AIR QUALITY MONITORING AND ANALYSIS BASED PREDICTIVE SYSTEM

Progress Presentation-1 (50% Completion)

2024 - 078

Meet Our Team !

Supervisor : Ms. Chathurangika Kahandawarachi

Co-supervisor : Ms. Pipuni Wijeseri

Inthikhaff M. I. M

IT21058028

Rimas M. J. M

IT21003714

KUMARI J.M.D

IT21152832

Pasan M.G.R

IT21057588

SNAPS FROM THE FIELD VISITS

Air Quality Data

NE NBRO ESSD <nbro.essd@gmail.com>
4/1/2024 4:53 PM

To: mhdinthikaff@gmail.com

Save all attachments

- AAQ daily Data Colombo_MET... 71.84 KB
- AAQ daily data_Mobile... 108.07 KB
- AAQ hourly data_Mobil 254.81 KB
- Air Quality hourly data_... 216.08 KB

Dear Student,

Please find related docs.

Thanks & Best Regards

Environmental Studies and Services Division
National Building Research Organisation
No. 99/1, Jawatte Road,
Colombo 05
Tel: 0112588946, 0112503826 Fax: 0112588365
Ext : 617 - Head of Division
589 - Air Quality, Noise & Vibration Studies Unit
621 - Land & Water Quality Studies Unit

RESEARCH PROBLEM

- **How can air quality monitoring and analysis be integrated into a predictive system to forecast and mitigate environmental risks?**
- **How we define Air pollution with the creation of IoT device.**
- **How we can develop user-friendly and easily understandable Pollution map?**
- **How can we encourage people to travel with minimal levels of pollution ?**
- **How can we enhance transparency and raise awareness among the public regarding air pollution and pollution levels?**
- **How can we find a solution for school students who are being affected by air pollution?**

RESEARCH OBJECTIVES

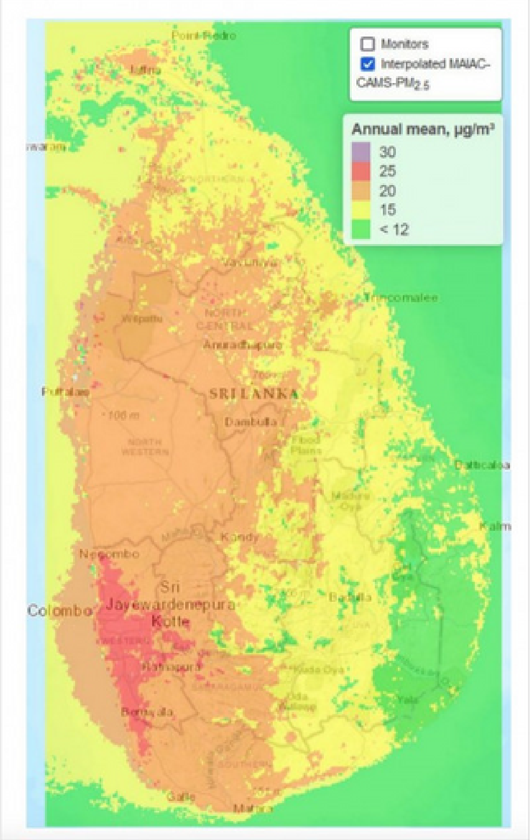
Main Objective

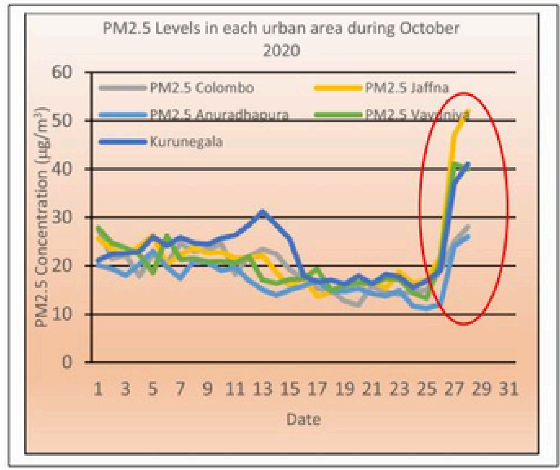
Minimize the adverse effects of air pollution for the general public population while travelling.

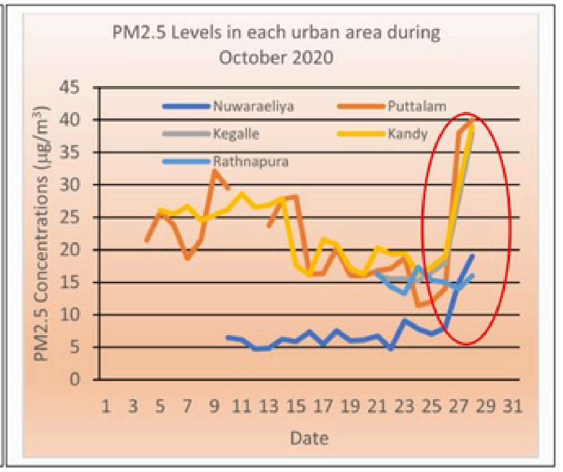
Sub Objectives

1. Develop sensor-based devices with a focus on accuracy and value creation on Air pollution.
2. Implement future pollution level prediction and forecasting mechanisms
3. Create a user-friendly real-time pollution heatmap accessible to all.
4. Develop real-time route generation based on pollution levels.

Significant of this study







Kriged MAIAC-CAMS-PM2S concentrations at 1 km intervals over Sri Lanka.

- **This map illustrates the air quality across Sri Lanka.**
- **Several areas are experiencing unexpected increases in air pollution levels.**

Addressing air pollution exposure among school children and the elders is a timely research topic.
By doing so, we can effectively mitigate the adverse effects of air pollution.

These table Illustrate the information about the selected disease categories and the discharge mode.

- More than one fourth Of the deaths are reported from the ischemic heart diseases.
- ***Almost one fourth Of the patients are discharged live those Who had the 'Other acute lower respiratory infections".***

Disease category		Live discharge	Transfers	Deaths	LAMA	Missing	Total
Acute upper respiratory infections	n	51023	2015	6	2	3	53049
	%	10.2%	5.1%	0.0%	2.7%	11.1%	9.5%
Influenza and pneumonia	n	12075	855	2328	4	0	15262
	%	2.4%	2.2%	15.8%	5.3%	0.0%	2.7%
Other acute lower respiratory infections	n	109910	5773	307	1	2	115993
	%	21.9%	14.7%	2.1%	1.3%	7.4%	20.8%
Other diseases of upper respiratory tract	n	10622	230	7	0	0	10859
	%	2.1%	0.6%	0.0%	0.0%	0.0%	2.0%
Chronic lower respiratory diseases	n	97055	4694	1182	9	4	102944
	%	19.3%	11.9%	8.0%	12.0%	14.8%	18.5%
Lung diseases due to external agents	n	646	43	805	0	0	1494
	%	0.1%	0.1%	5.5%	0.0%	0.0%	0.3%
Other respiratory diseases principally affecting the interstitium	n	631	24	120	1	0	776
	%	0.1%	0.1%	0.8%	1.3%	0.0%	0.1%
Suppurative and necrotic conditions of lower respiratory tract	n	590	28	28	0	0	646
	%	0.1%	0.1%	0.2%	0.0%	0.0%	0.1%
Other diseases of pleura	n	1265	115	30	1	0	1411
	%	0.3%	0.3%	0.2%	1.3%	0.0%	0.3%
Other diseases of the respiratory system	n	9554	432	140	2	0	10128
	%	1.9%	1.1%	1.0%	2.7%	0.0%	1.8%
Hypertensive diseases	n	39803	3350	393	9	5	43560
	%	7.9%	8.5%	2.7%	12.0%	18.5%	7.8%
Ischaemic heart diseases	n	45621	11424	3871	9	4	60929
	%	9.1%	29.1%	26.3%	12.0%	14.8%	11.0%
Cerebrovascular diseases	n	21388	6005	2270	12	1	29676
	%	4.3%	15.3%	15.4%	16.0%	3.7%	5.3%
Diabetes mellitus	n	37932	2345	626	7	6	40916
	%	7.6%	6.0%	4.3%	9.3%	22.2%	7.4%
Disorders of lens	n	41046	55	0	4	0	41105
	%	8.2%	0.1%	0.0%	5.3%	0.0%	7.4%
Malignant neoplasms of respiratory and intrathoracic organs	n	7619	140	536	10	2	8307
	%	1.5%	0.4%	3.6%	13.3%	7.4%	1.5%
Pulmonary heart disease and diseases of pulmonary circulation	n	2524	78	126	0	0	2728
	%	0.5%	0.2%	0.9%	0.0%	0.0%	0.5%
Other forms of heart disease	n	12977	1708	1929	4	0	16618
	%	2.6%	4.3%	13.1%	5.3%	0.0%	3.0%
Total	n	502281	39314	14704	75	27	556401
	%	100.0%	100.0%	100.0%	100.0%	100.0%	100.0%

Cross-tabulation of selected diseases and the discharge mode

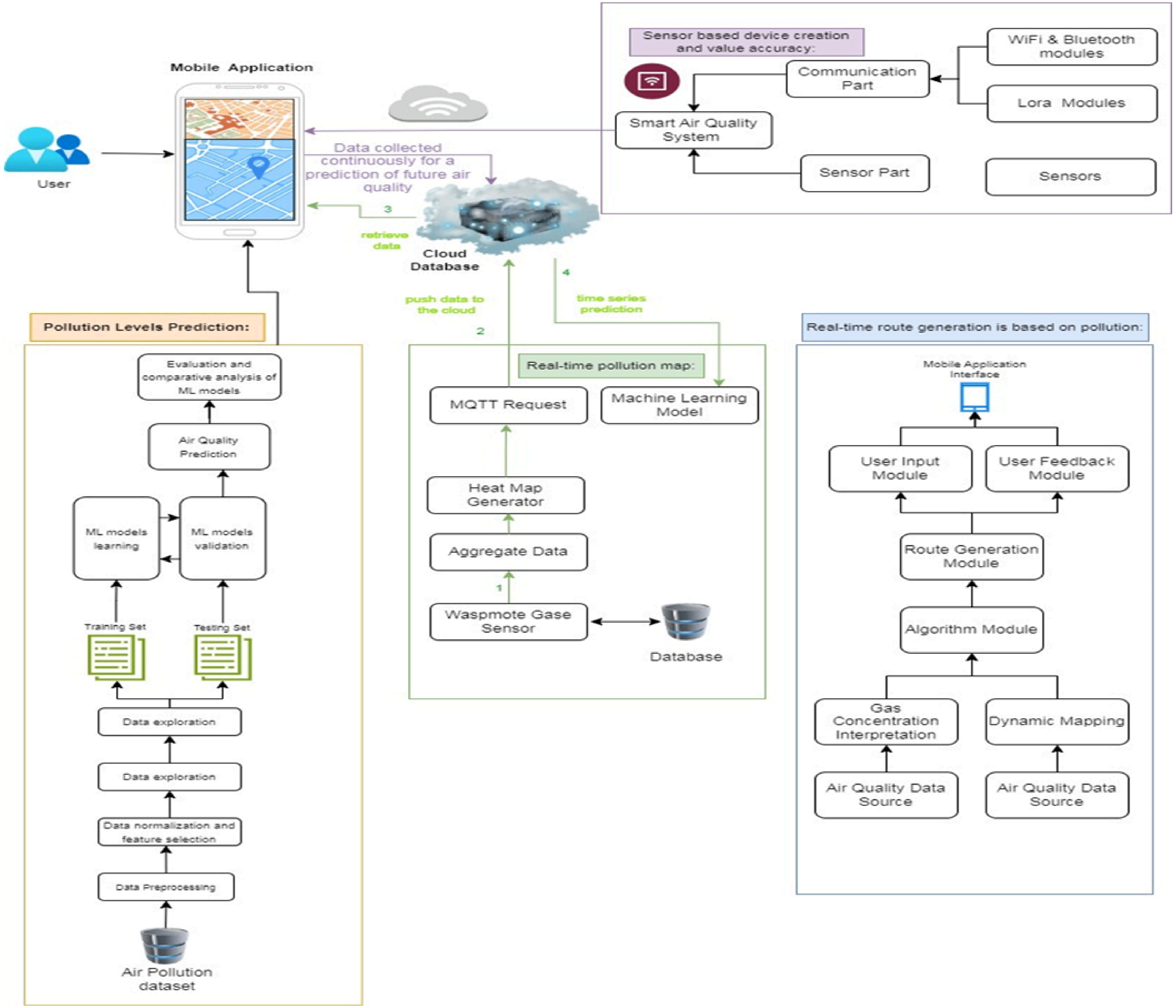
Comparison of deaths due to the air pollution related diseases and the N02 and S02 concentrations

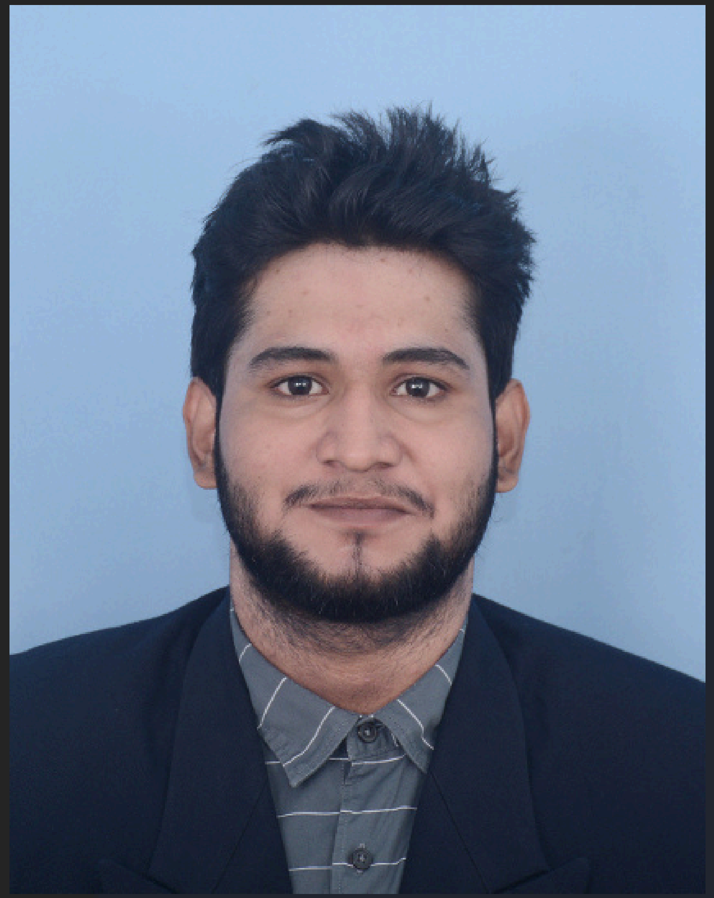
	Deaths	Population	Adjusted population*	Total estimated deaths**	Crude Death rate***	NO_2	SO_2
Anuradhapura	1261	918000	918000	1261	1.374	23	39
Colombo	2717	2419000	2177100	5846	2.685	39	51
Galle	1765	1113000	1113000	1765	1.586	27	48
Gampaha	2297	2391000	2151900	2297	1.067	32	49
Kalutara	1260	1271000	1143900	1260	1.101	32	38.5
Kandy	2002	1452000	1452000	2002	1.379	39	44
Kurunegala	2112	1694000	1694000	2112	1.247	41	49
Ratnapura	1290	1151000	1151000	1290	1.121	30	43

* Population consider the proportions obtain government health sector care
**Total deaths based on the estimation for NHSL
***Per 1000 population

Comparison of death rate and air quality levels in selected districts

System Diagram





IT21003714 | RIMAS M. J. M

Specialization : Information Technoogy

Component 2 : Sensor based device creation and value accuracy

INTRODUCTION

- The primary cause of Sri Lanka's ambient air pollution is emissions from vehicles. Air pollutants are many, the most important are **Particulate Matter (PM), Carbon Dioxide (CO_2), CH_4,** etc. which are found in the ambient air. Now it has begun a major environmental health problem that affects people
- in this issue, we are introduced to an IOT-based air Pollution Monitoring System in which we will monitor the Air pollution levels.
- These devices are considered automated devices that measure **CH_4, CO, CO_2, SO_2, NH_3, and NO_x** and record and use **MQ7, MQ4, and MQ135** information using sensors. With the real-time data monitoring ability of IoT
- get the real-time data from IoT and after converting this value to into a Parts per Million (PPM) concentration value
- this device consumes power so we can provide the power from Power Banks etc..

