Project Design Phase-II
Technology Stack (Architecture & Stack)

Date	31 January 2026
Team ID	LTVIP2026TMIDS82279
Project Name	Rising Waters: A Machine Learning Approach to Flood Prediction
Maximum Marks	4 Marks

Project Flow

Below is the project flow of flood prediction system.

Example: Flood Prediction System

Install Required Libraries

Installing necessary Python packages to perform data analysis, model training, and web application development.

Libraries Used:

- NumPy
- Pandas
- Matplotlib
- Scikit-learn
- XGBoost
- Flask

2 Data Collection

Collect historical environmental data (rainfall, temperature, humidity, etc.) from reliable sources or create a structured dataset in CSV format.

Tools/Libraries Used:

- Pandas
- Kaggle / Government datasets
- CSV files
- Excel

3️⃣ Data Preprocessing

▪ Import the Libraries

Load required Python libraries to handle data manipulation, visualization, and machine learning tasks.

Libraries: Pandas, NumPy, Matplotlib, Seaborn, Scikit-learn

▪ Importing the Dataset

Read the CSV dataset into a DataFrame for analysis and processing.

Libraries: Pandas

▪ Understanding Data Type & Summary

Analyze dataset structure, data types, and statistical summary to understand feature distribution.

Libraries: Pandas

▪ Handling Missing Data

Detect and treat missing values using imputation or removal techniques.

Libraries: Pandas, Scikit-learn

▪ **Data Visualization**

Create graphs and charts to understand trends, correlations, and patterns in flood-related data.

Libraries: Matplotlib, Seaborn

▪ **Drop Columns & Replace Missing Values**

Remove irrelevant columns and replace missing values with mean/median/mode.

Libraries: Pandas

▪ **Splitting Dependent & Independent Variables**

Separate input features (X) and target variable (y) for model training.

Libraries: Pandas

▪ **Splitting Data into Train & Test**

Divide dataset into training and testing sets to evaluate model performance.

Libraries: Scikit-learn

4️⃣ **Model Building**

▪ **Training and Testing the Model**

Train the XGBoost algorithm on training data and test its predictions on unseen data.

Libraries: XGBoost, Scikit-learn

▪ **Evaluation of Model**

Measure model accuracy using metrics like Accuracy, Precision, Recall, F1-score, and Confusion Matrix.

Libraries: Scikit-learn

▪ **Saving the Model**

Save the trained model into a file for future use in the web application.

Libraries: Joblib / Pickle

5 Application Building

▪ **Create an HTML File**

Design a web form to collect user input such as rainfall, temperature, etc.

Technologies: HTML, CSS

▪ **Build Python Code**

Develop backend logic using Flask to load the model and process user inputs.

Libraries: Flask, Joblib, NumPy

6️⃣ Final UI – Dashboard of Flask App

Create a user-friendly interface to display flood prediction results clearly.

Technologies/Libraries:

- Flask
- HTML
- CSS
- Bootstrap