**Project Proposal: Deployment of a Campus Water Consumption and Wastage Prediction System**

**Name:** *Mandar Shinde*
**Registration Number:** 2024SEPVUGP0002
**Course:** Hackathon 3 – Development of Pipelines and Maintenance of Models

## 1. Abstract

This project presents the design and deployment of a machine learning–based system to predict water consumption across a university campus. The system focuses on estimating water usage patterns for different campus buildings under varying operational and academic conditions. Synthetic but realistic data is generated to simulate campus water consumption while ensuring reproducibility and avoiding dependence on sensitive infrastructure data. A structured data pipeline backed by a SQL database is implemented to support continuous data growth and model retraining. Traditional regression-based machine learning models are trained and evaluated, with emphasis on deployment readiness and maintainability rather than aggressive performance optimization. A dashboard interface is developed to visualize historical trends, model performance, and to provide an interactive prediction interface. The project includes a clearly defined update and maintenance strategy to support long-term operational use.

## 2. Problem Statement

Universities rely heavily on water resources for daily operations across hostels, academic buildings, and laboratory facilities. Water consumption levels vary significantly depending on occupancy, time of usage, academic schedules, and vacation periods. In many cases, water management decisions are reactive, leading to inefficiencies and avoidable wastage.
The objective of this project is to predict water consumption (in liters) for campus buildings using historical and operational indicators through regression-based machine learning models. The focus of the project is on building a deployable, reproducible, and maintainable prediction pipeline that can assist university administrators in proactive water management and conservation planning, rather than solely maximizing prediction accuracy.

## 3. Data Description

The dataset used in this project is synthetically generated using a Python-based data generation script designed to simulate realistic campus water consumption behavior. The data represents water usage records across different building types and time periods, capturing variations caused by occupancy levels, academic phases, and temporal factors. Synthetic data generation ensures full reproducibility and avoids reliance on static files or confidential infrastructure records.

All generated data is stored in a lightweight SQL database (SQLite), which serves as the single source of truth for the system. The database supports incremental data insertion, enabling the simulation of real-world data growth and facilitating periodic model retraining.

### Input Features

- building_id
- building_type (Hostel / Academic / Laboratory)
- day_of_week
- time_of_day
- occupancy_percentage
- academic_phase (Normal / Examination / Vacation)
- historical_average_water_usage
- water_tank_level_percentage

### Target Variable

- water_consumption_liters

## 4. Model Implementation and Evaluation

In accordance with course guidelines, only traditional machine learning techniques are used, and neural networks are explicitly excluded. The following regression models are implemented and compared:

- Linear Regretion
- Random Forest Regressor

Model training is performed using data retrieved directly from the SQL database. The dataset is split into training and testing subsets, and feature preprocessing is applied where required.

Model performance is evaluated using standard regression metrics, including Mean Absolute Error (MAE) and Root Mean Squared Error (RMSE). These metrics are used to assess predictive stability and generalization capability. Based on comparative evaluation, a suitable model is selected for deployment.

## 5. Prediction Readiness and Dashboard

The trained model is integrated into a prediction pipeline that allows new operational inputs to be processed using the same preprocessing steps applied during training. A dashboard interface is developed using Streamlit to provide the following functionalities:

- Visualization of historical water consumption trends across campus buildings
- Display of model evaluation metrics and comparison results
- An interactive input form for generating real-time water consumption predictions

The dashboard is configured to always load the most recently trained model, ensuring that predictions reflect the latest available data and model version.

## 6. Model Update and Maintenance Timeline

The project incorporates a clear and realistic maintenance strategy to support long-term usability:

- New synthetic water consumption records are periodically appended to the SQL database to simulate continuous data inflow.
- Model retraining is performed after the accumulation of approximately six months of new data.
- Each retraining cycle results in a newly versioned trained model, while previous model versions are preserved for reference and comparison.
- The dashboard automatically uses the latest trained model for all prediction tasks.

## 7. Version Control and Reproducibility

A public Git repository is maintained to ensure transparent version control and reproducibility of the project. The repository follows a clean and modular folder structure separating data generation scripts, database logic, model training and evaluation code, and dashboard implementation. Commit history reflects incremental development and maintenance milestones. Large datasets and trained model binaries are excluded from version control; instead, data generation scripts are included to allow complete reproduction of the system.

## 8. Expected Output

- A SQL-backed, reproducible data pipeline for campus water consumption simulation
- Trained and evaluated regression models for water usage prediction
- A prediction-ready system capable of handling new operational inputs
- An interactive dashboard for analysis and forecasting
- A maintainable machine learning pipeline with a defined retraining strategy