AIR PROTOCOL

"The winds come and go"

TEAM NAME - BRUTE FORCE

General Description

Air Patrol is an android app. In the app user will get all the weather information based on his current information and as well as an air quality prediction of his area.
The ML model deployed int the app will give air quality prediction based on several parameters like AQI level, weather, temperature and pollution indexes of the city ,country as well.
The ML model will be trained on a scaled multiple feature dataset.
The app will the interactive notifying the user when to wear mask based on the air quality of the region he/she is in.

Project Specification

Real time weather forecasting.

- Platform: Android
- IDE: Android Studio coded in Java
- Takes user's geolocation as input to provide weather forecast
- Hyper-Local Forecast: App will predict air quality, rain,storm and weather changes with a per minute accuracy based on the user current location.
- Displays detailed weather of the current time.
- Weather visualization
- Can also take custom location as input to provide weather details for that location (GoogleManualSearch API).
- REST API used to fetch data from www.openweathermap.org

Scenario Analysis

Screen and Interaction Analysis:
The users will use this mobile app on android smart phones and tablets. All the information of this mobile app will be displayed full screen. Basically, the interactions include touch and click and scroll. For example, when users would like to view weather information, they click the icon to open this app; when they want to add a new city, they click the add icon and type in the city that they want.

Usage Analysis
Users can use this mobile app **whenever they want, every day at home, on their way to travel**, and other situations as long as they want to know weather information.User must

be **connected to the internet**,being offline is not an option.

Environment Analysis:
Apart from weather forecasting it also predicts air quality of the user's location based on a ML Model deployed with the app.It notifies the user when the air quality is bad and indeeed they shoukd wear a mask.

Deliverables

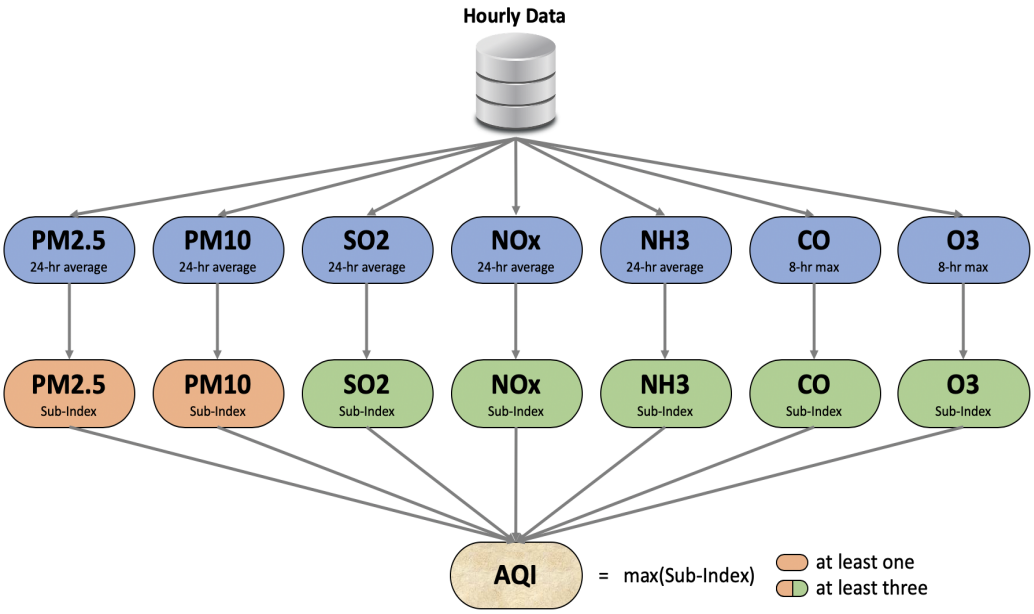
- A properly working and executable .apk file that will run in any android device.
- The app takes only takes 50-60 Kb space on device.
- A documentation that will provide details about the requirements, specifications and other information.