Introduction

Research Question

How effective is the IoT-based air quality monitoring hardware system in collecting and transmitting accurate air quality data?

- How effective are MQ4 and MQ7 gas sensors in measuring Methane (CH_4) and Carbon Monoxide (CO) concentrations in urban environments compared to other sensor technologies

Introduction

Specific and Sub Objective

Sensor based device creation and value accuracy

- **Integrate MQ4 and MQ7 sensors with an Arduino board and get the reading values**
- Set up MongoDB database to store sensor data securely
- **Send a Notification to the user when the device detects the pollution level is high**

Methodology

Existing Studies on sensor based device creation and value accuracyon

Author	Application	Techniques	Remark
Rauniyar, A., and colleagues	IoT-based car emissions monitoring system for smart cities	Reacts, a PostgreSQL database	85.2% of the vehicles were classified as normal emitters, 7.1% were medium emitters, while 7.7% of the vehicles were classified as high emitter vehicles
Kaivonen, S. and Ngai, E.C.H	real-time air pollution monitoring using wireless sensors on public transport vehicles	Python scripts, MongoDB, JSON, HTTP API, PHP	
Zhang, D. and Woo, S.S.	air quality pattern in the area using both moving and fixed IoT sensors mounted in the vehicles	mobile and stationary IoT sensors integrated into the vehicles	machine learning algorithms and real-time
Kumar, A	IoT-based system for monitoring vehicle pollution	proposed an Internet of Things (IoT) solution aimed at monitoring vehicle emissions	
Asha, P., et	IoT enabled environmental toxicology for air pollution monitoring using AI techniques	Artificial Algae Algorithm	model, which involves the monitoring of eight pollutants (NH3, CO, NO2, CH4, CO2, PM2.5, temperature, and humidity) through an IoT-based sensor
JunHo Jo	IoT-driven Smart-Air system for effective air quality monitoring and real-time	AWS server, PHP, MySQL, IoT-based, JavaScript, web server developed for Android OS	

existing studies

Methodology
System diagram

Used Techniques and Technologies

Techniques

- Conversion of sensor readings to sensor voltage values using Equation
- Use of MongoDB database for storing sensor data securely.
- Implementation of Arduino code on the Arduino Uno board to handle sensor inputs, process data,

Technologies

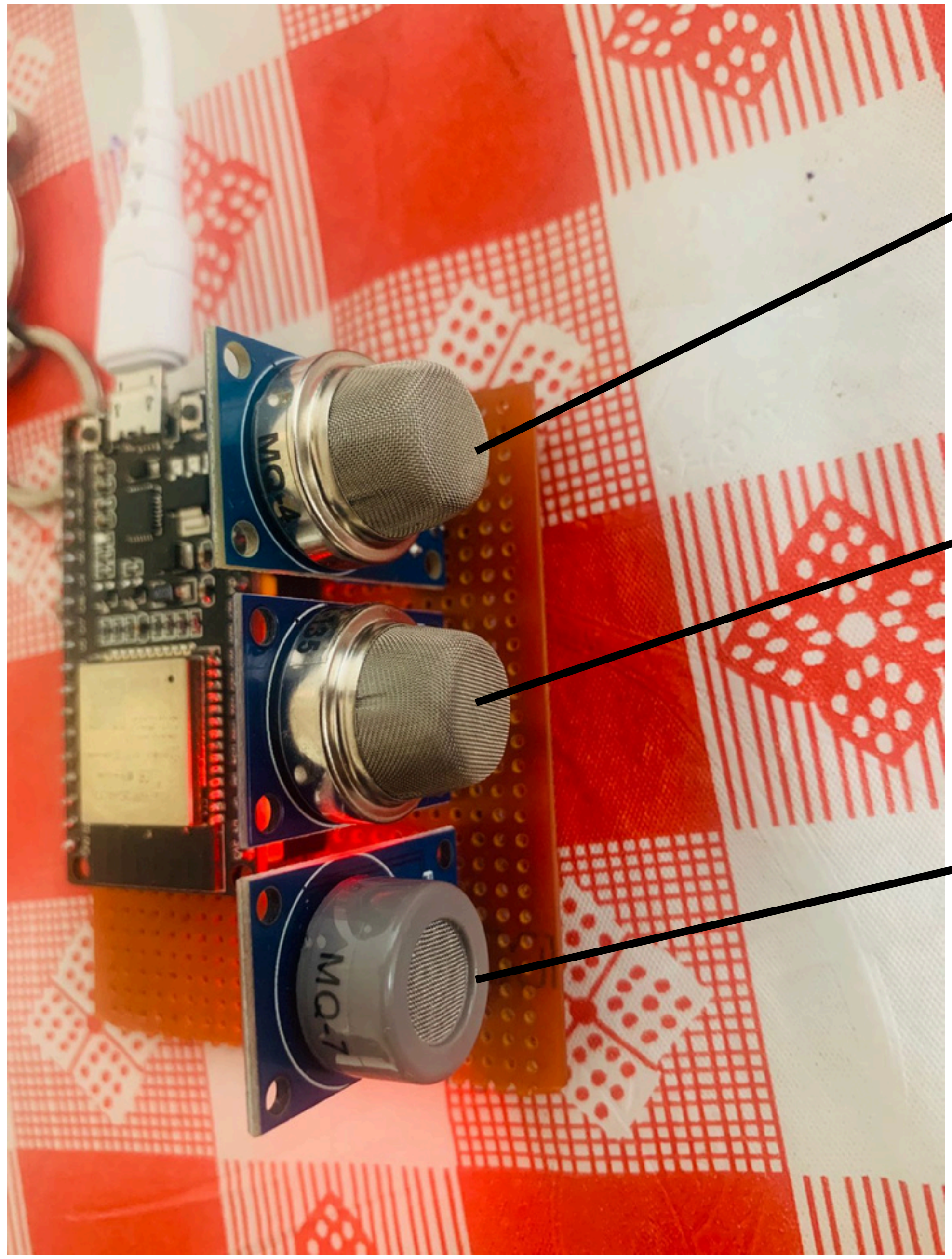
- React Native
- MongoDB
- Node Js
- Postman API
- Arduino IDE
- Visual Studio Code

Methodology

Evidence of Completion

- create the IoT device
- Connect with Backend server from IoT devices and send the sensor data in database(DB)
- get the real-time Sensor Data from IoT devices
- Host the finalized model in the fast server
- calculate the sensor data values (PPM) concentration
- Displayed the results in mobile application
- Realtime sensor data accuracy level

Evidence of Completion



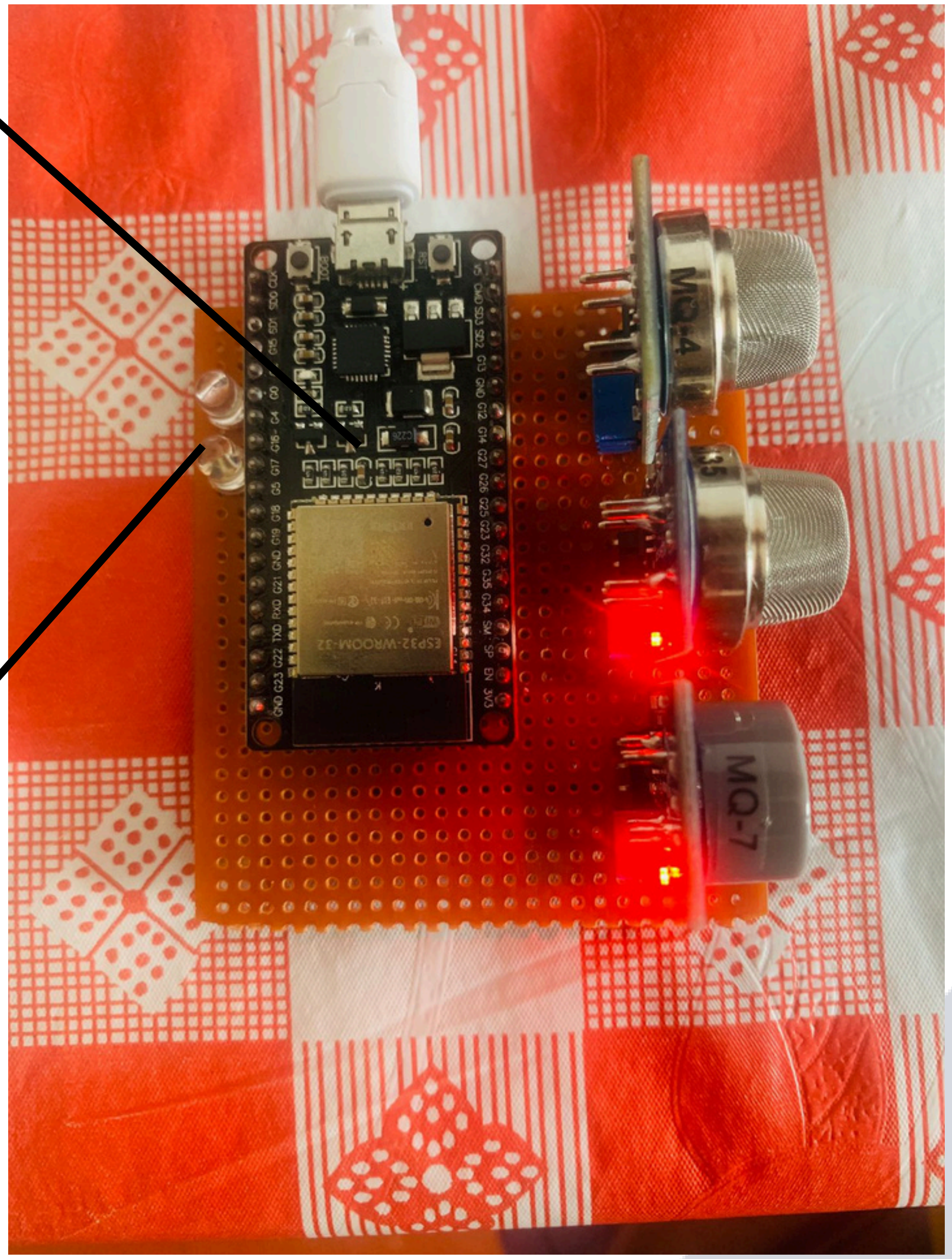
MQ4 sensor
measure CH4

MQ135 measure g
CO, SO2, NH3, NOx

MQ7 measure
co2 and co

ESP32
boards

LED light



16

Evidence of Completion

We predicted the air pollution real-time data from IoT and after converting this value to The output analog value (0 - 4095) of each sensor should be converted into sensor voltage value

Sensor Voltage = (Analog Reading * 3.3V) / 4095

After sensor voltage calculation, we can convert that value into a Parts per Million (PPM) concentration value

CH4 concentration = 10.938 * e(1.7742*V_RL)

CO concentration = 3.027 * e(1.0698*V_RL)

Evidence of Completion

National Ambient air quality Standards

Pollutant	Averaging time	Maximum Permissible level	
		µg/m-3	ppm
Particulate Matter – PM_{10}	Annual	50	
	24 hr	100	
$PM_{2.5}$	Annual	25	
	24 hr	50	
Nitrogen Dioxide (NO_2)	24 hr	100	0.05
	8 hr	150	0.08

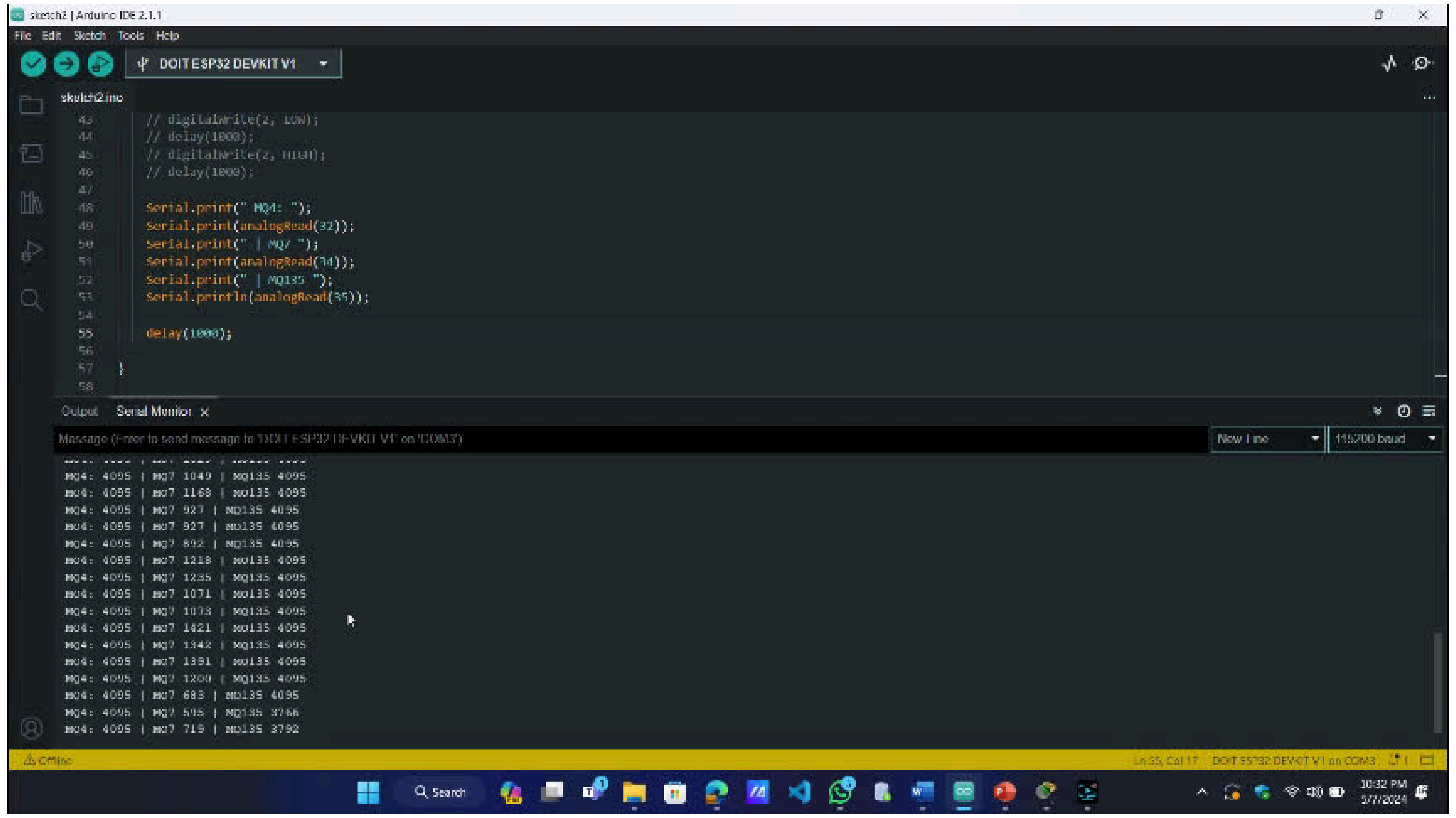
	1 hr	250	0.13
Sulphur Dioxide (SO_2)	24 hr	80	0.03
	8hr	120	0.05
	1 hr	200	0.08
Ozone (O_3)	1 hr	200	0.10
Carbon Monoxide (CO)	8 hrs	10,000	9.00
	1 hr	30,000	26.00
	anytime	58,000	50.00

Methane	
Ranges	Quality
Below 600 PPM	Good
600-800 PPM	Okay
800-1000 PPM	Acceptable
Above 1000 PPM	Dangerous

