Project Proposal

Title: Data-Driven Course Demand Prediction & Resource Planning System
University: Vijaybhoomi University
Name: Laxmi Mahesh Parab
Registration Number: 2024SEPVUGP0028

---

1. Abstract

Efficient academic planning is a critical challenge for universities due to the dynamic nature of student course preferences and enrollment patterns. Incorrect estimation of course demand often leads to overcrowded classrooms, underutilized faculty, scheduling conflicts, and inefficient use of institutional resources.

This project proposes a Data-Driven Course Demand Prediction and Resource Planning System for Vijaybhoomi University. The system will use synthetic but realistic university data to train traditional machine learning models capable of predicting expected enrollment for different courses. The predicted demand will assist academic administrators in planning faculty allocation, classroom capacity, and course offerings.

The project focuses not only on model development but also on building a production-style pipeline including data generation, SQL-based storage, model training scripts, prediction interfaces, dashboard visualization, and periodic model retraining to simulate real-world deployment.

---

2. Problem Statement

Universities must decide in advance:

- How many sections of a course to offer
- How many faculty members are required
- Which classrooms should be allocated

Currently, such decisions are often based on historical trends or manual estimation. However, student preferences change due to factors such as course popularity, faculty reputation, difficulty level, and industry demand.

Lack of accurate demand forecasting can result in:

- Overcrowded classes
- Course cancellations due to low enrollment
- Inefficient faculty workload distribution
- Poor classroom utilization

Therefore, there is a need for a data-driven system that can predict course demand and support resource planning.

---

3. Data Description

Since institutional data cannot be publicly shared, the project will use a synthetic data generator that simulates realistic university data. The generator script will be included in the repository instead of uploading datasets.

The synthetic dataset will include features such as:

- Course name/category
- Semester
- Previous enrollment count
- Faculty rating
- Course difficulty level
- Student interest trend (simulated)
- Expected enrollment (target variable)

The data will be periodically regenerated to simulate the arrival of new academic data. All generated data will be stored in a SQL database for further processing.

---

4. Proposed Solution

The system will consist of the following components:

4.1 Data Pipeline

- Synthetic data generation using Python
- Storage of data in a SQL database
- Data retrieval scripts for model training

4.2 Model Training

Three traditional machine learning models will be implemented:

- Linear Regression
- Decision Tree Regressor
- Random Forest Regressor

Model performance will be evaluated using:

- R^2 Score
- Mean Absolute Error (MAE)

The best-performing model will be serialized and stored for prediction.

4.3 Prediction System

A prediction module will allow users to input:

- Course details
- Faculty rating
- Previous enrollment
- Course difficulty

The system will output:

- Predicted student enrollment
- Suggested resource planning insights

4.4 Dashboard

A dashboard (using Streamlit or similar framework) will display:

- Data analysis and trends
- Model performance comparison
- Prediction interface for user input

4.5 Model Lifecycle Management

To simulate real-world deployment:

- New data will be generated periodically
- The model will be retrained at defined intervals
- Model versions will be maintained to track performance changes over time

---

5. Expected Output

The final system will provide:

1. Predicted enrollment for selected courses
2. Model performance metrics and comparison
3. Visual analysis of course demand trends
4. User interface for making predictions
5. SQL-based data storage and retrieval
6. Versioned and retrained models over time

This system will help simulate a real-world academic decision-support tool.

---

6. Implementation and Technology Stack

- Programming Language: Python
- Environment: Google Colab / Local Python Scripts
- Database: SQLite / MySQL
- Libraries: Pandas, NumPy, Scikit-learn, Joblib
- Dashboard: Streamlit
- Version Control: Git and GitHub

The project will follow a structured folder organization with Python scripts instead of relying only on notebooks.

---

7. Project Timeline

Week	Tasks
Week 1	Repository setup, project structure, synthetic data generator
Week 2	SQL integration and data storage pipeline
Week 3	Model training and performance evaluation
Week 4	Prediction module and model serialization
Week 5	Dashboard development
Week 6	Model retraining simulation and version tracking
Week 7	Documentation, testing, and final submission

---

8. Version Control and Repository

A public GitHub repository will be maintained throughout the project. The repository will include:

- Clean folder structure
- Regular commits showing development progress
- Data generation scripts instead of datasets
- Documentation and usage instructions

Large datasets and model files will not be uploaded; instead, scripts will be provided to recreate them.

---

9. Conclusion

This project aims to demonstrate how machine learning can be integrated into a production-style pipeline to support academic planning at Vijaybhoomi University. By focusing on data pipelines, model lifecycle management, and deployment readiness, the system aligns with real-world engineering practices beyond basic model development.