Machine Learning model

Dataset

The dataset used is hourly air quality data (2015 - 2020) from various measuring stations across India.

AQI Calculation



- The AQI calculation uses 7 measures: **PM2.5, PM10, SO2, NOx, NH3, CO and O3**.
- For **PM2.5, PM10, SO2, NOx and NH3** the average value in last 24-hrs is used with the condition of having at least 16 values.
- For **CO and O3** the maximum value in last 8-hrs is used.
- Each measure is converted into a Sub-Index based on pre-defined groups.
- Sometimes measures are not available due to lack of measuring or lack of required data points.
- Final AQI is the maximum Sub-Index with the condition that at least one of PM2.5 and PM10

should be available and at least three out of the seven should be available.

The final AQI is the maximum Sub-Index among the available sub-indices with the condition that at least one of PM2.5 and PM10 should be available and at least three out of the seven should be available.

After data analytics, visualization and generalization of the training data , a model was bulit predicting AQI values with a high accuracy based on the inputs of indivual pollutantant indexes.

Good (0–50)	Minimal Impact	**Poor (201–300)**	Breathing discomfort to people on prolonged exposure
Satisfactory (51–100)	Minor breathing discomfort to sensitive people	**Very Poor (301–400)**	Respiratory illness to the people on prolonged exposure
Moderate (101–200)	Breathing discomfort to the people with lung, heart disease, children and older adults	**Severe (>401)**	Respiratory effects even on healthy people

Work Plan

Literature search and review:
We will study about Android programming, App development, Android Studio environment & features, network requests, use of APIsand JSON formatting the fetched data.

Analysis and modeling:
Based on our gathered knowledge base, we will analyze the processes and make a prototype of the application.

ML Model for Prediction:
The ML model deployed int the app gives air quality prediction based on several parameters like AQI level, weather, temperature and pollution indexes of the city ,country as well.
The ML model will be trained on a scaled multiple feature dataset.

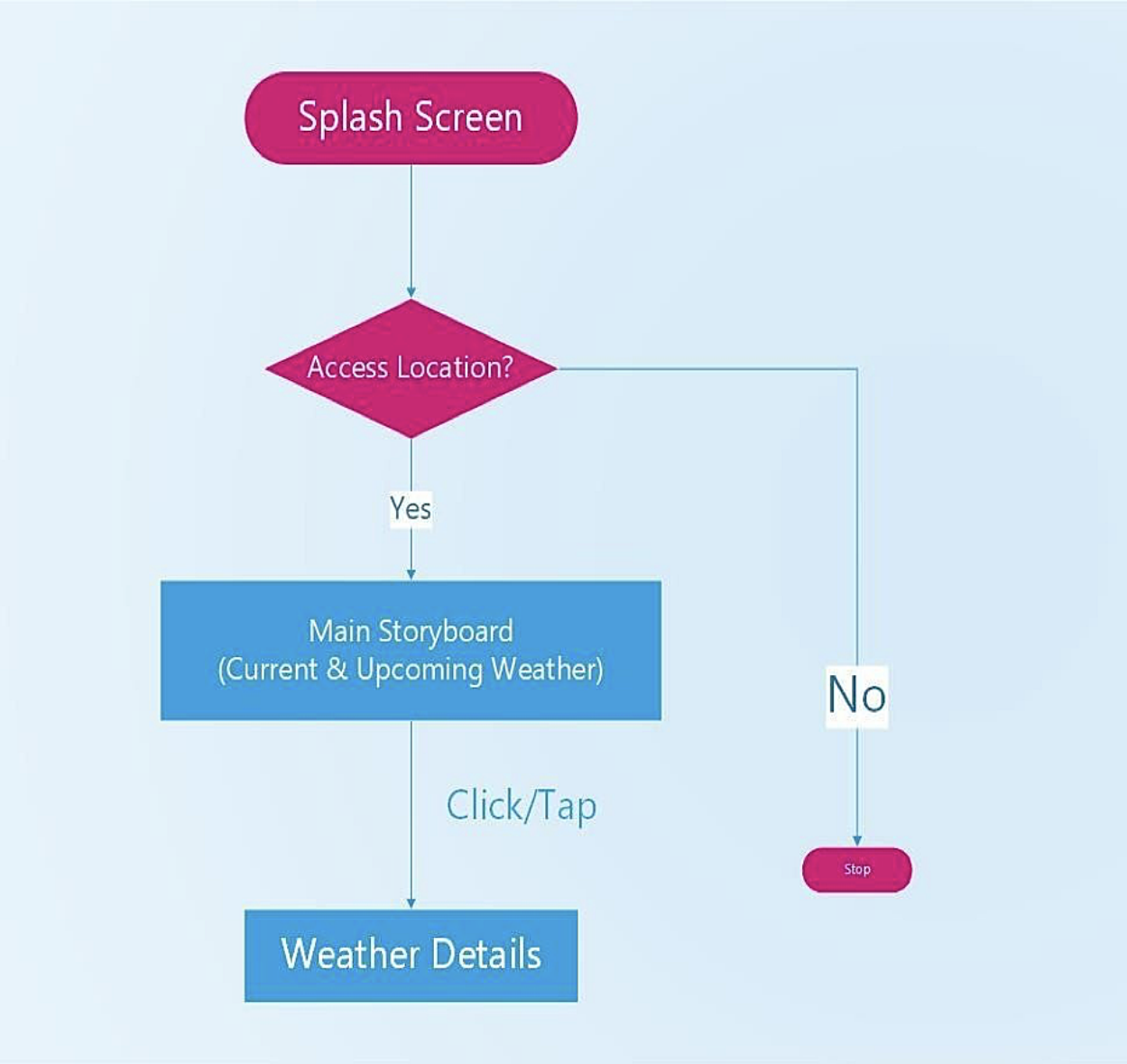
Navigation and UI design:
We will design application layout and application flow. Also, we will design the splash screen.