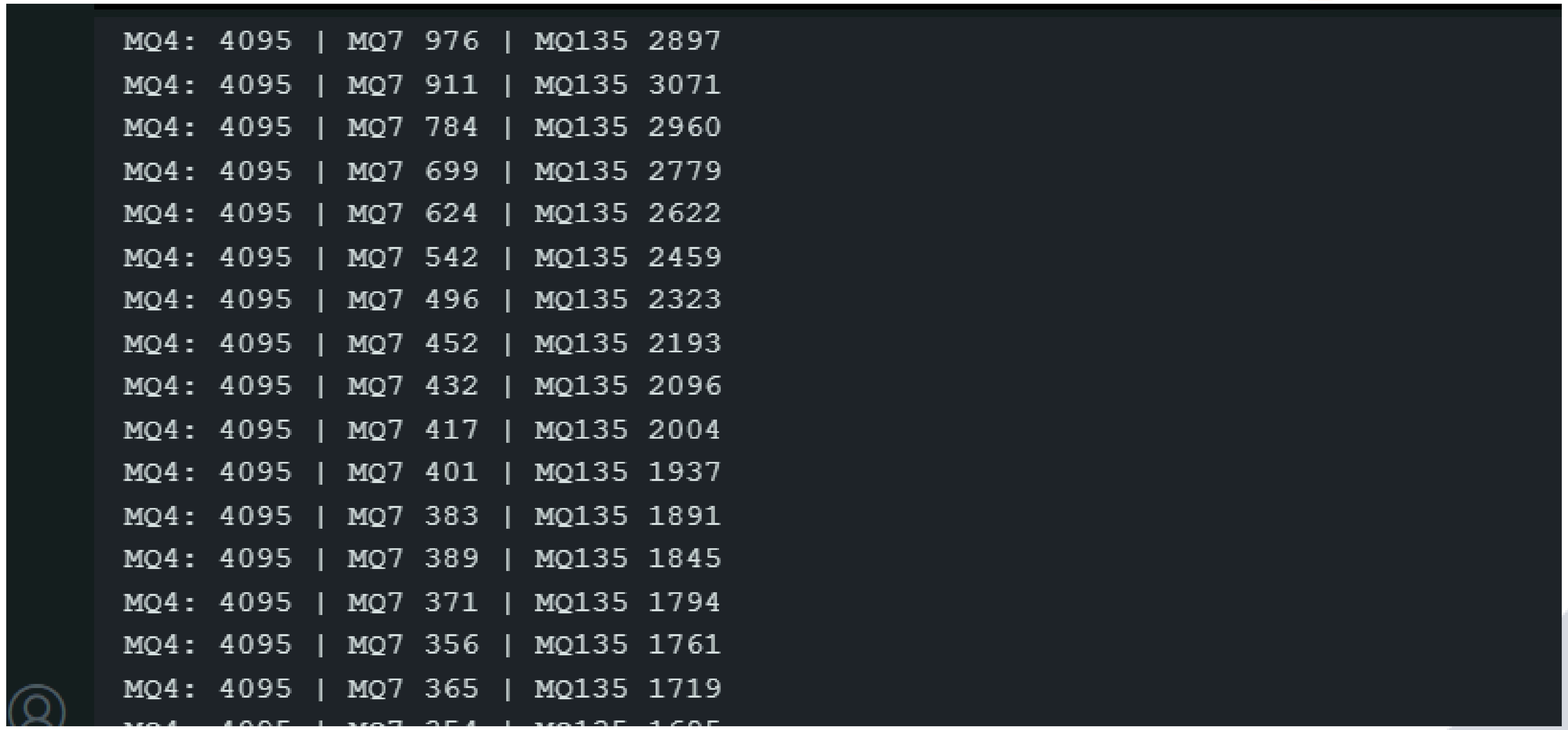
after calculating the concentration level and check the air quality

REAL TIME DATA COLLECTION PARTICULAR LOCATION

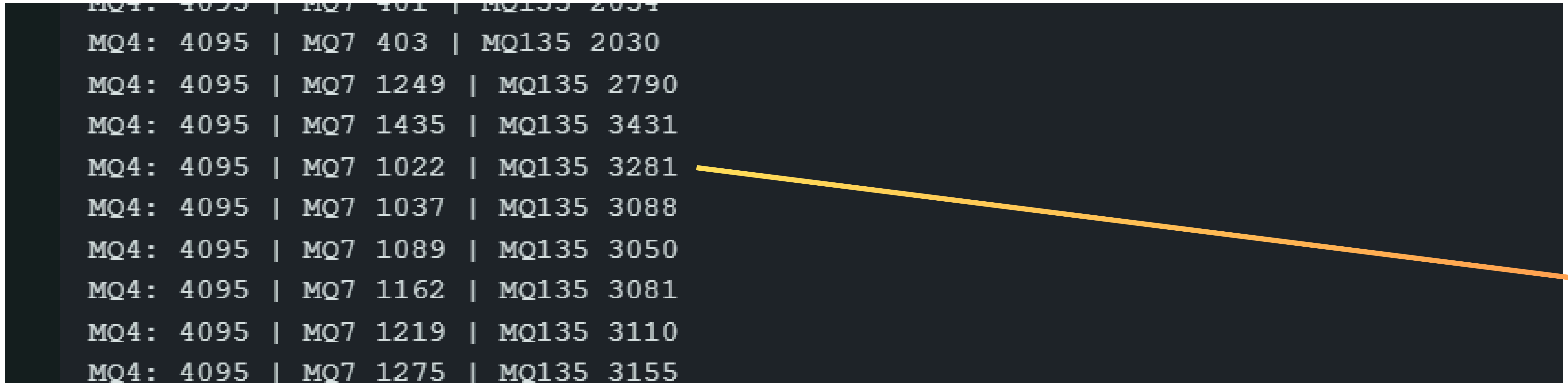
Evidence of Completion



get the data in night time



get the data in day time



get the data in evening time

the reding velues is high

Functional, Non-Functional and Personnel Requirements

Functional Requirements

- **Integrate MQ4 and MQ7 sensors in IoT devices**
- **implement data processing algorithms for sensor data conversion**
- **Set up MongoDB database to store real-time air quality data.**
- **Calculating the PPM value**
- **Implement data transmission at regular intervals (e.g., every one minute) to maintain up-to-date information**

Non-Functional Requirements

- User-friendly interfaces.
- Provide continuous operation and data transmission
- Ensure compatibility with different mobile platforms (iOS, Android).

Personnel Requirements

- **Identify the sensors and board details to creat an IoT device.**

Completion and Future works

50% Completion of the components

- Build an IoT Device
- Reading the data from MQ4 AND MQ7 AND MQ135 sensors
- Calculating the PPM value from meter Readings

50% Future Implementations

- Connect with the Backend server from IoT devices and send the sensor data to the database
- Displayed the real-time pollution level in the mobile application
- Send the Alert Notification when reading high pollution level

References

[1] Naik, U. U., Salgaokar, S. R., & Jambhale, S. (2023). IOT based air pollution monitoring system. Int. J. Sci. Res. Eng. Trends, 9, 835-838.

[2] Pandithurai, O., Jawahar, M., Arockiaraj, S., & Bhavani, R. (2023). IOT Technology Based Vehicle Pollution Monitoring and Control (Doctoral dissertation, Department of Electrical and Electronics Engineering, Mepco Schlenk Engineering College, Sivakasi).

[3] Dhanalakshmi, M. (2021). A survey paper on vehicles emitting air quality and prevention of air pollution by using IoT along with machine learning approaches. Turkish Journal of Computer and Mathematics Education (TURCOMAT), 12(11), 5950-5962.

[4] Potbhare, Piyush Devidas, et al. "IoT based Smart Air Pollution Monitoring System." 2022 International Conference on Applied Artificial Intelligence and Computing (ICAAIC). IEEE, 2022..

[5] [1]Mihăiţă, A. S., Dupont, L., Chery, O., Camargo, M., & Cai, C. (2019). Evaluating air quality by combining stationery, smart mobile pollution monitoring and data-driven modelling. Journal of cleaner production, 221, 398-418..

[6] [1] .N.S. Attanayake and R.A.B. Abeygunawardana, A Comprehensive Comparison of Air Pollution in Main Cities in Sri Lanka Department of Statistics, Faculty of Science, University of Colombo, Colombo 03, Sri Lanka

IT21058028 | Inthikhaff M. I. M

Specialization : Information Technoogy

Component 1 : Air Pollution Level Prediction

Introduction

Background

- **This component main objective is to predict the Air Pollution Level** using regression models such as Random Forest, Linear Regression, Xgbhoost, Support Vactor Regression and Prophet Method.
- Our primary focus is on enhancing existing methods to accurately predict and understand air quality.
- Colombo consistently experiences an average unhealthy AQI, making reliable predictions crucial for guiding public precautions regarding indoor and outdoor activities.
- The correlation between AQI and threats to human health, including short-term and long-term effects, emphasizes the importance of accurate predictions.
- Through out this various Model training, we are finding the best model which gives high accuracy in air pollution level prediction.

Introduction

Research Question

How to predict air pollution levels of a given time in future and forecasted?

How to filter which inputs(atributes) perform more side effect on air pollution

Introduction

Specific and Sub Objective

Implement future pollution level prediction and forecasting mechanisms

- **Identification of data set, cleaning and preprocessing.**
- **Training data set using different model and Finding best model which gives high level accuracy.**
- **Graphical and comparative visual representation.**
- **Notification to user when predicted polution level is high**

Methodology

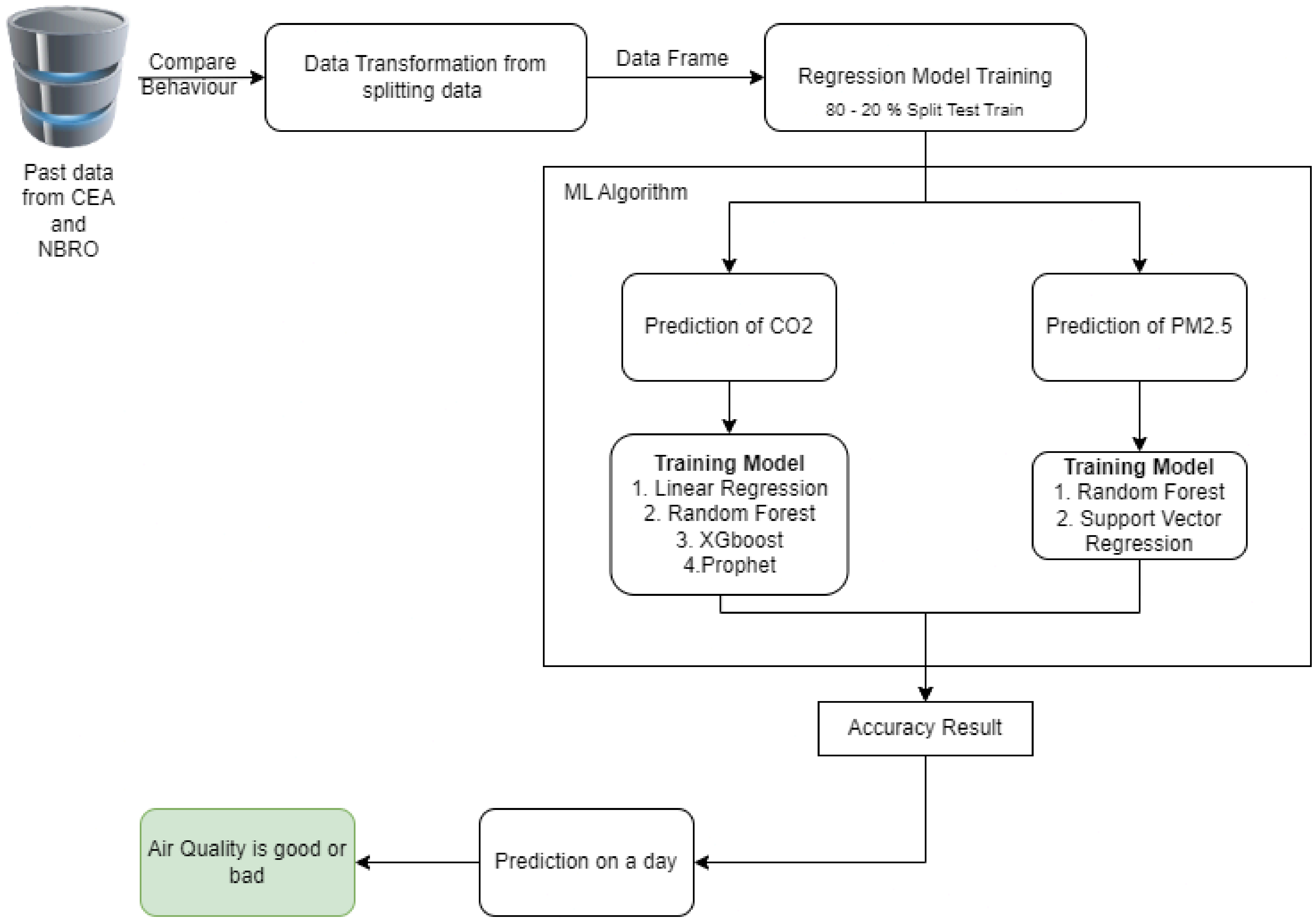
Existing Studies on Pollution Level Prediction

Author	Application	Techniques	Remark
Min Lee and others	Air pollution prediction	• Deep Learning	• Predict against PM 2.5, PM 10 particulars. • Accuracy based on PM 10 is very low. • Accuracy based on PM2.5 is very high.
Usha Mahalingam and others	Air quality prediction	• Neural Networks • Support Vector Machine	• Accuracy of 91.62% for neural network. • Accuracy of 97.3% for support vector machine
S. Silva and others	Air quality prediction for smart cities	• Support Vector Regression	• Predict PM 2.5 levels variability. • Model is suitable for predict hourly air pollution. • Obtain an accuracy of 94.1%
Timothy M. A. and others	Development of Air Quality Monitoring model	• Naive Bayesian • KNN • Support Vector Machines • Neural Networks • Random Forest	• Highest accuracy was obtained through Neural Networks. • Sometimes Neural Network leads slower response.
C. Zhao and others	Air Quality Index Prediction	• Linear regression	• AQI Prediction based on a year data of PM2.5, PM10 etc. • There is a deviation between predicted results and actual date.
Ismail Ahmadi	Air pollution prediction	• Data mining • Decision Tree	• Used clementine software for data clustering. • Data sample include climate data
Colin Belinger and others	A systematic review based on Machine Learning and data mining for Air Pollution.	• Machine Learning Algorithms • Data Mining • Big Data	• Refer 400 research papers & Reduce to 47 after the inclusion/exclusion criteria's. • Divided research papers into three categories • End of the Literature survey that highest accuracy levels always obtain in Machine Learning Algorithms based approaches.

