AudiEnv Prototype

Team RA Cube:

Ankur Bhatia

Rupam Bhattacharya

Amit Kumar Mondal

Aakash Nayyar

PROJECT as part of AUDI APP CHALLENGE 2015, TECHNICAL UNIVERSITY OF MUNICH

31.03.2016

Table of Contents

Introduction

Motivation:

The Internet of Things is indeed the talk of the town these days. From the perspective of the automotive industries, it makes them more efficient by enabling us to control them and get their state from virtually anywhere. The biggest motivation for us was to be a part of this revolution in the automotive industry and go back with some learning that we can apply later on in our careers as well. As mentioned, being a very new topic, there was a lot of scope for research and an opportunity to come up with a prototype that has not been developed by many yet. It gives us immense pleasure and a sense of satisfaction, that we gave our 100% and developed something meaningful that can be used as a starting point to develop a prototype that can be used in the actual production environment.

General Description of the Application:

Our project was to build an android app to monitor the air pollution and weather conditions around the world by using sensors in cars which could be extended to public transports as well. Right now, it's possible to get air pollution and weather data from only the stationary points in the city. Air pollution data from mobile nodes has not been implemented in the industry yet.

We would like to use pollution sensors, gas sensors, flame sensors, temperature sensors and precipitation sensors in order to get the real time data at each node of a city.

The data we obtain from here can be used for further data mining applications like if there are two factories operating in a particular area, we would be able to use the data to see the pollution level and deduce whether they can be operated at different times. The challenges lie in processing huge parallel streams of big data and updating to a centralized database using algorithms to deduce which data should be updated.