Implementation:
From prototype we will start to integrate the modules together and we will finish all the features enlisted.The program must be proper functioning and error free.

Testing and debugging:
Testing and debugging will be a challenging job for us, as we will let some people use the app and note suggestions from them.Also,we always need to emphasize on the comprehensiveness of our data and the magnificence of our UI.

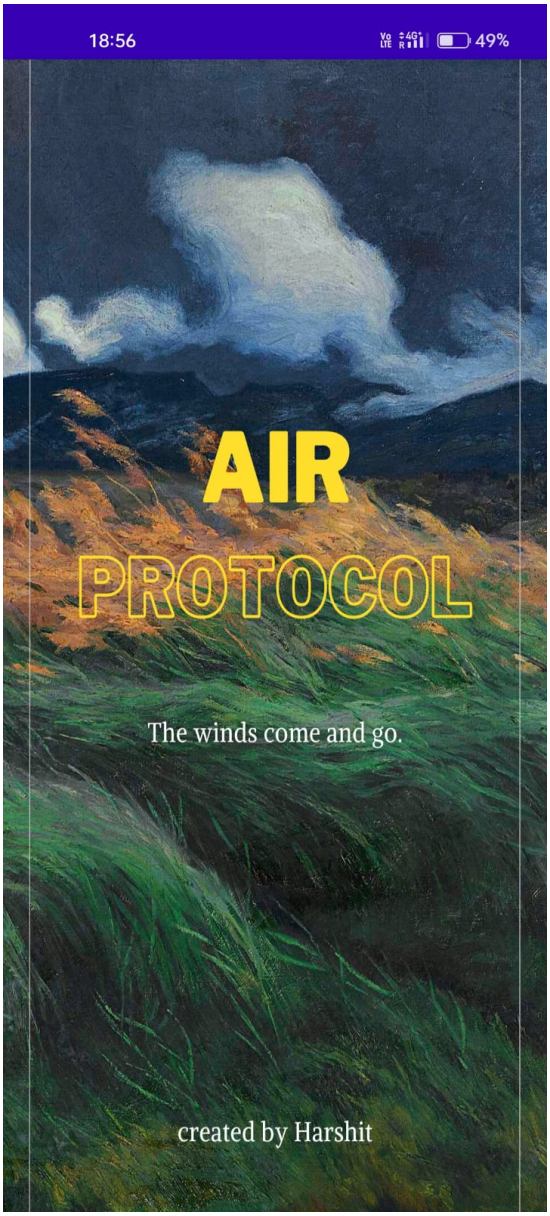
Work on final report and documentation: Worked on the final report and software documentations in the last week.Also kept track of our limitations and enlisted some unimplemented features that we intend to work on in the near future

