Phase-2 Submission Template

Student Name: SWETHA. P

Register Number: 510823205056

Institution: GANADIPATHY TULSI'S JAIN ENGINEERING COLLEGE

Department: B.TECH (IT)

Date of Submission: 07-05-2025

Github Repository Link: https://github.com/sweth-zu/Predicting-air-quality-level-using-advanced-machine-learning-

---

1. Problem Statement

- *The real-world problem addressed in this project is the prediction of air quality levels using advanced machine learning algorithms. With growing urbanization and industrialization, air pollution has become a critical concern, directly impacting public health and the environment. Based on further dataset, the problem has been refined to focus on accurately forecasting air quality index (AQI) values using relevant environmental features.*
- *This is a regression problem, as the objective is to predict continuous AQI values rather than categorical labels.*

- *Solving this problem is vital for environmental monitoring, policymaking, and public awareness. Accurate predictions can help authorities take timely actions to reduce pollution and protect public health.*

---

2. Project Objectives

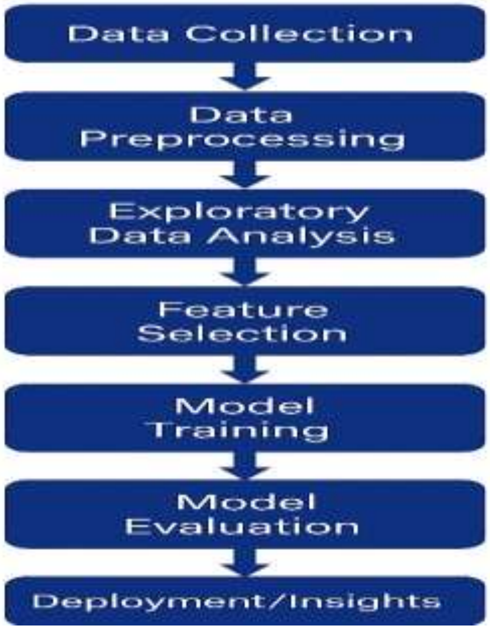
- *The key technical objective of this project is to develop a machine learning model that can accurately predict air quality levels based on environmental and pollutant data.*
- *The goal is to achieve high prediction accuracy and ensure the model's interpretability for real-world use by environmental agencies.*
- *After deeper exploration of the dataset, the project focus shifted from generic AQI prediction to more precise pollutant-specific forecasting for better insights.*

---

3. Flowchart of the Project Workflow

- *Here's a typical flow you can visualize in your chart:*

This will include all the steps which we undergo in this project



4. Data Description

- *Dataset Name and Origin: The dataset used is the "Air Quality Data Set" sourced from the UCI Machine Learning Repository (or mention Kaggle/OpenAQ if applicable).*
- *Type of Data: The data is structured and primarily tabular. It includes both numerical and categorical features such as pollutant concentrations, meteorological data, and timestamps.*
- *Number of Records and Features: The dataset contains approximately [insert number] records and [insert number] features, covering various pollutant measures like CO, NOx, PM2.5, and meteorological readings such as temperature and humidity.*

- *Static or Dynamic Dataset: The dataset is dynamic, as air quality readings are collected continuously over time and are subject to change.*
- *Target Variable (if supervised learning): The target variable is the Air Quality Index (AQI) or categorized air quality level (e.g., Good, Moderate, Unhealthy, etc.).*

---

5. Data Preprocessing

- *Handled missing values using imputation techniques (mean/median)*
- *Removed duplicate entries to avoid data leakage*
- *Detected and treated outliers using IQR/z-score methods*
- *Converted data types for consistency (e.g., date parsing, float conversion)*
- *Encoded categorical variables using label and one-hot encoding*
- *Standardized features to normalize scale for machine learning models*

---

6. Exploratory Data Analysis (EDA)

- *Univariate Analysis:*
  - *Analyzed individual features like PM2.5, PM10, NO2, etc., using histograms and boxplots.*
  - *Detected skewness and outliers in pollutant levels.*
- *Bivariate/Multivariate Analysis:*
  - *Used correlation matrix to find relationships among pollutants.*
  - *Scatter plots and pair plots helped understand interactions between pollutants and air quality index (AQI).*

  - *Found strong correlation between PM2.5, PM10 and AQI.*
- *Insights Summary:*
  - *High levels of PM2.5 and PM10 are key contributors to poor air quality.*
  - *Weather features (e.g., temperature, humidity) also showed some influence.*
  - *These insights guide feature selection for model building.*