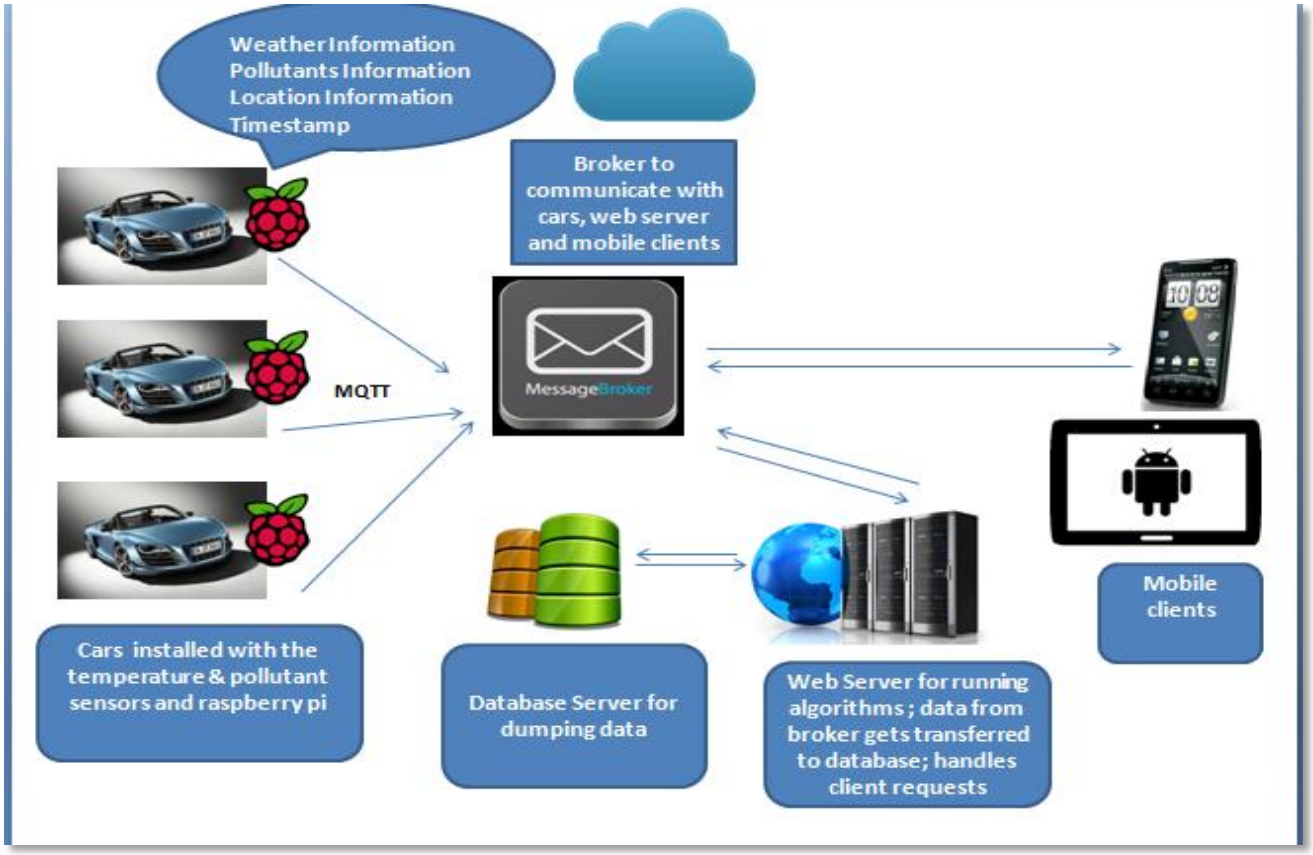
Scalable Architecture

Tools & Technologies:

Our project is an application of Internet of Things, Crowd Sourcing, Data Mining & Social Networking.The following tools, technologies and protocols were incorporated due to projects requirements as well as to make the prototype.

1. MQTT
2. Android Studio
3. Google Map Services
4. MP Android Chart Library
5. Facebook and Twitter Libraries

Overall Architecture:



MQTT:

The primary mechanism that devices and applications use to communicate with the IBM Internet of Things Foundation is MQTT; this is a protocol designed for the efficient exchange of real-time data with sensor and mobile devices.

MQTT runs over TCP/IP and, while it is possible to code directly to TCP/IP, you might prefer to use a library that handles the details of the MQTT protocol for you. You will find there's a wide range of MQTT client libraries available at mqtt.org, with the best place to start looking being the Eclipse Paho project. IBM contributes to the development and support of many of these libraries.

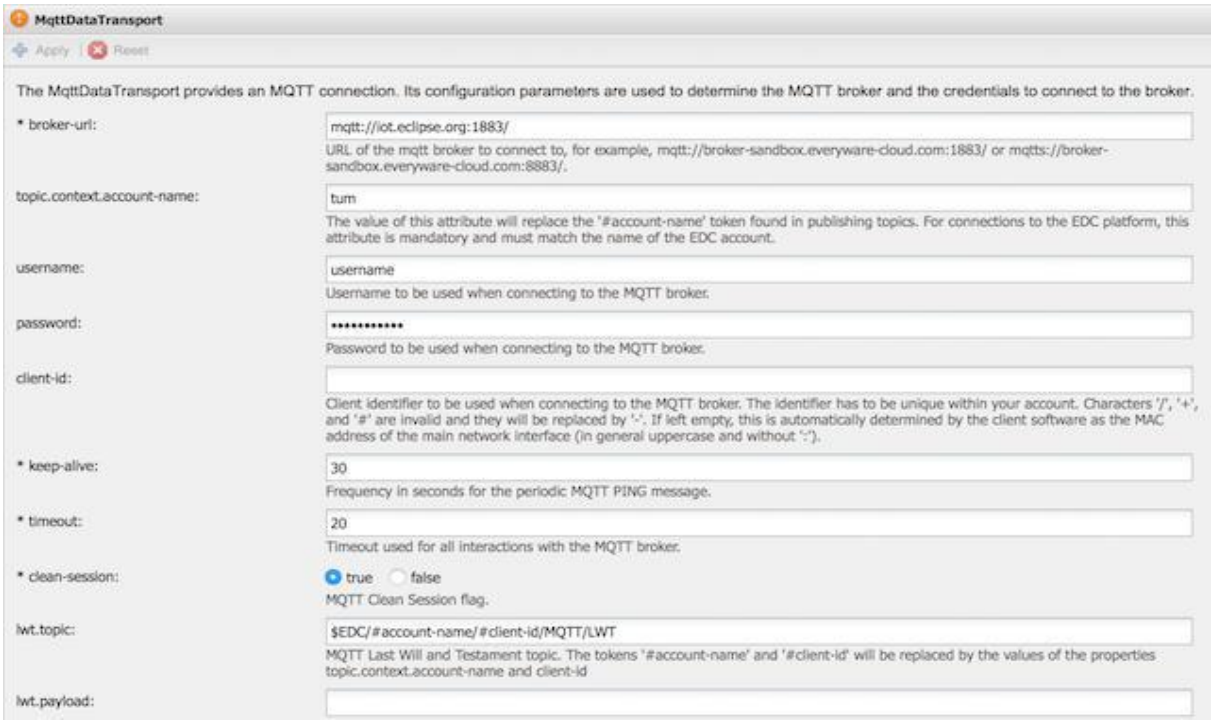
MQTT 3.1 is the version of the protocol that is in widest use today. Version 3.1.1 contains a number of minor enhancements, and has been ratified as an OASIS Standard.