	Time	CO2	PM2.5	PM10	Unnamed: 4	Unnamed: 5	Unnamed: 6	Unnamed: 7	Unnamed: 8
0	2022-12-21 00:00:00	562	15	16.0	NaN	NaN	NaN	NaN	NaN
1	2022-12-21 01:00:00	511	35	40.0	NaN	NaN	NaN	NA- Not Available	NaN
2	2022-12-21 02:00:00	511	36	42.0	NaN	NaN	NaN	MF- Malfunctioned	NaN
3	2022-12-21 03:00:00	507	38	44.0	NaN	NaN	NaN	PM10	Units - µg/m3
4	2022-12-21 04:00:00	498	39	46.0	NaN	NaN	NaN	PM2.5	Units - µg/m3
...	...	...	...	...	...	...	...	...	...
5938	2024-03-21 09:00:00	MF	42	50.0	NaN	NaN	NaN	NaN	NaN
5939	2024-03-21 10:00:00	MF	38	45.0	NaN	NaN	NaN	NaN	NaN
5940	2024-03-21 11:00:00	MF	34	38.0	NaN	NaN	NaN	NaN	NaN
5941	2024-03-21 12:00:00	MF	33	37.0	NaN	NaN	NaN	NaN	NaN
5942	2024-03-21 13:00:00	MF	26	27.0	NaN	NaN	NaN	NaN	NaN

From these I have used Time, CO2, PM 2.5 and PM 10 for prediction

Methodology
System Diagram



Used Techniques and Technologies

Techniques

- DataCollection
- Data Preprocessing
- Normalization
- Regularization
- Algorithm tuning

Algorithm

- Linear Regression
- Random Forest
- XGBoost (Extreme Gradient Boosting)
- Support Vector Regressor
- Prophet (Facebook's Forecasting Model)

Technologies

- React Native
- Python
- Fast Server
- Node Server
- Jupyter Notebook
- Google Colab
- VS code

Methodology

Evidence of Completion

- Data collection
- Data pre-processing
- Model Training
- Model Evaluation
- Model Testing
- Data Visualization
- Host the finalized model in the fast server
- Displayed the results in mobile application

Data Collection and Pre-processing

- **Data pre-processing implemetation**

```
# Differentiate with time vs predicted variable(creating seperate)

PM_25_dataframe = dataframe_cleaned[['Time',"PM2.5"]]
PM_10_dataframe = dataframe_cleaned[["Time","PM10"]]
CO2_dataframe = dataframe_cleaned[['Time','CO2']]
```

```
#This code filters the DataFrame CO2_dataframe to remove rows where the value in the "CO2" column is equal to "MF".
condition = CO2_dataframe["CO2"] == "MF"
CO2_dataframe = CO2_dataframe[~condition]
```

```
condition = PM_25_dataframe["PM2.5"] == "MF"
PM_25_dataframe = PM_25_dataframe[~condition]
```

```
PM_25_dataframe.dropna(inplace=True)
```

```
<ipython-input-116-56ec3935287e>:1: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  PM_25_dataframe.dropna(inplace=True)
```

```
PM_25_dataframe
```

	Time	PM2.5
0	2022-12-21 00:00:00	15
1	2022-12-21 01:00:00	35
2	2022-12-21 02:00:00	36
3	2022-12-21 03:00:00	38
4	2022-12-21 04:00:00	39
...	...	...
5938	2024-03-21 09:00:00	42
5939	2024-03-21 10:00:00	38
5940	2024-03-21 11:00:00	34
5941	2024-03-21 12:00:00	33
5942	2024-03-21 13:00:00	26

Classes	Train, Validation and Test data
CO2 (5702)	Train - 4562 Test - 1140
PM 2.5 (3057)	Train - 2446 Test - 611
PM 10 (5703)	Train - 4563 Test - 1140

- **Data collection implemetation**

```
from google.colab import drive
drive.mount('/content/drive')
```

```
Mounted at /content/drive
```

```
Start coding or generate with AI.
```

oad the hourly data set and clean the data

```
import pandas as pd

dataframe = pd.read_excel(r"/content/drive/MyDrive/air_pollution_project/Air Quality hourly data_ Colombo_MET Dept.xlsx")
```

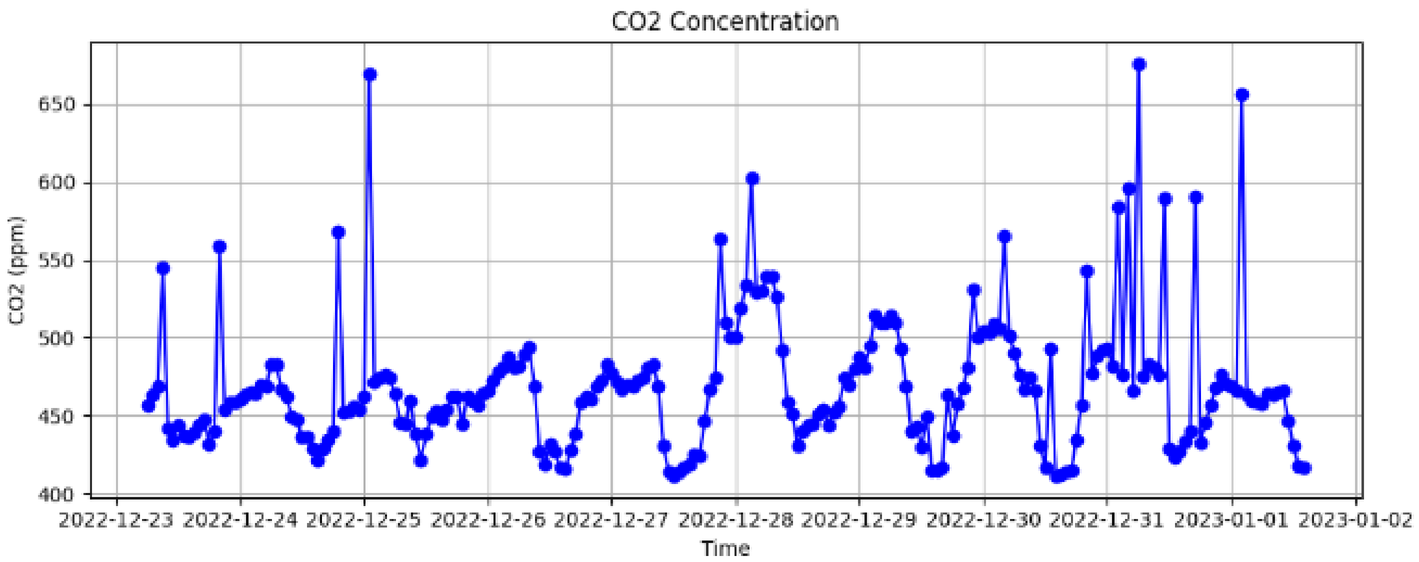
```
dataframe
```

	Time	CO2	PM2.5	PM10	Unnamed: 4	Unnamed: 5	Unnamed: 6	Unnamed: 7	Unnamed: 8
0	2022-12-21 00:00:00	562	15	16.0	NaN	NaN	NaN	NaN	NaN
1	2022-12-21 01:00:00	511	35	40.0	NaN	NaN	NaN	NA- Not Available	NaN
2	2022-12-21 02:00:00	511	36	42.0	NaN	NaN	NaN	MF- Malfunctioned	NaN
3	2022-12-21 03:00:00	507	38	44.0	NaN	NaN	NaN	PM10	Units - µg/m3
4	2022-12-21 04:00:00	498	39	46.0	NaN	NaN	NaN	PM2.5	Units - µg/m3
...	...	...	...	...	...	...	...	...	...
5938	2024-03-21 09:00:00	MF	42	50.0	NaN	NaN	NaN	NaN	NaN
5939	2024-03-21 10:00:00	MF	38	45.0	NaN	NaN	NaN	NaN	NaN
5940	2024-03-21 11:00:00	MF	34	38.0	NaN	NaN	NaN	NaN	NaN
5941	2024-03-21 12:00:00	MF	33	37.0	NaN	NaN	NaN	NaN	NaN
5942	2024-03-21 13:00:00	MF	26	27.0	NaN	NaN	NaN	NaN	NaN

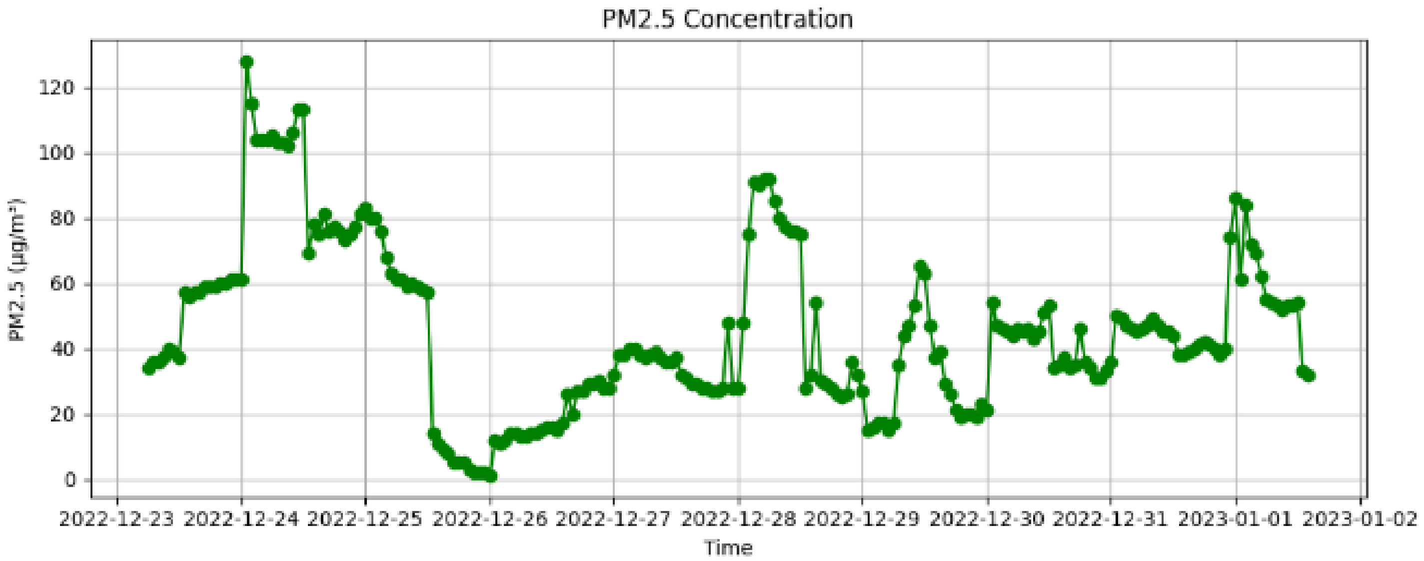
5943 rows × 9 columns

Total Data : 5943 | Preprocessed Data : 5703 | 80 -20 % Train Test Split

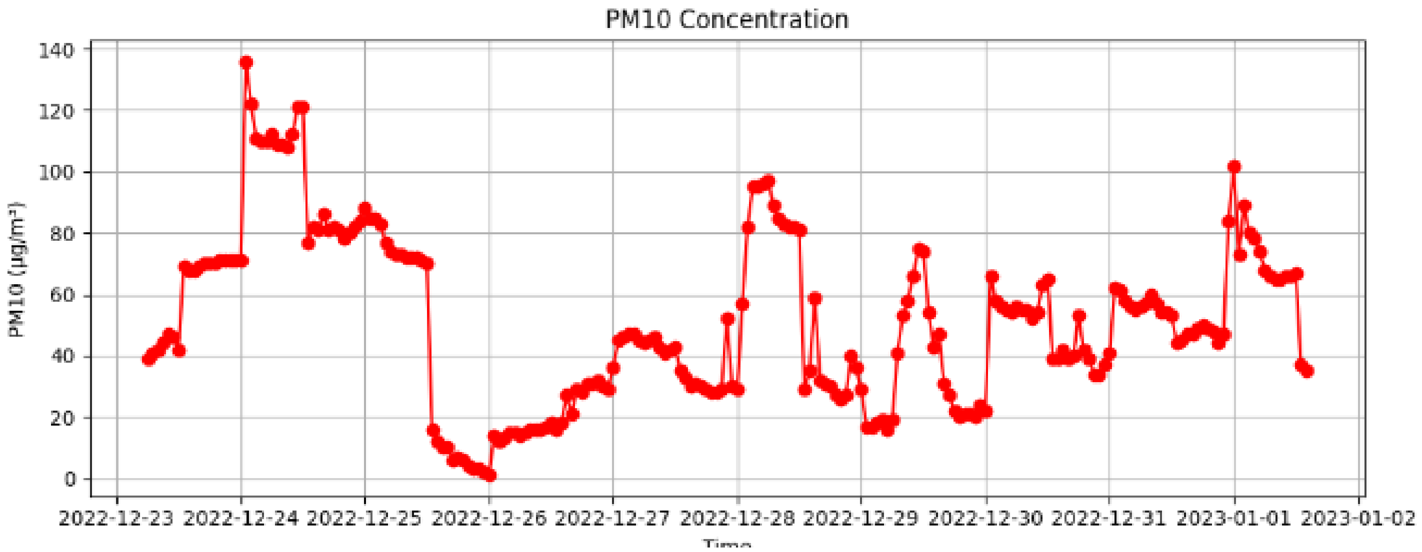
Actual Representation of Data
Evidence of Completion



CO2



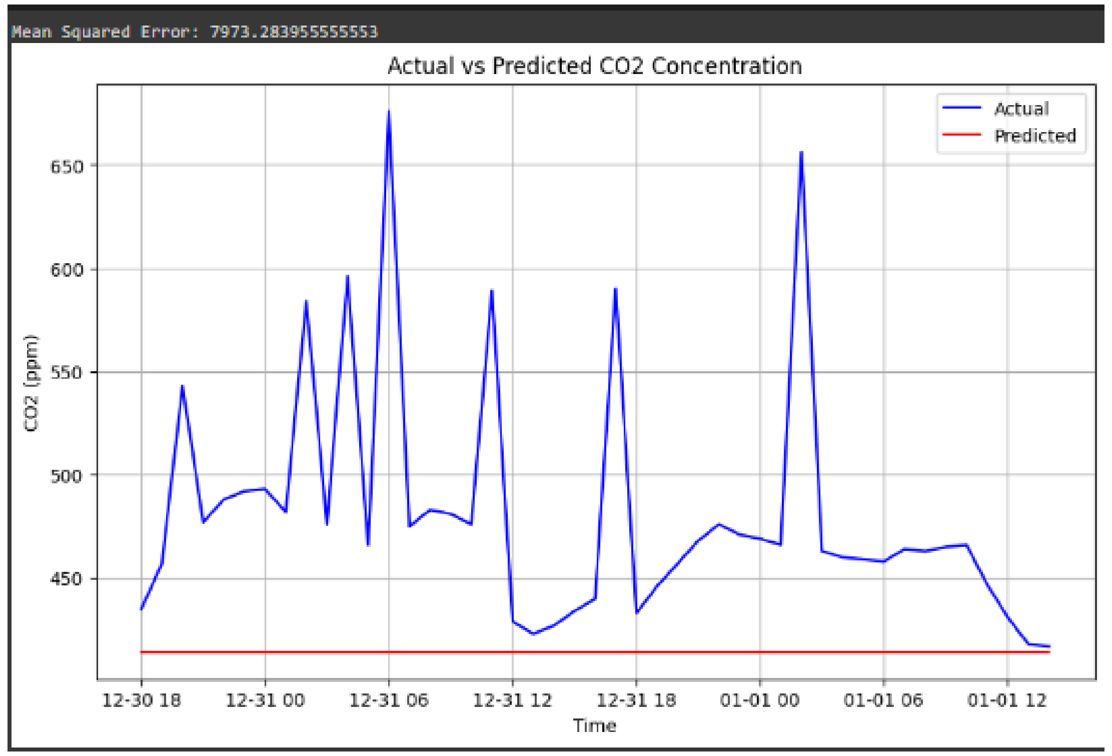
PM 2.5



PM 10

Train Model - CO2
Evidence of Completion

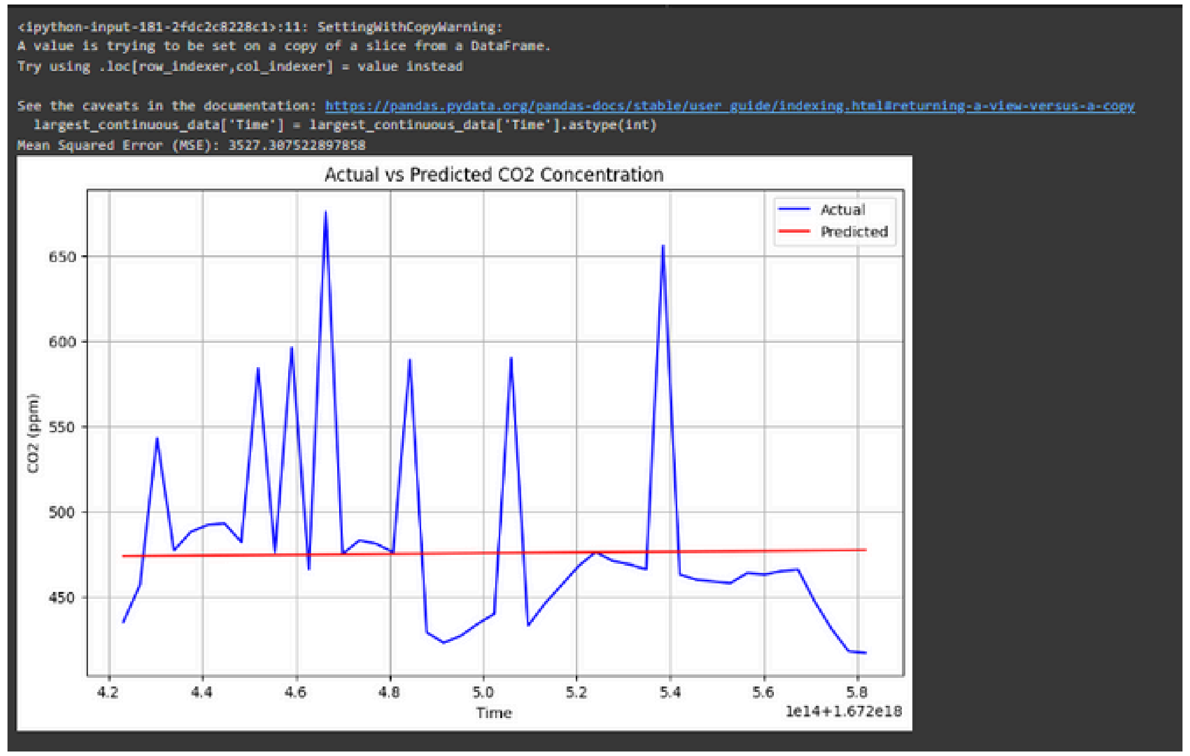
Random Forest




```
Mean Squared Error: 7973.283955555553
```