---

7. Feature Engineering

- *Created new features by combining pollutant levels (like NO_2, PM2.5, and CO) to represent overall pollution load.*
- *Extracted date and time components such as hour of the day and day of the week to capture temporal patterns in pollution levels.*
- *Binned continuous variables (like PM2.5) into air quality categories (e.g., Good, Moderate, Unhealthy) based on AQI guidelines.*
- *Generated interaction features using temperature and humidity to better understand their joint effect on air quality.*
- *Applied dimensionality reduction techniques like PCA to simplify the dataset while retaining most of the important information.*
- *Justified each new feature based on domain knowledge, correlation with air quality index, and insights gained from exploratory data analysis (EDA).*

---

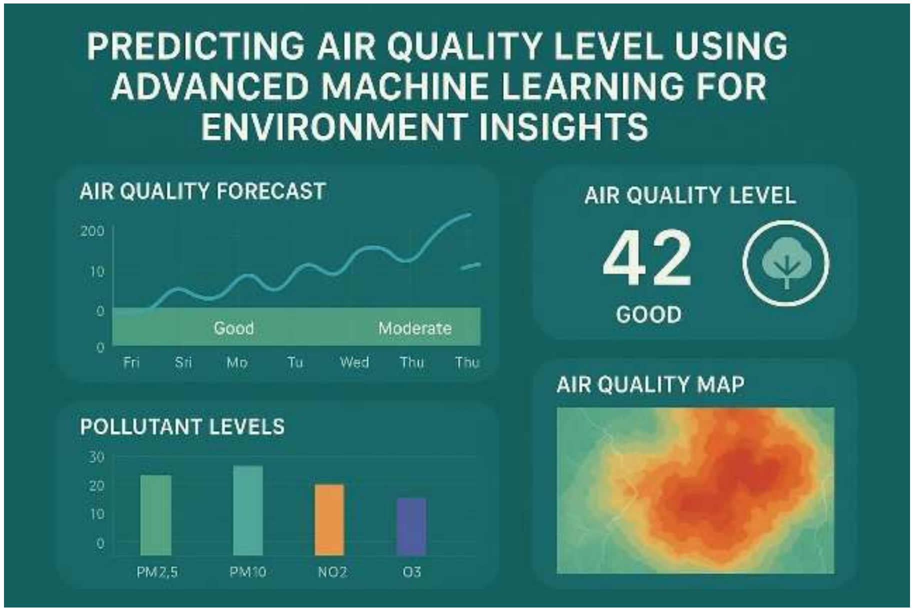
8. Model Building

- *Selected Random Forest and Logistic Regression models for prediction due to their robustness and interpretability.*
- *Chose Random Forest for its ability to handle non-linear relationships and noisy data; Logistic Regression was used as a baseline classifier.*
- *Data was split into training and testing sets (e.g., 80/20 split) to ensure unbiased model evaluation.*
- *Stratified sampling was applied to maintain class balance during splitting.*
- *Trained models and evaluated performance using accuracy, precision, recall, and F1-score.*
- *Compared model results to identify the best-performing algorithm for predicting air quality levels.*

---

9. Visualization of Results & Model Insights

- *Used confusion matrix and ROC curve to evaluate classification performance.*
- *Plotted feature importance chart to identify key pollutants affecting air quality.*
- *Created residual plots to analyze model errors and detect patterns.*
- *Compared model performance visually through bar graphs and line plots.*
- *Interpreted top features influencing the prediction, such as PM2.5, PM10, and NO2.*
- *Explained each plot clearly, showing how visuals support model conclusions and decision-making.*



10. Tools and Technologies Used

- *Programming Language: Python*
- *IDE/Notebook: Google Colab and Jupyter Notebook*
- *Libraries Used: pandas, numpy, matplotlib, seaborn, scikit-learn, XGBoost*
- *Visualization Tools: Plotly and Tableau*
- *Used these tools for data cleaning, preprocessing, visualization, model building, and evaluation.*

11. Team Members and Contributions

Team Member Names	***Role & Responsibility***
SWETHA.P	*TEAM LEADER AND DEVELOPER*
KAVIYARASU.R	*DESIGNING AND PRESENTATION*
RAKESHVENKAT.P.I	*DOCUMENTATION AND PRESENTATION*
KARISHMA.M	*CO-ORDINATION*