on analyzing factors like sleep and extracurricular activities.
- **Reference**:
    - Cited research from IEEE papers and educational datasets to support the model's feature selection.

📘 Chapter 3: Requirement Artifacts

- **Hardware Requirements**:
    - System with at least 4GB RAM and Python 3.8+.
- **Software Requirements**:
    - Python, Scikit-learn, Pandas, Matplotlib, Joblib.
- **Functional Requirements**:
    - Model must load trained data.
    - Accept multiple inputs at once.
    - Normalize and predict performance index.
- **Non-Functional Requirements**:
    - Accuracy should exceed 95% on test data.
    - Response time < 2 seconds per prediction.

📙 Chapter 4: Design Methodology and Its Novelty

- **Pipeline Overview**:
    - Load data
    - Handle missing/duplicate values
    - Encode categorical features
    - Scale numerical data
    - Train Linear Regression model
    - Save model and scaler
    - Accept new user input and preprocess it

    - Predict and display results

- **Novelty**:
    - Real-time, multi-student input capability.
    - Blended approach of feature engineering with usability.
    - Custom scaler handling for unseen inputs.

📕 Chapter 5: Technical Implementation and Analysis

- **Model Used**: `LinearRegression()` from `sklearn.linear_model`
- **Dataset**: Custom dataset with:
    - 5 Features: Hours Studied, Previous Scores, Sleep Hours, Extracurricular Activities, Papers Practiced
    - 1 Target: Performance Index
- **Preprocessing**:
    - Handled Yes/No as binary values
    - Scaled numerical data using `StandardScaler`
- **Evaluation Metrics**:
    - MAE: 1.61
    - MSE: 4.08
    - RMSE: 2.02
    - R^2 Score: 0.99 (Excellent fit)

📒 Chapter 6: Project Outcome and Applicability

- **Outcome**:

    - Achieved high predictive accuracy.
    - Successful multi-student prediction from user inputs.
- **Use Cases**:
    - Educational counselors can prioritize support.
    - Schools and colleges can implement for semester assessments.
- **Scalability**:
    - Easily extendable to larger datasets and ML models (like Random Forest or XGBoost).

📓 Chapter 7: Conclusion and Recommendations

- **Conclusion**:
    - The project met its goals and demonstrated practical, high-accuracy predictions.
    - Usable in educational environments to monitor academic health.
- **Recommendations**:
    - Extend model to handle text inputs using NLP.
    - Add GUI/website for more accessibility.
    - Store predictions in a database for analysis.