One reason for using version 3.1.1 is that the maximum length of the MQTT Client Identifier (Client ID) is increased from the 23 character limit imposed by 3.1. The IoT service will often require longer Client ID's and will accept long Client ID's with either version of the protocol however some 3.1 client libraries check the Client ID and enforce the 23 characters limit.

Every registered organization has a unique endpoint which must be used when connecting MQTT clients for applications and devices in that organization.

The MQTT protocol provides three qualities of service for delivering messages between clients and servers: "at most once", "at least once" and "exactly once". Events and commands can be sent using any quality of service level; however you should carefully consider whether what the right level is for your needs. It is not a simple case that QoS2 is "better" than QoS0.

The following image shows the various options to configure MQTT in Gateway.



Web Server:

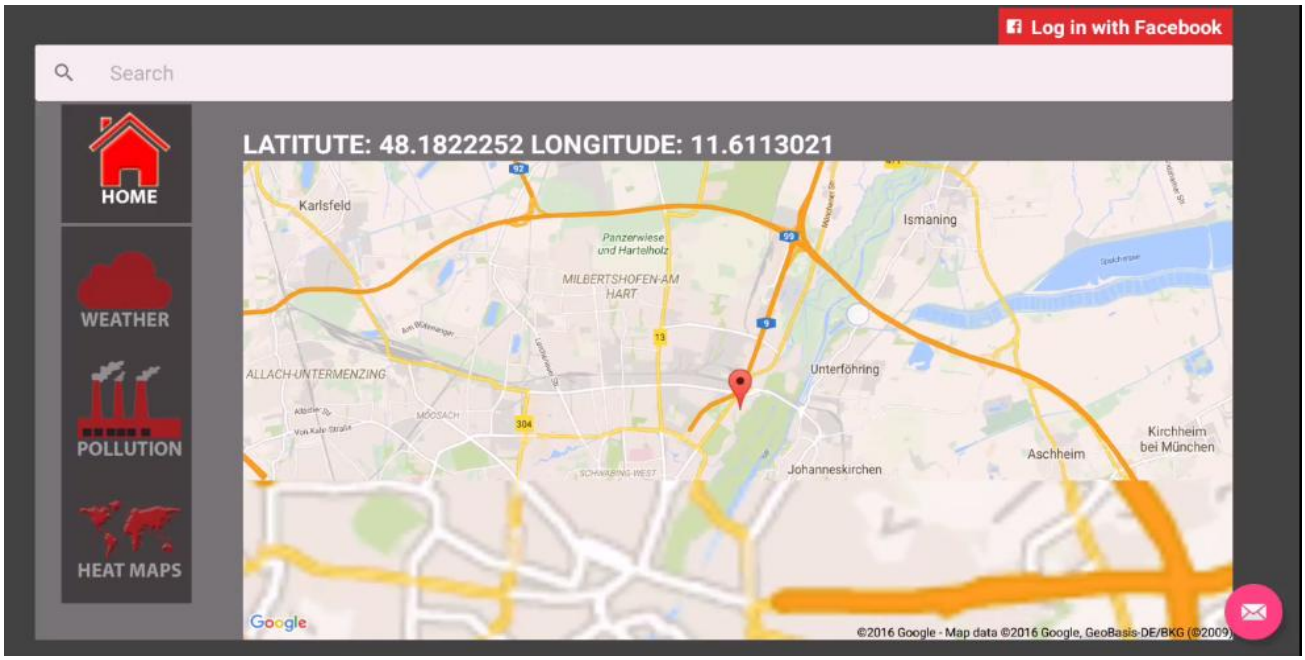
For now, the HTTP web server is hosted locally which handles request & response from the android client. The web server has to be ported to a cloud to handle multiple requests and response for better scalability & performance.