UI Diagram:

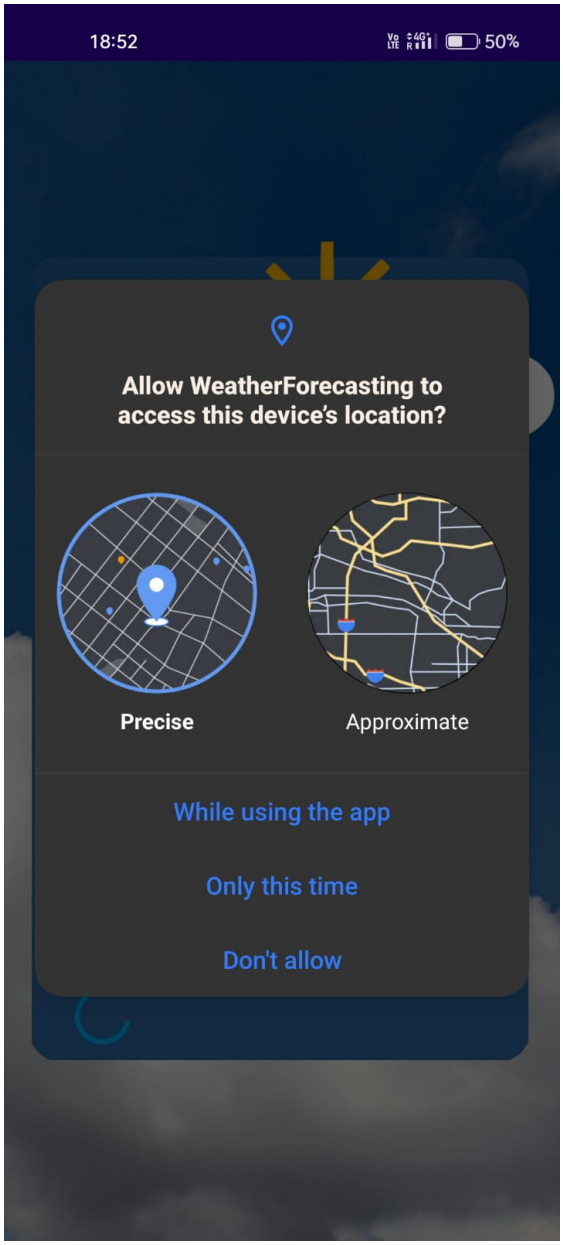


Screenshots of the App

Splash Page



Permissions



Main Page


18:51
50%
28°
Today
The weather is partly cloudy
AIR QUALITY
Poor(AQI>150)
Suggested to wear a mask
Gūduvāncheri, India

Required Links

Github Repository

https://github.com/smartinternz02/SBSPS-Challenge-9535-Real-Time-Air-Quality-Monitoring-Weather-Forecasting-System

References

Model

https://www.kaggle.com/rohanrao/calculating-aqi-air-quality-index

https://www.kaggle.com/datasets/rohanrao/air-quality-data-in-india

App

AccuWeather inc. developed the Accuweather app that provides the most comprehensive data comparatively

Haze developed by Robocat & Taptanium is the most simplistic and gorgeous app one will ever find.