MSE - 7973.28

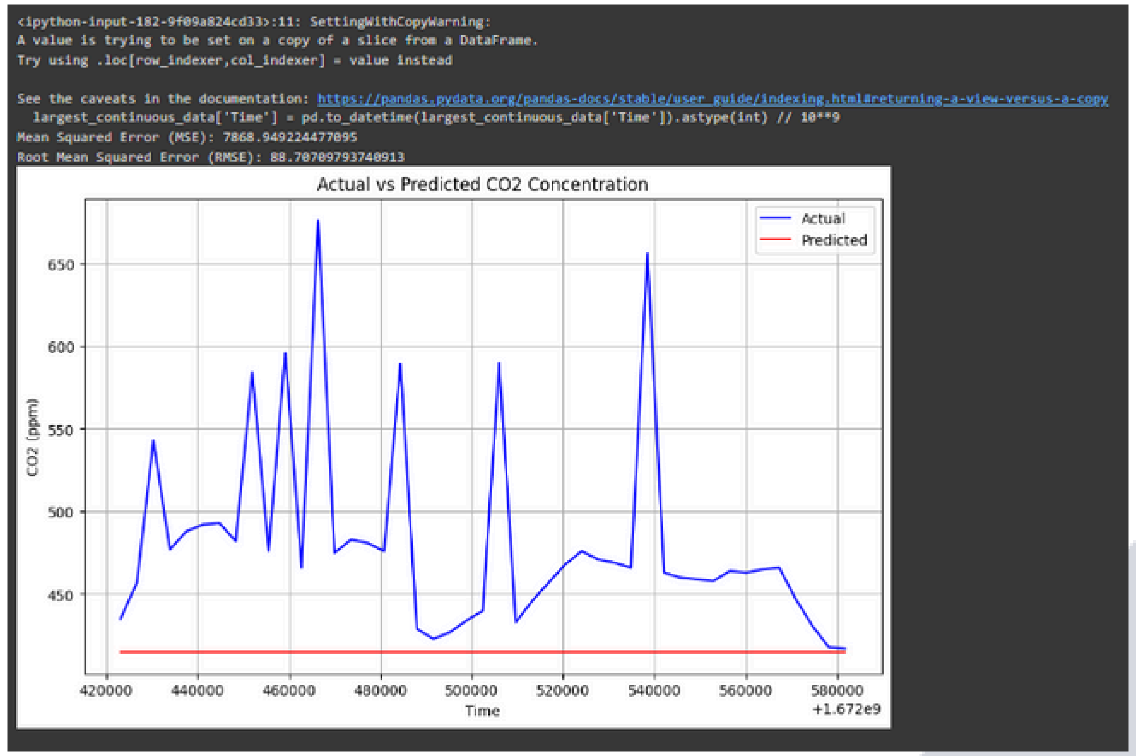
Linear Regression




```
<ipython-input-181-2fdc2c8228c1>:11: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  largest_continuous_data['Time'] = largest_continuous_data['Time'].astype(int)
Mean Squared Error (MSE): 3527.307522897858
```

MSE - 3527.31

XGbhoost



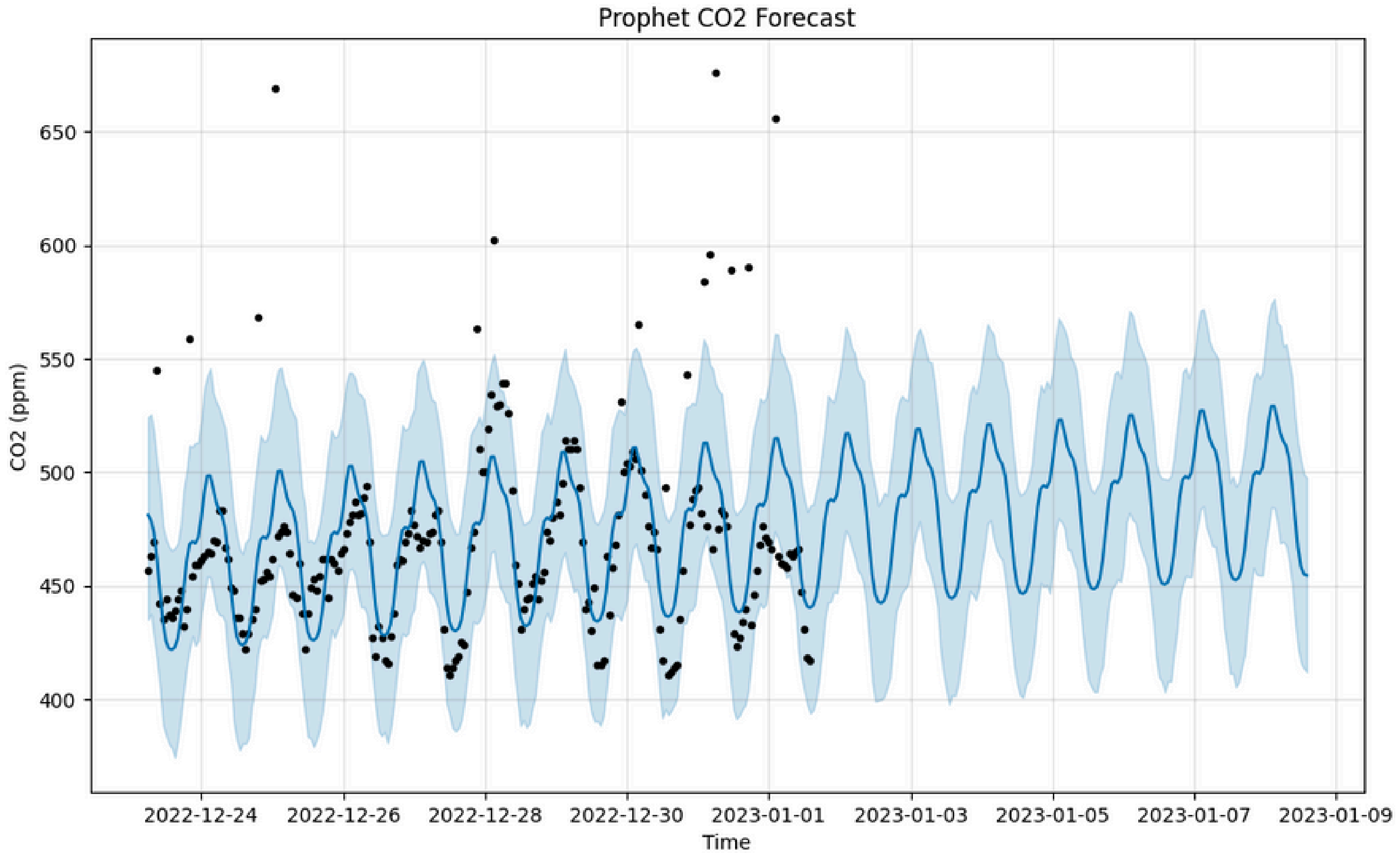

```
<ipython-input-182-9f09a824cd33>:11: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  largest_continuous_data['Time'] = pd.to_datetime(largest_continuous_data['Time']).astype(int) // 10**9
Mean Squared Error (MSE): 7868.949224477095
Root Mean Squared Error (RMSE): 88.70709793740913
```

MSE - 88.71

Lover value indicates a better fit in MSE

Train Model - CO2
Evidence of Completion

Prophet



Root Mean Squared Error (RMSE): 54.452076807335985

MSE - 54.45

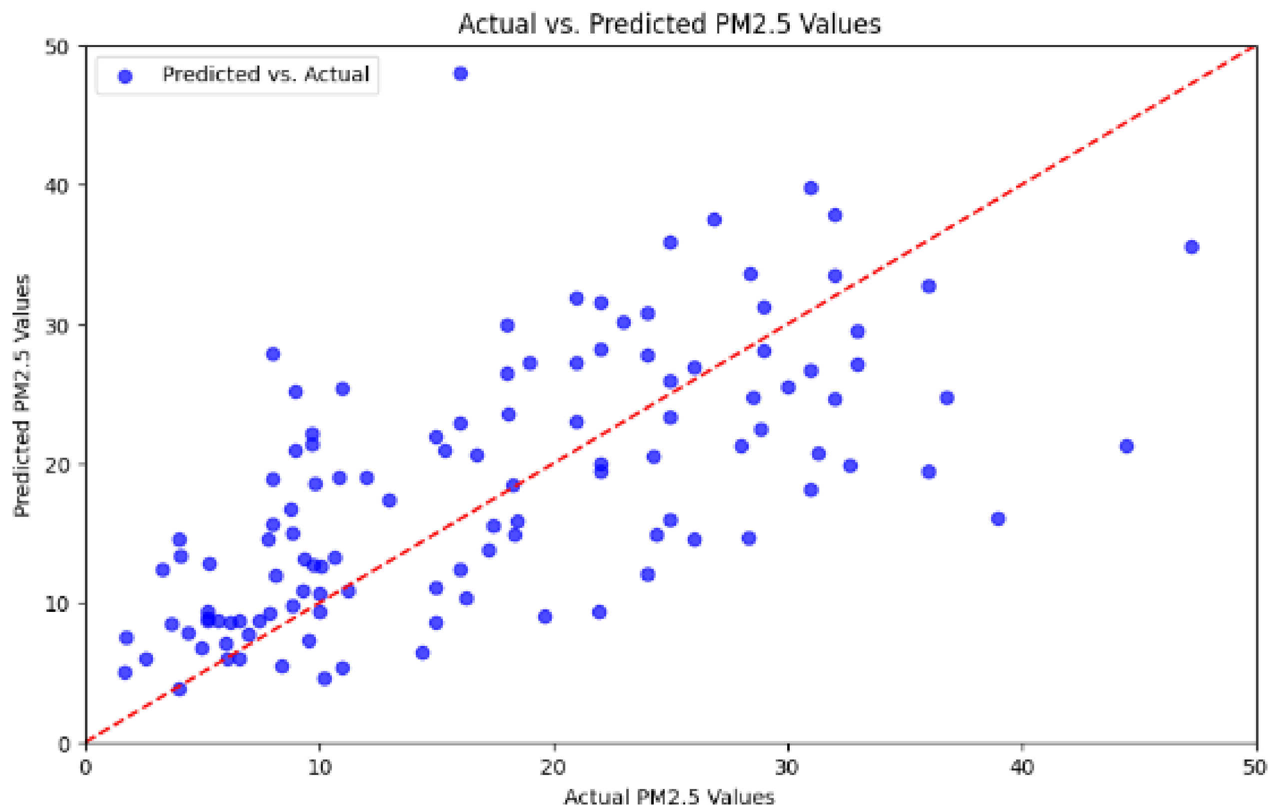
According to MSE values

Random Forest < Linear Regression < XGbhoost < Prophet

Lover value indicates a better fit in MSE

Training Model - PM 2.5
Evidence of Completion

Random Forest




```
R-squared score: 0.362
Mean Squared Error: 123.266
```

MSE - 123.26

Suppor Vector Regressor

```python
from sklearn.svm import SVR
from sklearn.metrics import mean_squared_error

# Create and train the SVR model
svr_model = SVR(kernel='rbf')
svr_model.fit(X_train, y_train)

# Make predictions on the test set
y_pred_svr = svr_model.predict(X_test)

# Calculate the Mean Squared Error
r2 = r2_score(y_test, y_pred_svr)
print(f"R-squared score: {r2:.3f}")

mse_svr = mean_squared_error(y_test, y_pred_svr)
print(f"Mean Squared Error for SVR: {mse_svr:.3f}")
```

```
R-squared score: 0.316
Mean Squared Error for SVR: 132.065
```

RMSE - 132.065

Lover value indicates a better fit in MSE

Functional, Non-Functional and Personnel Requirements

Functional Requirements

- Polluted Air Data Collection and Preprocessing(clean and transform)
- Feature Selection and Analysis
- Model Development and Training
- Explore and select the most suitable predictive model based on accuracy metrics.
- Dynamic Predictions for Real-time Adjustments.
- Real-time Pollution Data and Comparison and Visualization.

Non-Functional Requirements

- User-friendly Interfaces.
- Should properly work in cross platforms (for android and IOS devices)
- The application should be reliable.
- Higher accuracy of results.
- Results should be more efficient.

Personnel Requirements

Resources and Dataset Air pollution level in Colombo

- Central Environment Authority (CEA)
- The National Building Research Organization (NBRO).

Completion and Future works

65%

Completion of the components

- Polluted Air Data Collection and Preprocessing, Cleaning and Transforming.
- Identification of the best Model for predicting CO2 level in the environment - (Sub-Objective 02
- Identification of the best model for predicting PM 2.5 - (Sub-Objective 03)

35%

Future Implementations

- Host the finalized model in the fast server
- Displayed the results in mobile application

References

[1] S. Mahanta, T. Ramakrishnudu, R. R. Jha, and N.Tailor, “Urban Air Quality Prediction Using RegressionAnalysis,” in TENCON 2019 - 2019 IEEERegion 10 Conference (TENCON), Kochi, India,Oct. 2019, pp. 1118–1123. doi: 10.1109/TENCON.2019.8929517.

[2] M. Castelli, F. M. Clemente, A. Popoviˇc, S. Silva,and L. Vanneschi, “A Machine Learning Approachto Predict Air Quality in California,” Complexity,vol. 2020, pp. 1–23, Aug. 2020, doi:10.1155/2020/8049504.

[3] S. Zhong, Z. Yu, and W. Zhu, “Study of the Effectsof Air Pollutants on Human Health Based onBaidu Indices of Disease Symptoms and Air QualityMonitoring Data in Beijing, China,” IJERPH,vol. 16, no. 6, p. 1014, Mar. 2019, doi:10.3390/ijerph16061014.

[4] Y. L. S. Nandasena, A. R. Wickremasinghe, andN. Sathiakumar, “RAesierarpchoarltilculetion andhealth in Sri Lanka: a review of epidemiologic studies,”p. 14, 2010.

[5] T. Xayasouk and H. Lee, “AIR POLLUTION PREDICTIONSYSTEM USING DEEP LEARNING,”Naples, Italy, Jun. 2018, pp. 71–79. doi:10.2495/AIR180071.

[6] D. Iskandaryan, F. Ramos, and S. Trilles, “AirQuality Prediction in Smart Cities Using MachineLearning Technologies based on Sensor Data: AReview,” Applied Sciences, vol. 10, no. 7, p. 2401,Apr. 2020, doi: 10.3390/app10072401.

IT21152832 | Kumari J.M.D

Specialization : Information Technology

Component 3 : Realtime Pollution Map(Heatmap)

Introduction

Background

- The Real-Time Pollution Heatmap addresses the urgent need for accessible air quality information in our urbanizing world. With increasing vehicle emissions to health and the environment from air pollution.

- Collecting the values of CH_4, CO_2, SO_2, NO_x gases and PM2.5 in real-time, providing with up-to-date pollution data through visualizations and personalized recommendations in realtime pollution map, users can minimize exposure.

- Gas concentrations and PM2.5 values are displayed visually with user-friendly techniques(icons,numerical labels) indicating whether values are high or low. Real-time updates ensure immediate detection of high pollution areas.

- Air pollution levels are represented with color-coded schemes to help identifying about air pollution.

- Personalized recommendations suggest optimal travel times, minimizing exposure. Users can explore weekly pollutant exposure through a graph, identifying exposure trends. Color-coded indicators highlight periods of high exposure, empowering user to take proactive steps for their health and well-being.

Research Question

What are the different techniques to represent gas concentration levels and air quality which make user easily understandable?

what method can be used for personalized recommendations on travelling in low pollution time with low exposure of effect ?

How to visualize Summary of Exposure to the Air pollution?