How to run the Android Application

Login Screen:

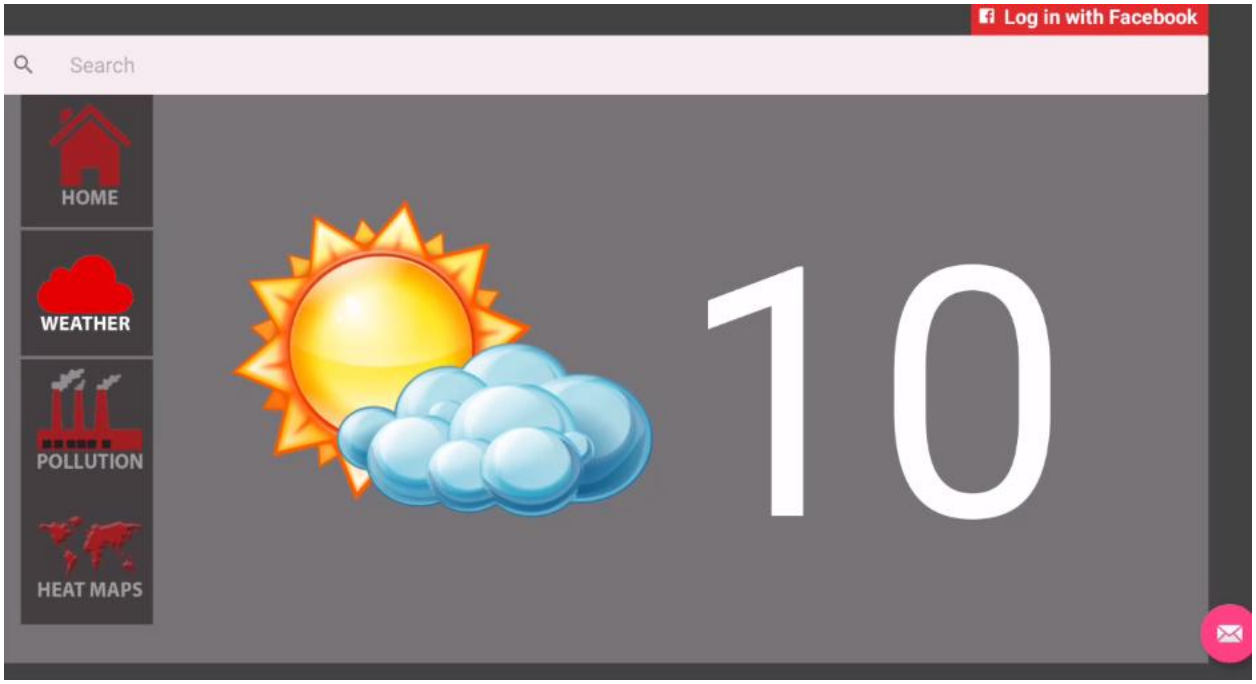
The user can login to the Weather and Pollution Monitor app by providing credentials (audi as email & audi as password). The user has also the option to login via Facebook.

