# 1 Design Prototype (modifications can be made):

- Create basic forecasting model for any one city/state/country to forecast any one pollutant NO2/O3/SO2/CO – This will demonstrate the capability of forecasting
- Forecasting model would be wrapped in an API which would act as a user interface which would accept GET/POST request and returns the forecasted values
- Forecasting framework would then be scaled to incorporate multiple city/state/country to forecast all the four pollutants for the final project delivery
- The API would be scaled to incorporate parameters for user to select the desired pollutant and city/state/country for which forecasting is needed

# 2 Task Division Team Members

Each member of the team can be assigned a different Task which would be then integrated together to form the complete working project.

## 2.1 Member #1 Colin Naehr

- Responsible for Exploratory Data Analysis, Data Visualization Descriptive Analysis, Derive Analytical Insights – if possible and Data Preparation for Forecasting
- Integrate modules developed by other team members
- Creating Presentation and focus on data story telling
- Identify test cases/scenario where application will fail so that other members can handle this kind of scenario

## 2.2 Member #2 Yuko Matsumoto (plus planning, etc.)

- Develop Classical/Statistical Forecasting Model Training, Validation, and Inference Script
- Classical/Statistical Model would include
    - ARIMA
    - Hot Winter
    - Exponential Smoothing
    - Help other Team members in developing scripts and solve for error

## 2.3 Member #3 Haotian Wang

- Develop Machine Learning Base Forecasting Model Training, Validation, and Inference Script
- Machine Learning base Model would include
    - LSTM- RNN
    - Facebook's Prophet Auto ML model

## 2.4 Member #4 Derek Pena

- Design REST API input and output for forecasting API
- Document API Specification
- Develop code to load would load the final forecasted Model for file, accept the input from the GET/POST function of the API and returns the output.

## 2.5 Member #5 Daniel Casto

- Develop Flask API to accept the input parameters as per the specification and return output as per the specification
- Valid Input Parameters
- Error Handling and Exception Handling for incorrect input or model failure to forecast