Specific and Sub Objective

Specific Objective

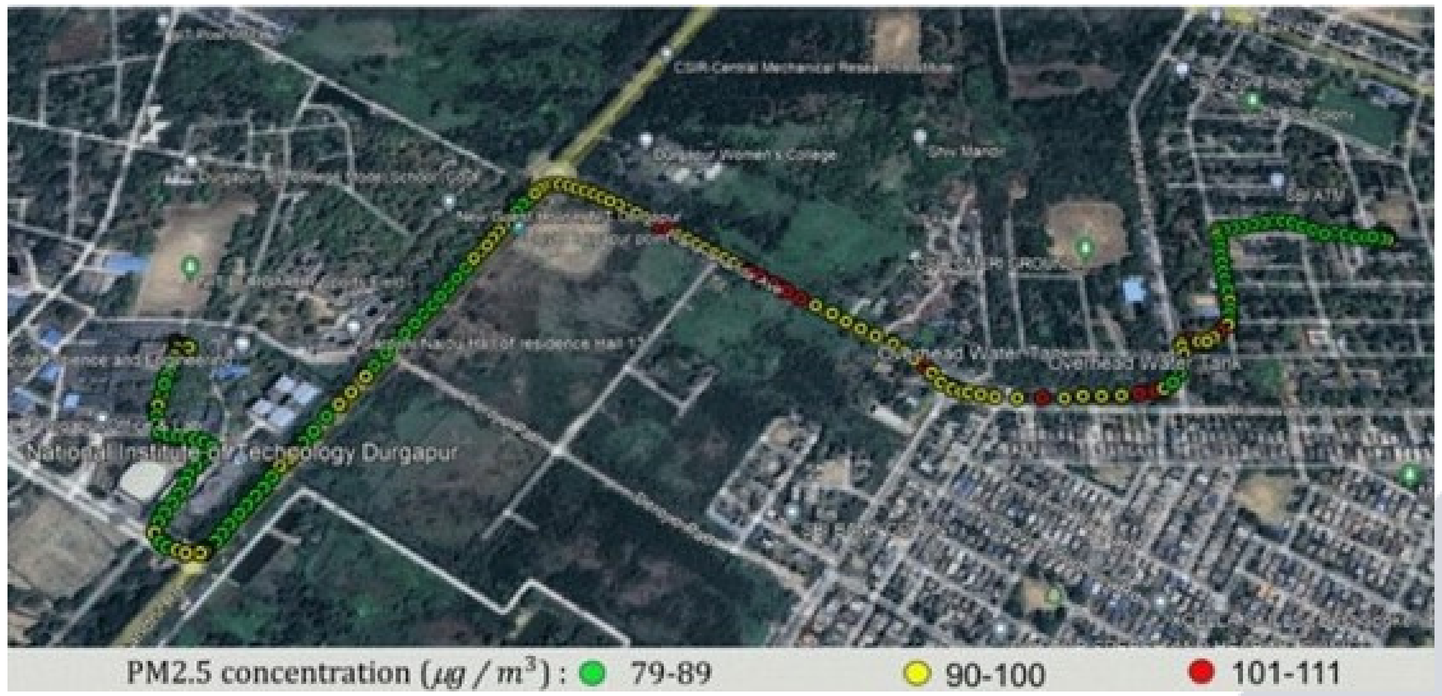
To develop a real-time pollution map that provides accurate and timely information on air quality levels at specific locations, enabling users to make informed decisions to minimize exposure to pollutants.

- Identify high or low gas concentration levels for CH_4, CO_2, SO_2, and NO_x gases and use PM2.5 values on the heatmap.
- Use color-coded schemes for air pollution levels affects users understand and interpret air quality data on the map.
- Analyze weekly exposure of pollutants and recommendations for travel times.

Methodology

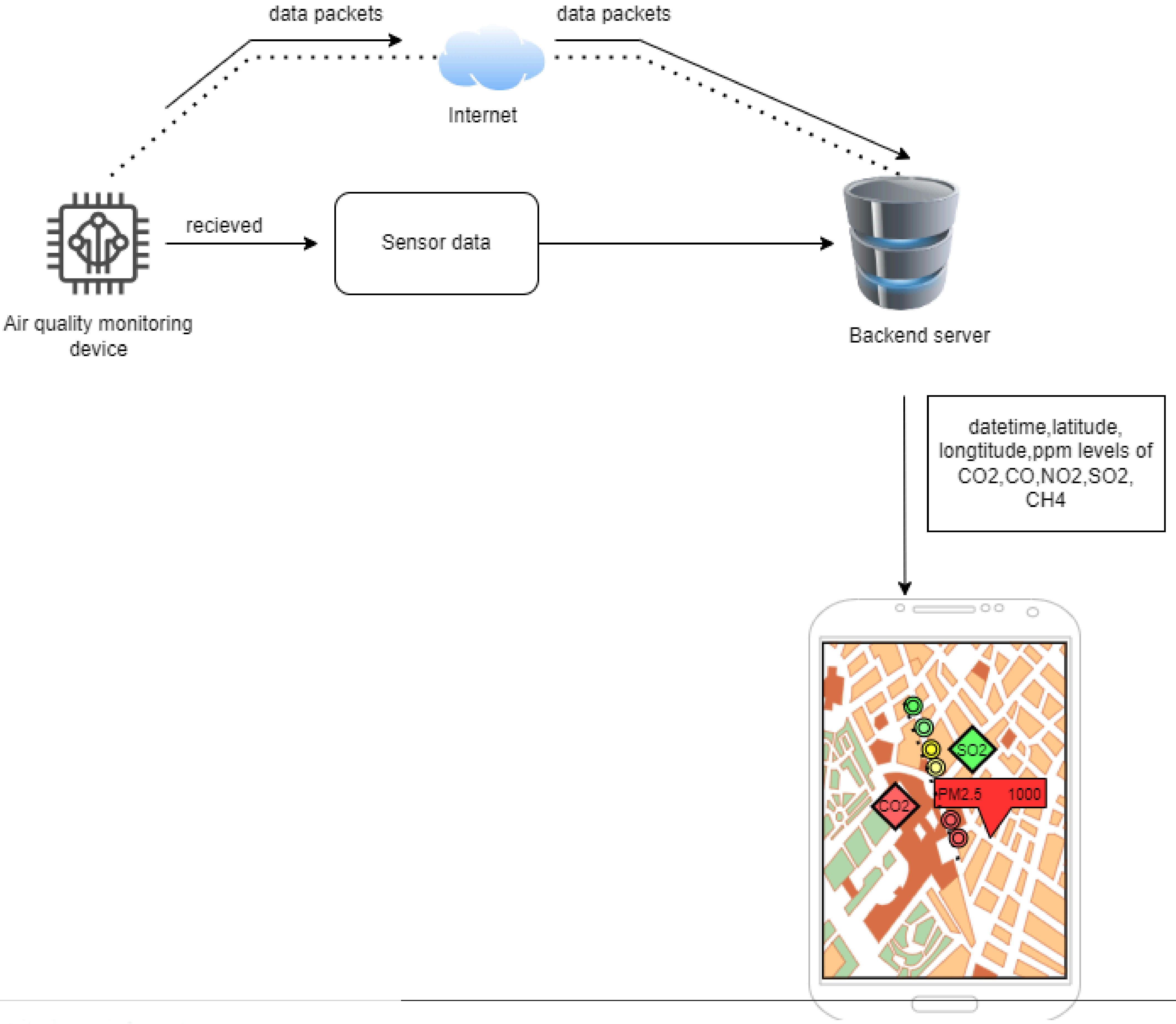
Existing Studies

Applications	Research A	Research B	Research C	Proposed System
Color coding schemes to identify different levels of air pollution.	✓	✓	✓	✓
Integration of gas concentration values and PM2.5 values into dynamic heat maps.	✗	✓	✗	✓
Analyze weekly exposure of pollutants and recommendations for travel times. Analyze weekly exposure of pollutants and recommendations for travel times.	✗	✗	✗	✓
Mobile Application	✓	✗	✗	✓



Proposed System

System Diagram



Used Techniques and Technologies

Technologies and Libries

- React Native
- MongoDB
- Google Map
- React Marker
- Vs code

Techniques

- Visualization Tools
- Geographic Information Systems (GIS)
- Sensor Technology
- Internet of Things(IOT) Monitoring System

Completion of the project
Methodology

Get realtime data through API

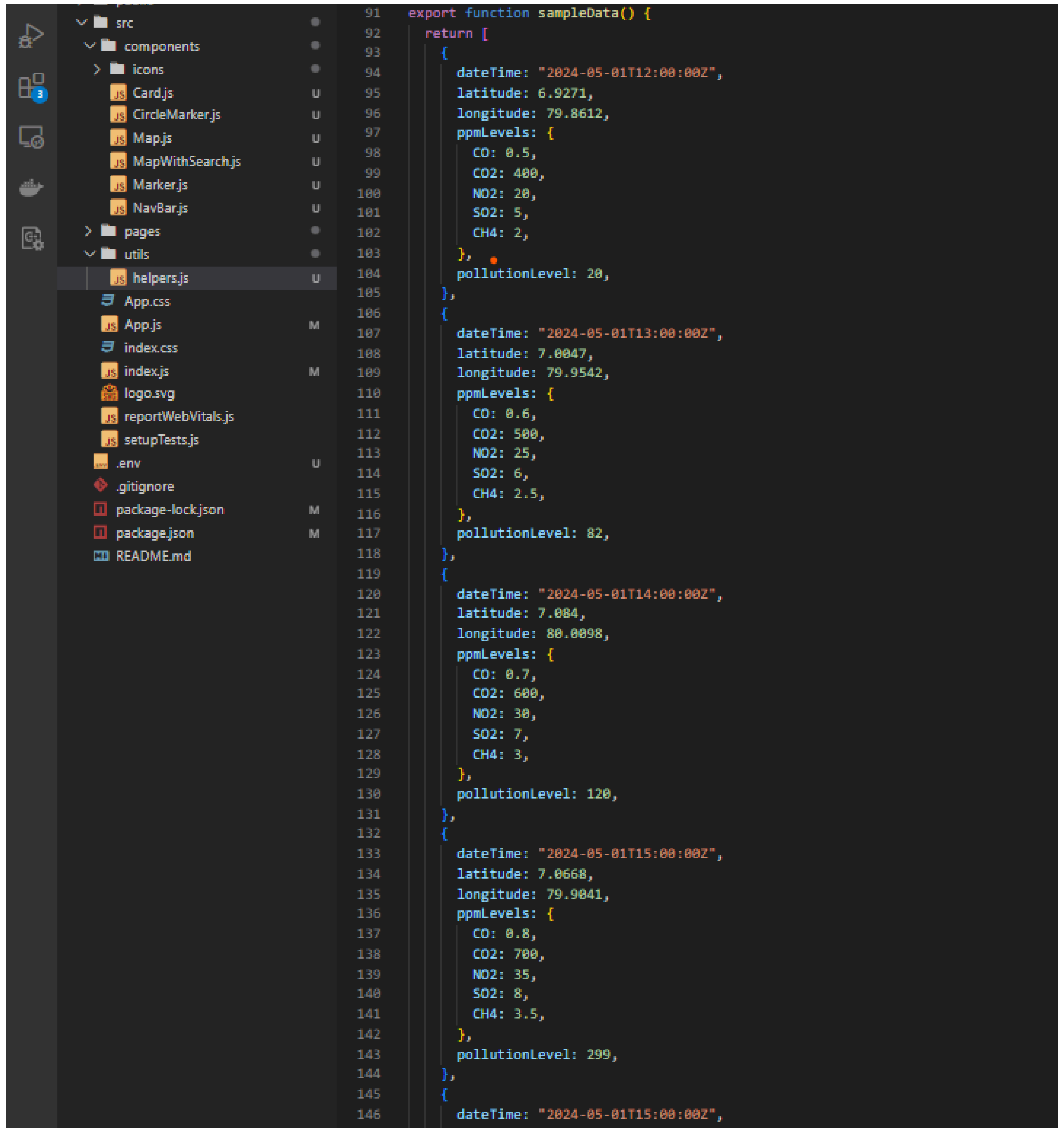
Data Visualization and mapping in mobile application

