

Air Quality Prediction - Project Workflow & Source Code

Flowchart of Project Workflow

Project Workflow Flowchart:

1. Data Collection
2. Data Preprocessing
3. Exploratory Data Analysis (EDA)
4. Feature Engineering
5. Model Building (e.g., XGBoost)
6. Model Evaluation (MAE, RMSE, R2 Score)
7. Deployment (Streamlit-based Web App)

Source Code (Streamlit + XGBoost)

```
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
from sklearn.model_selection import train_test_split
from sklearn.metrics import mean_squared_error, r2_score, mean_absolute_error
from xgboost import XGBRegressor
import streamlit as st

# Load dataset
df = pd.read_csv("air_quality_data.csv")
df.dropna(inplace=True)
df = df.drop_duplicates()

# Feature selection
features = ['PM2.5', 'PM10', 'NO2', 'SO2', 'CO', 'O3', 'TEMP', 'HUMIDITY']
X = df[features]
y = df['AQI']

# Split dataset
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

# Train model
model = XGBRegressor()
model.fit(X_train, y_train)

# Evaluate model
y_pred = model.predict(X_test)
st.write("### Model Evaluation")
st.write(f"MAE: {mean_absolute_error(y_test, y_pred):.2f}")
st.write(f"RMSE: {mean_squared_error(y_test, y_pred, squared=False):.2f}")
```

```
st.write(f"R2 Score: {r2_score(y_test, y_pred):.2f}")

# Streamlit UI
st.title("AQI Predictor")
st.sidebar.header("Input Data")
input_data = {f: st.sidebar.slider(f, float(X[f].min()), float(X[f].max())) for f in features}
input_df = pd.DataFrame([input_data])
if st.button("Predict AQI"):
    prediction = model.predict(input_df)[0]
    st.success(f"Predicted AQI: {prediction:.2f}")
    if prediction <= 50:
        st.info("Air Quality: Good")
    elif prediction <= 100:
        st.warning("Air Quality: Moderate")
    else:
        st.error("Air Quality: Poor")
```