**Student Name:** Dharshini.S

**Register Number:** 613023104021

**Institution:** Vivekanandha College of Technology For Women

**Department:** BE Computer Science And Engineering

**Date of Submission:** 02.04.2025

**GitHub Repository Link:** https://github.com/dharshini-123-code/project-1.git

### **1. Problem Statement**

* Air pollution impacts public health, climate, and ecosystems. This project focuses on predicting air quality levels based on pollutant and meteorological data. It is primarily a regression problem (predicting AQI), but can also be treated as a classification problem (AQI category).
* **Why it matter:** Accurate AQI prediction enables early warnings, policy interventions, and supports smart city infrastructure.

### **2. Project Objectives**

* Predict AQI or its category using machine learning models.
* Ensure accuracy, interpretability, and real-world usability.
* Post-EDA, included both regression and classification to adapt to AQI categorization.

### **3. Flowchart of the Project Workflow**

* **Data Acquisition and Preparation:** Collect data from various sources (weather stations, air quality monitoring sensors).
* **Data Cleaning and Preprocessing:** Handle missing values, outliers, and inconsistent data formats.
* **Exploratory Data Analysis (EDA):** Analyze the data to understand patterns, relationships, and distributions of variables.
* **Feature Engineering:** Create new features by combining existing ones or using domain knowledge.
* **Model Selection and Training:** Choose appropriate machine learning models (e.g., regression models like linear regression, decision trees, etc.) and train them on the prepared data.
* **Model Evaluation and Refinement:** Evaluate the performance of the models using appropriate metrics (e.g., MAE, RMSE) and refine the models based on the evaluation results.
* **Visualization and Reporting:** Visualize the results using plots and charts and create a report summarizing the findings.
* **Deployment and Monitoring:** Deploy the final model for real-time AQI prediction and monitor its performance.

**4. Data Description**

* **Source:** Kaggle (or other open APIs like CPCB or OpenAQ)
* **Type:** Structured, time-series
* **Size:** ~30,000 rows, 15 features
* **Nature:** Dynamic dataset
* **Target Variable:** AQI (numeric or categorical)

### **5. Data Preprocessing**

* **Missing Values:** Imputed using mean/median/mode
* **Duplicates:** Removed based on timestamps and location
* **Outliers:** Handled using IQR and capping
* **Data Types:** Ensured correct formatting (e.g., datetime, float)
* **Encoding:** One-hot encoding for wind direction and AQI category
* **Scaling:** StandardScaler/MinMaxScaler applied to numerical features

### **6. Exploratory Data Analysis (EDA)**

* **Univariate:** Histograms and boxplots for PM2.5, PM10, temperature
* **Multivariate:** Correlation heatmap, scatter plots
* **Target Analysis:** AQI rises with PM2.5 and PM10; meteorological factors indirectly influence
* **Insights:** PM2.5 and PM10 are key contributors to AQI

### 

### **7. Feature Engineering**

* **Created:** AQI Category (e.g., Good, Moderate, Poor), Time-based features (month, season)
* **Ratios:** PM2.5/PM10
* **Optional:** PCA for dimensionality reduction
* **Justification:** Features selected based on correlation with AQI

### **8. Model Building**

* **Models Used:**
* **Regression:** Random Forest Regressor, XGBoost Regressor
* **Classification:** Decision Tree, Logistic Regression
* **Train-Test Split:** 80/20 (stratified for classification)
* **Metrics:**
* **Regression:** MAE, RMSE, R²
* **Classification:** Accuracy, Precision, Recall, F1-score

### **9. Visualization of Results & Model Insights**

* **Tools:** Matplotlib, Seaborn, Plotly
* **Plots:** Confusion matrix, ROC curve, feature importance, residual plots
* **Key Insights:** PM2.5 and PM10 most influential; tree-based models outperformed linear ones

### **10. Tools and Technologies Used**

* Language: Python
* IDE: Jupyter Notebook, Google Colab
* Libraries: pandas, numpy, scikit-learn, seaborn, matplotlib, XGBoost
* Visualization Tools: Matplotlib, Seaborn, Plotly

### 

### **11. Team Members and Contributions**

**Team head: Geetha Rubaha M**

* **Responsibilities:** Handled data cleaning, missing value treatment, and ensured data quality.

**Kaviya C**

* **Responsibilities:** Performed exploratory data analysis (EDA) and engineered new features based on insights.

**Bhuvaneshwari E**:

* **Responsibilities:** Built and trained machine learning models, evaluated performance metrics.

**Charunethra M:**

* **Responsibilities:** Created visualizations, summarized results, and compiled the final documentation.

**Dharshini S:**

* **Responsibilities:** Managed preprocessing pipeline, tuned hyperparameters, and compared model performance.