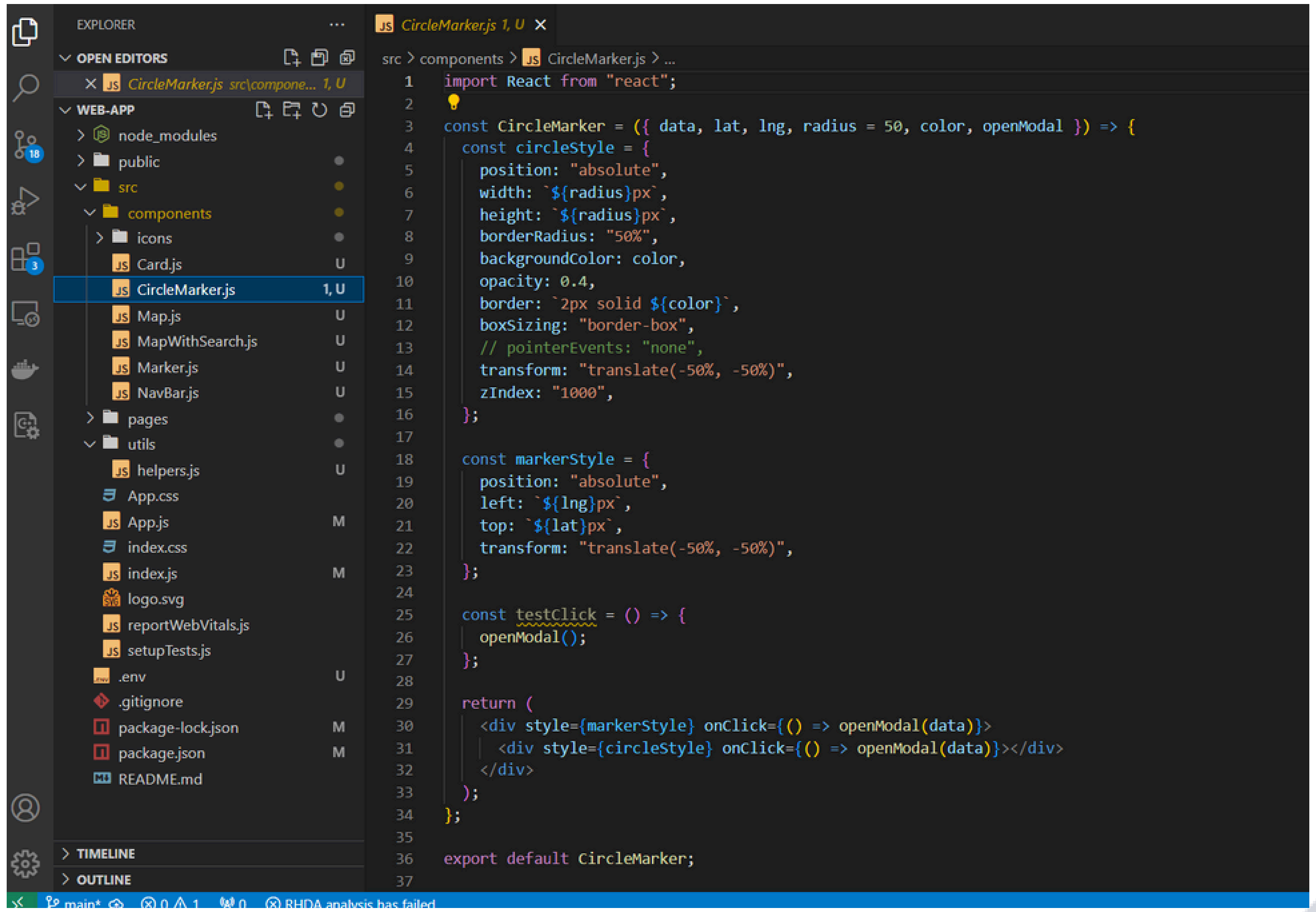
Map integration with Google Maps API

Pollution Level Classification

Implementation of collecting data points

Evidence of Completion



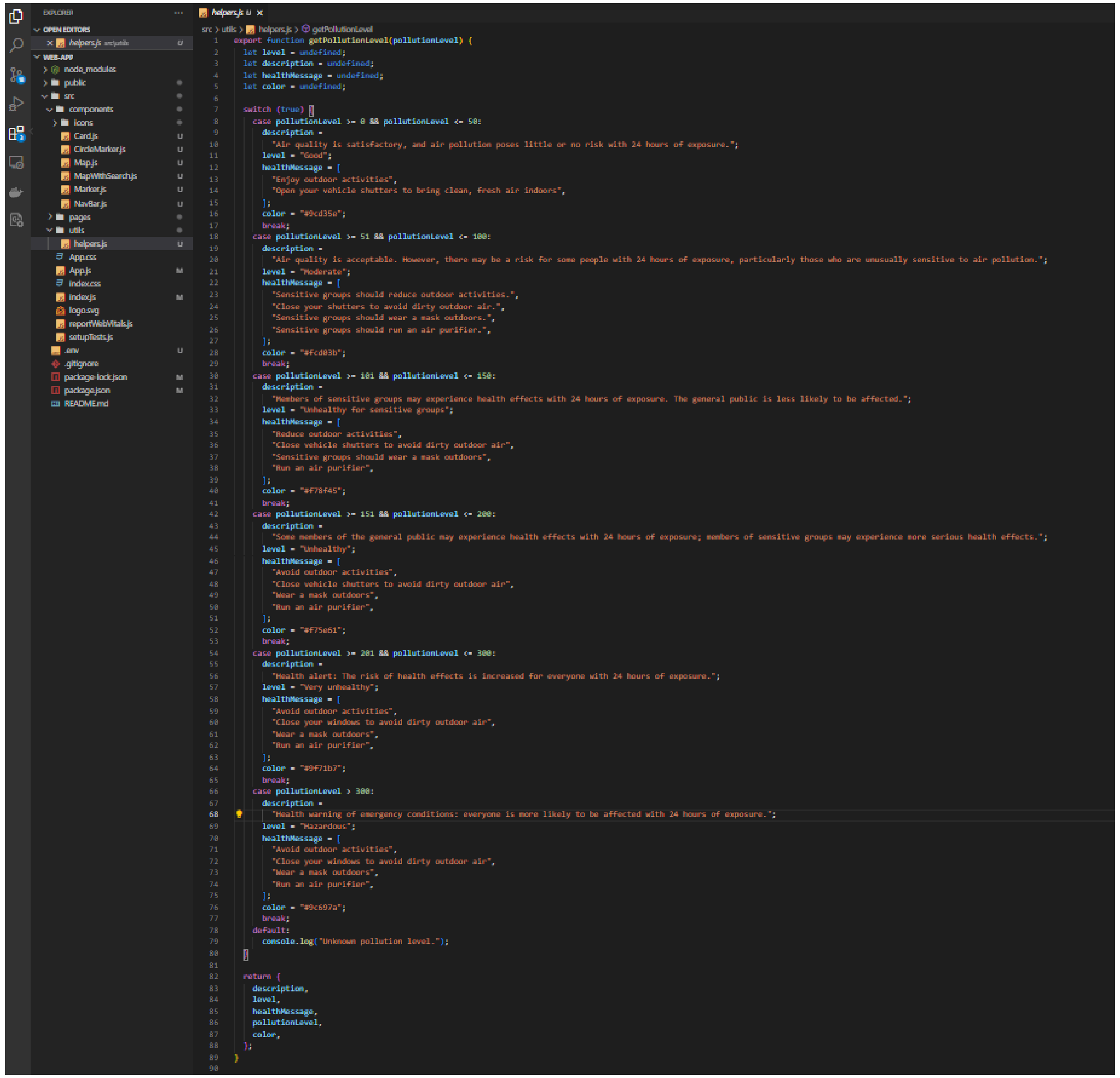


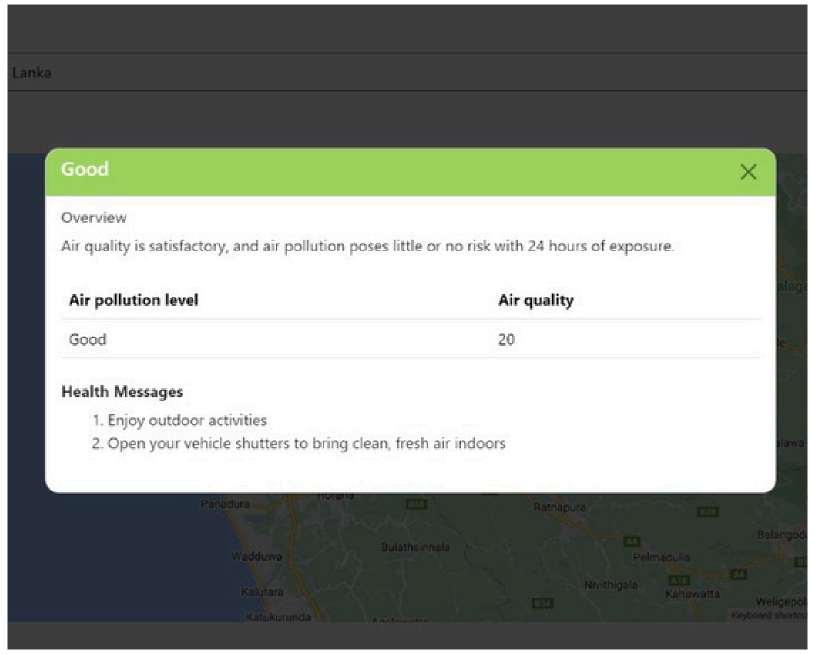
Implementation of collecting data points (dateTIme,latitude, longitude,gas concentration levels of CO,CO2,NO2,SO2,CH4 and pollution levels)

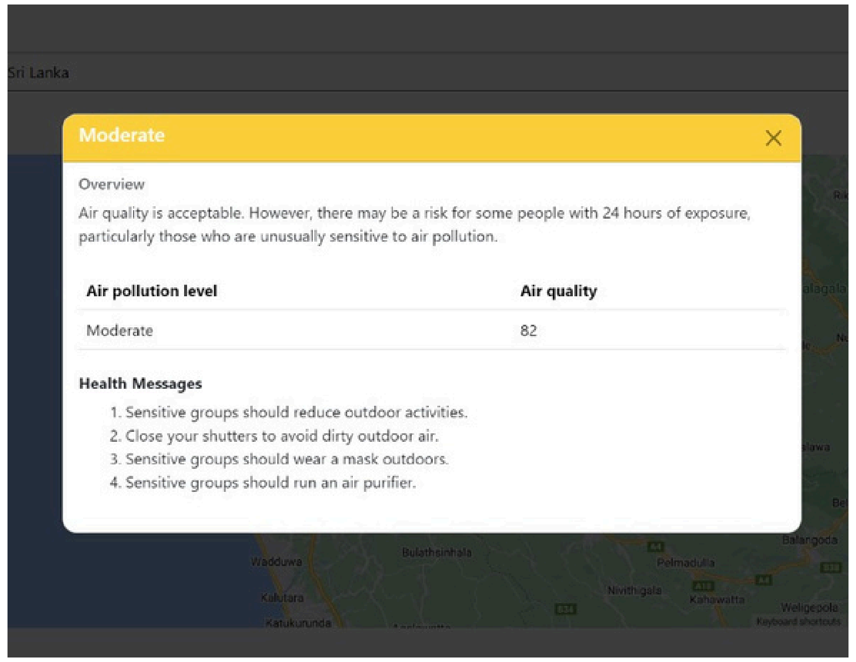
Implementation of Circle markers for identify data points using different color schemes

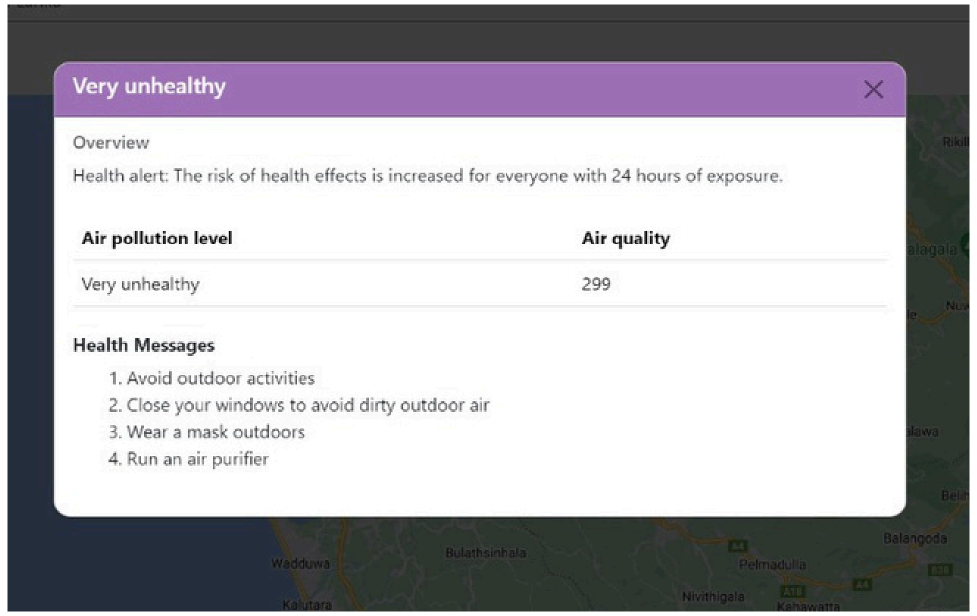
Pollution Levels Classification

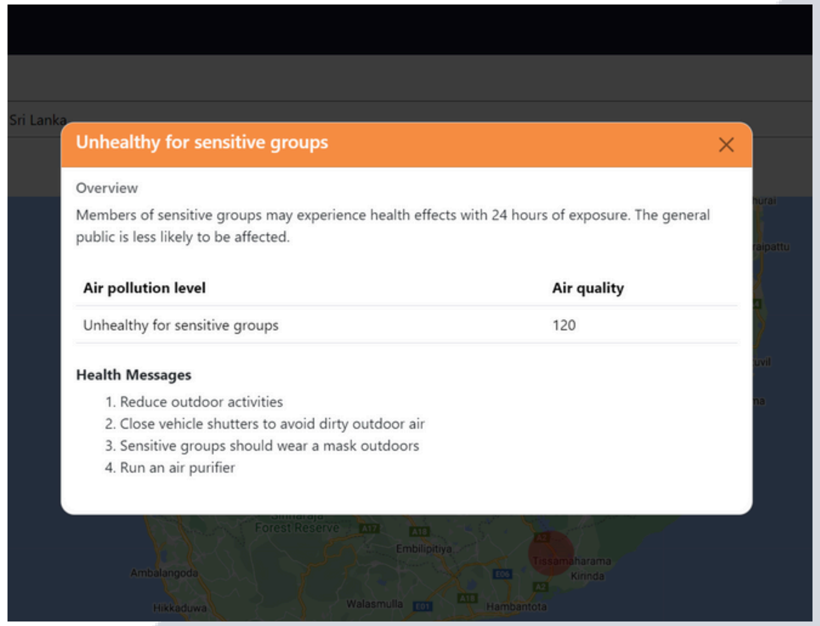
Evidence of Completion











Impelementation of pollution levels (good,moderate,unhealthy,very unhealthy)

Data Visualization and Mapping

Evidence of Completion

Display the Results in Application

Functional, Non-Functional and Personnel Requirements

Functional Requirements

- Classify pollution levels into categories (e.g.good, moderate, unhealthy) Implemented color-coded schemes or other visual cues to represent different pollution levels.
- View detailed gas concentration data for the selected area through different icons
- Analyze weekly exposure of pollutants and recommendations for travel times.
- Collecting high and low gas concentration levels for CH_4, CO_2, SO_2, and NO_x gases and use numerical labels for PM2.5 values on the heatmap

Non-Functional Requirements

- Interfaces should be User-friendly
- Application should properly work for cross platform
- Application should be reliable
- Higher accuracy of results
- Results should be more efficient
- Application should be able to give fast results

Personnel Requirements

Resources and Dataset in Colombo

- Central Environment Authority (CEA)
- The National Building Research Organization (NBRO).

Completion and Future works

50% Completion of the component

- Use color-coded schemes for air pollution levels on the map.(sub objective -2)
- Map integration with Google Maps API
- Data Visualization and Mapping in mobile application

50% Future Implementations

- Display gas concentration levels for CH_4, CO_2, SO_2, and NO_x gases and PM2.5 values on the heatmap
- Analyze weekly exposure of pollutants and Recommendations for travel times.
- Show suggestions within travelling these times user can travel with less effect of air pollution
- Connect with Server backend to integrate with Realtime pollution levels and data

References

[1]Castell, N., Dauge, F.R., Schneider, P., Vogt, M., Lerner, U., Fishbain, B., Broday, D., and Bartonova, A. (2017). "Can commercial low-cost sensor platforms contribute to air quality monitoring and exposure estimates?" Environment International, 99, 293-302.

[2]Zhang, W., Vazquez-Canteli, J.R., and Sadiq, R. (2016).” A real-time air quality routing algorithm: Reducing human exposure to outdoor air pollution. “69, 281-300.

[3]Tsai, J.H., Lv, N., and Wang, Y. (2016). “A heatmap-based visualization framework for air pollution monitoring data. Journal of Ambient Intelligence and Humanized Computing,” 7(4), 499-512.

IT21057588 | Pasan M.G.R

Specialization : Information Technoogy

Component 4 : Route Generation Based on pollution

Background

.

- Proposing the least air pollution shortest route when there are several routes to go between two locations within the Colombo district.

- A new algorithm has been developed by combining Dijkstra's algorithm and the dynamic mapping process to suggest the best way.

- Finally, the suggested shortest path and pollution levels for the route are sent to the Mobile application to display it on the map.

Research Questions

How data service for accessing and processing road network information.

How is the shortest route identified?

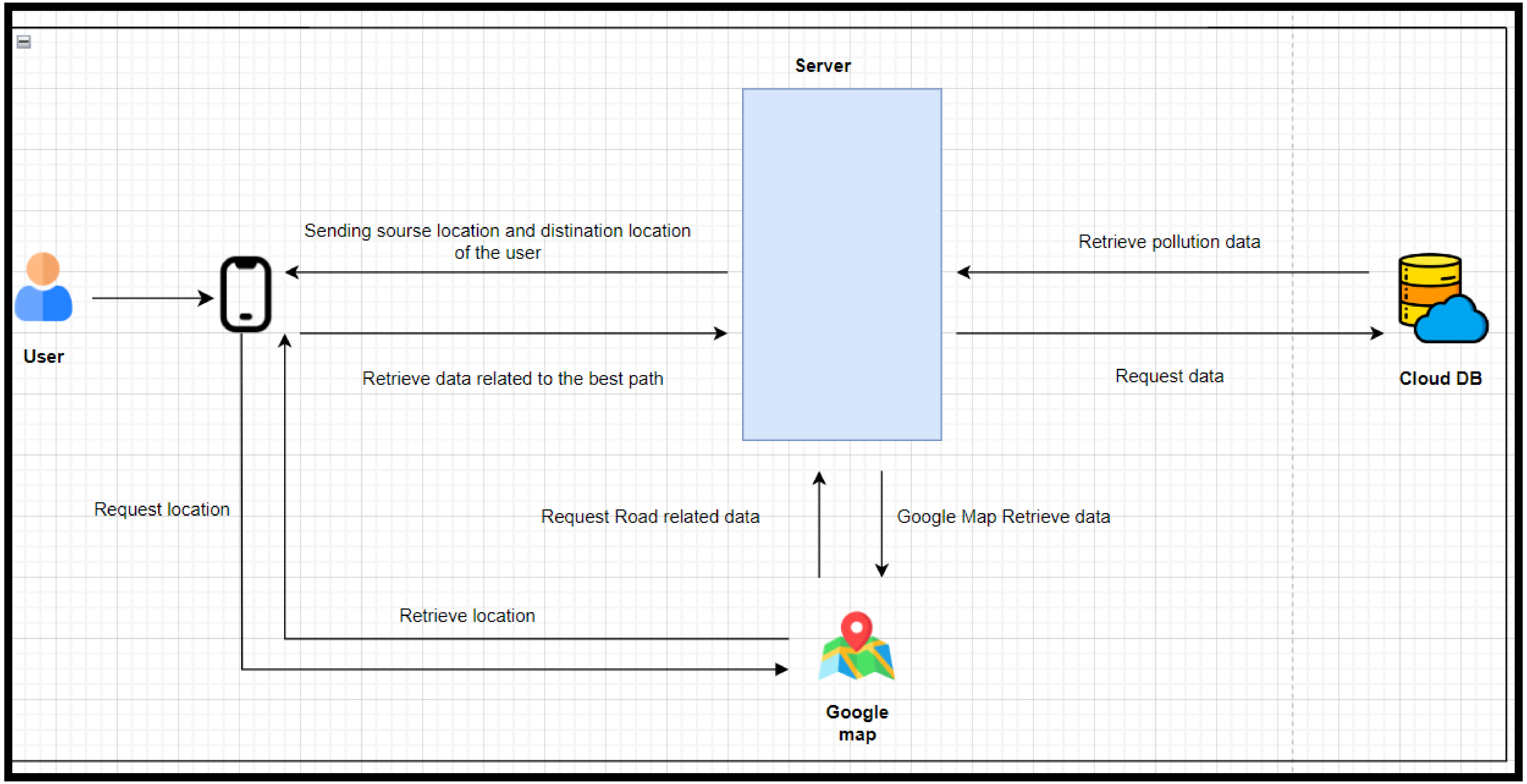
How is the Pollution Level Considered in the Route Map

Specific and Sub Objective

Route Generation Based on Pollution

- Identification of Polluted Route Areas
- Writing algorithm to calculate the near distance level low polluted route
- Finding best route with low pollution level

System Diagram



Used Techniques and Technologies

Technologies

- DataCollection
- Algorithm tuning

Algorithm

- Dijkstra's Algorithm

Techniques

- React Native
- Python
- Fast Server
- Node Server
- Jupyter Notebook
- Google Colab
- VS code

Methodology
Evidence of Completion

- **Data collection**
- **Host the finalized model in the fast server**
- **Model Evaluation**
- **Displayed the results in mobile application**
- **Model Testing**

Path Visualization

```python
[ ]  import heapq
     import networkx as nx
     import matplotlib.pyplot as plt

     def create_graph():
         graph = {
             'A': [('B', 5), ('C', 1)],
             'B': [('A', 5), ('C', 2), ('D', 1)],
             'C': [('A', 1), ('B', 2), ('D', 4), ('E', 8)],
             'D': [('B', 1), ('C', 4), ('E', 3), ('F', 6)],
             'E': [('C', 8), ('D', 3)],
             'F': [('D', 6)]
         }
         return graph
```

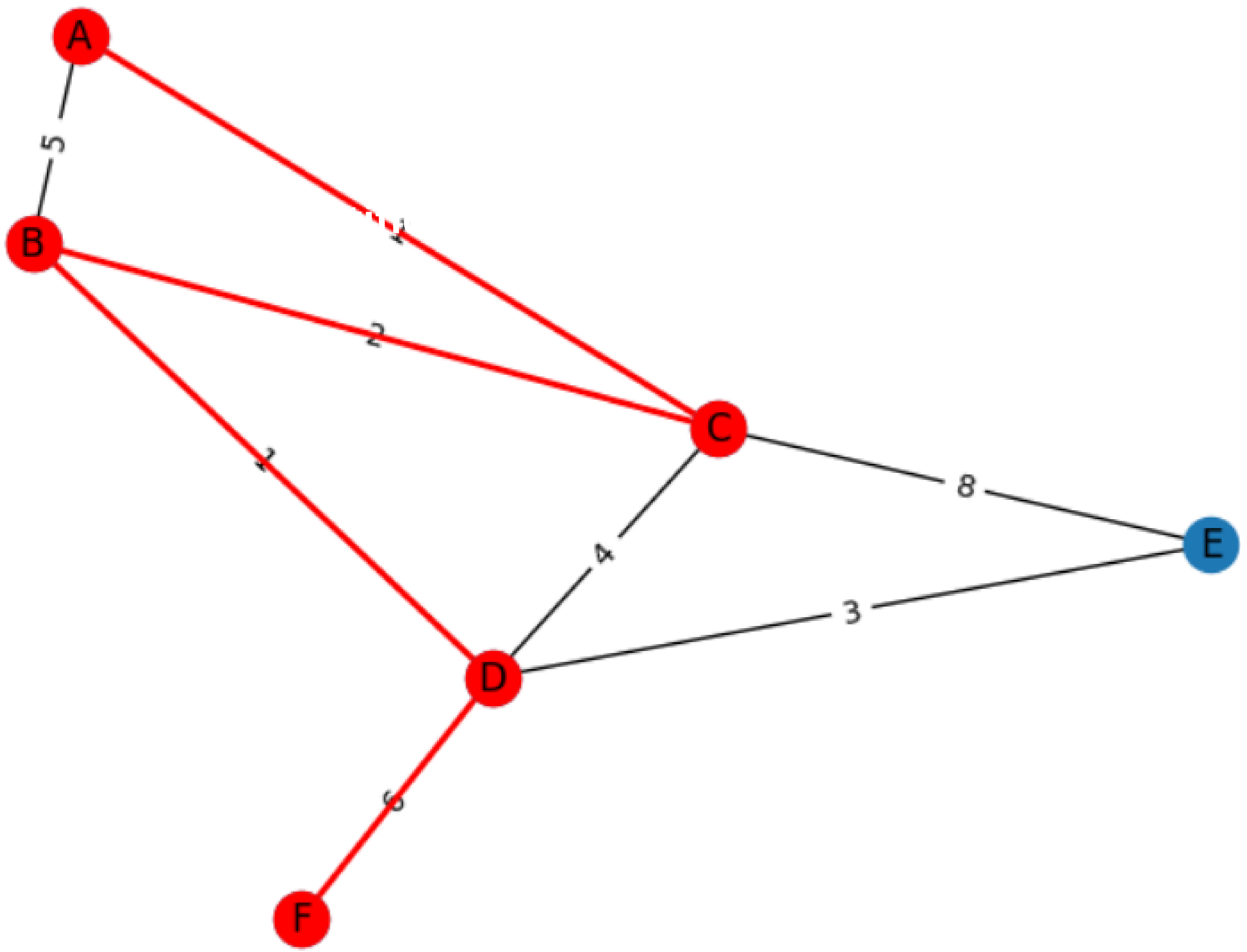
```python
def shortest_path(previous, start, end):
    path = []
    node = end
    while node is not None:
        path.append(node)
        node = previous[node]
    path.reverse()
    return path

def visualize_graph(graph, path=None):
    G = nx.Graph()

    for node in graph:
        G.add_node(node)

    for node, neighbors in graph.items():
        for neighbor, weight in neighbors:
            G.add_edge(node, neighbor, weight=weight)

    pos = nx.spring_layout(G)

    nx.draw(G, pos, with_labels=True)
    edge_labels = nx.get_edge_attributes(G, 'weight')
    nx.draw_networkx_edge_labels(G, pos, edge_labels=edge_labels)

    if path:
        path_edges = list(zip(path, path[1:]))
        nx.draw_networkx_nodes(G, pos, nodelist=path, node_color='r')
        nx.draw_networkx_edges(G, pos, edgelist=path_edges, edge_color='r', width=2)

    plt.axis('off')
    plt.show()
```

```python
[ ]  def dijkstra(graph, start, end):
         distances = {node: float('infinity') for node in graph}
         distances[start] = 0
         previous = {node: None for node in graph}

         pq = [(0, start)]

         while pq:
             current_dist, current_node = heapq.heappop(pq)

             if current_dist > distances[current_node]:
                 continue

             if current_node == end:
                 break
```

```python
if __name__ == "__main__":
    graph = create_graph()
    start = 'A'
    end = 'F'
    distances, previous = dijkstra(graph, start, end)
    path = shortest_path(previous, start, end)
    visualize_graph(graph, path)
```

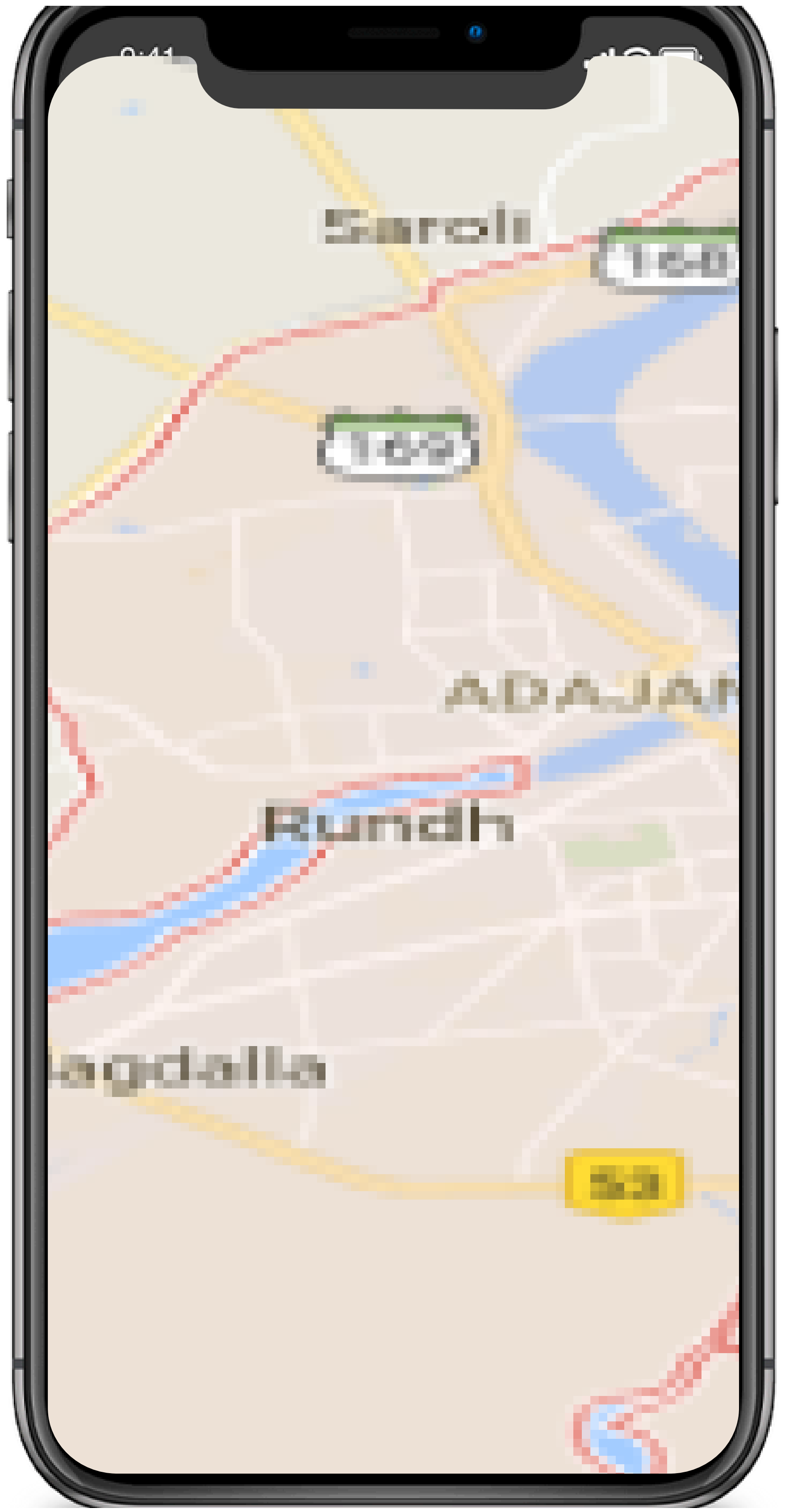
02

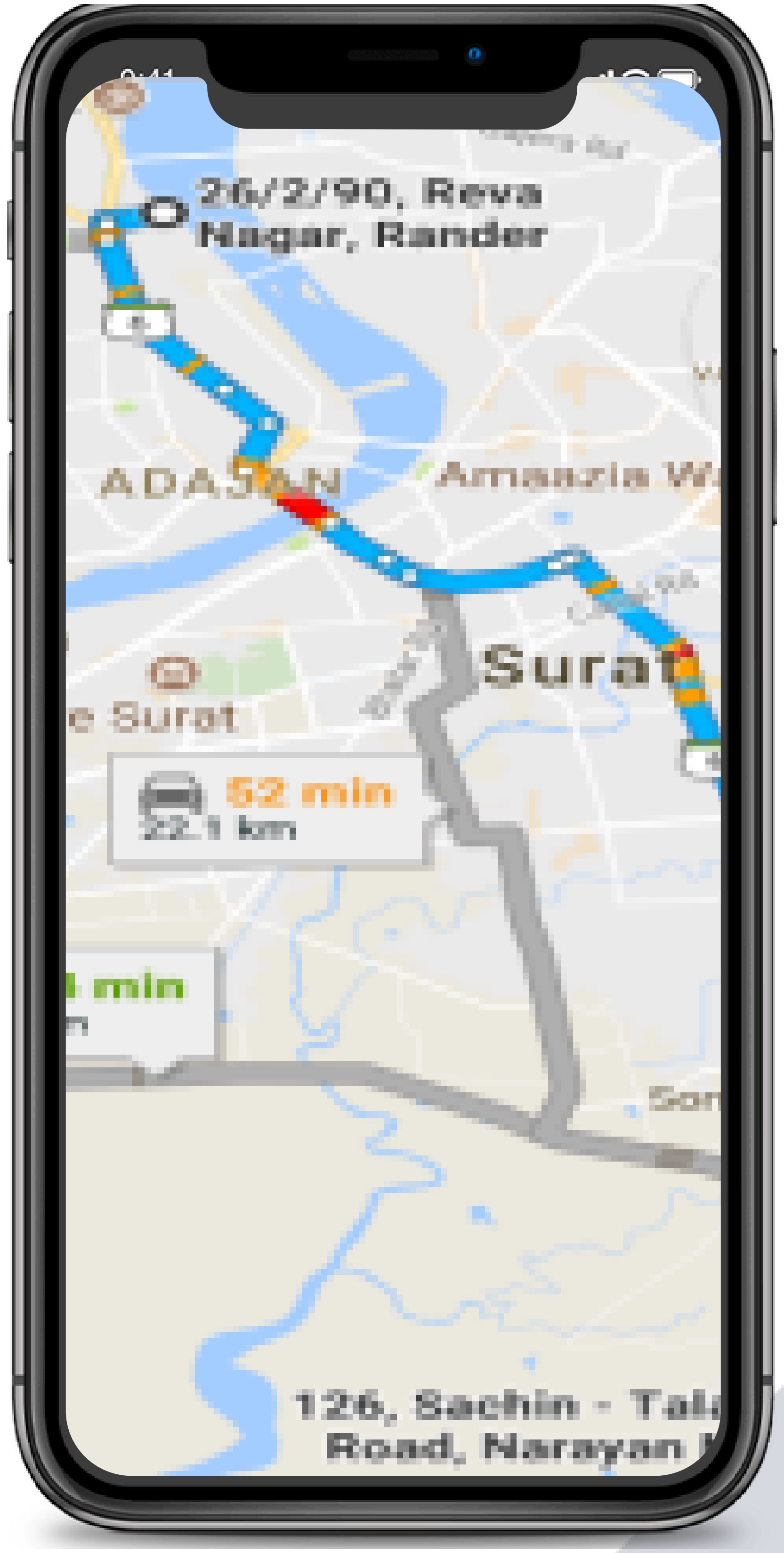
Output Graph



02

Display the result in Application





System, Personnel, and Software Specification Requirements

Functional Requirements

- When there are multiple routes between any two locations, display the non-polluted route on the map.
- Algorithms should be used to select the path with the least pollution level.
- The pollutant concentration data should be displayed above the minimum path of the Pottution level .

Non-Functional Requirements

- Interfaces should be User-friendly
- Should properly work for android and IOS devices
- The application should be reliable
- Higher accuracy of results
- Results should be more efficient

Personnel Requirements

- Central Environment Authority(CEA)
- The National Buildng Research Organization(NBRO)

50% Completion of the components

- identification of low polluted route.
- path visulization using dijkarst Algoridum.

50% Future Implementations

- Finish the Backend development.
- Display route and ppm values in frontend.

References

- Wang, Z., Novack, T., Yan, Y. and Zipf, A., 2020. Quiet route planning for pedestrians in traffic noise polluted environments. IEEE Transactions on Intelligent Transportation Systems, 22(12), pp.7573-7584.
- Yang, J., Cai, B., Li, X. and Ge, R., 2023, May. Optimal path planning for electric vehicle travel time based on Dijkstra. In 2023 35th Chinese Control and Decision Conference (CCDC) (pp. 721-726). IEEE.
- Zheng, Z., Yao, S., Li, G., Han, L. and Wang, Z., 2023. Pareto Improver: Learning Improvement Heuristics for Multi-Objective Route Planning. IEEE Transactions on Intelligent Transportation Systems.
- Jayalath, K.G., Deeyamulla, M.P. and de Silva, R.C.L., 2023. An Assessment of Transboundary Pollution from Colombo to Kandy on the Atmospheric Deposition of Heavy Metals Using Moss (Hyophila involuta). Sri Lankan Journal of Applied Sciences, 2(01), pp.39-43.

02

Thank You !