User Home Screen:



After login, the user would be redirected to the home screen where the current location would be shown to the user with the map. The user has the option to browse to any of the pollutant or weather information screen or heat map screen. Further, the user can search for any location with the auto complete search feature.

Real time Weather Information Screen:



The user can go to weather screen to see the weather details for the current location. Weather information details like temperature, humidity, precipitation would be available to the user. The user can also search location and see the details for that particular location.

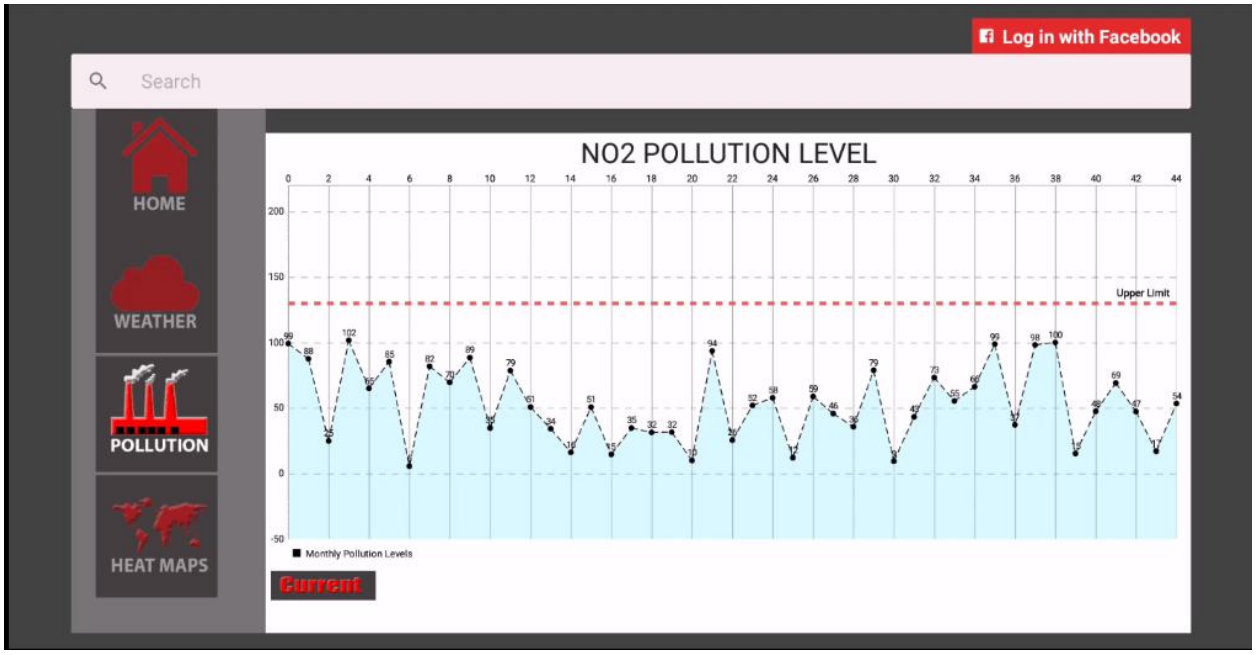
Real time Pollution Information Screen:

From the home screen, the user can go to pollution screen to see the pollution details for the current location. Pollution information details like sulfur dioxide levels, carbon monoxide levels, nitrogen oxide levels, lead, particulate matter levels would be shown to the user. The colors indicate the permissible limit of the pollutants with red being the most dangerous and green the safest.

The prototype offers recommendations to the user whether it is safe to go out or not or related recommendations based on the values of the pollutants exceed the permissible value or not.

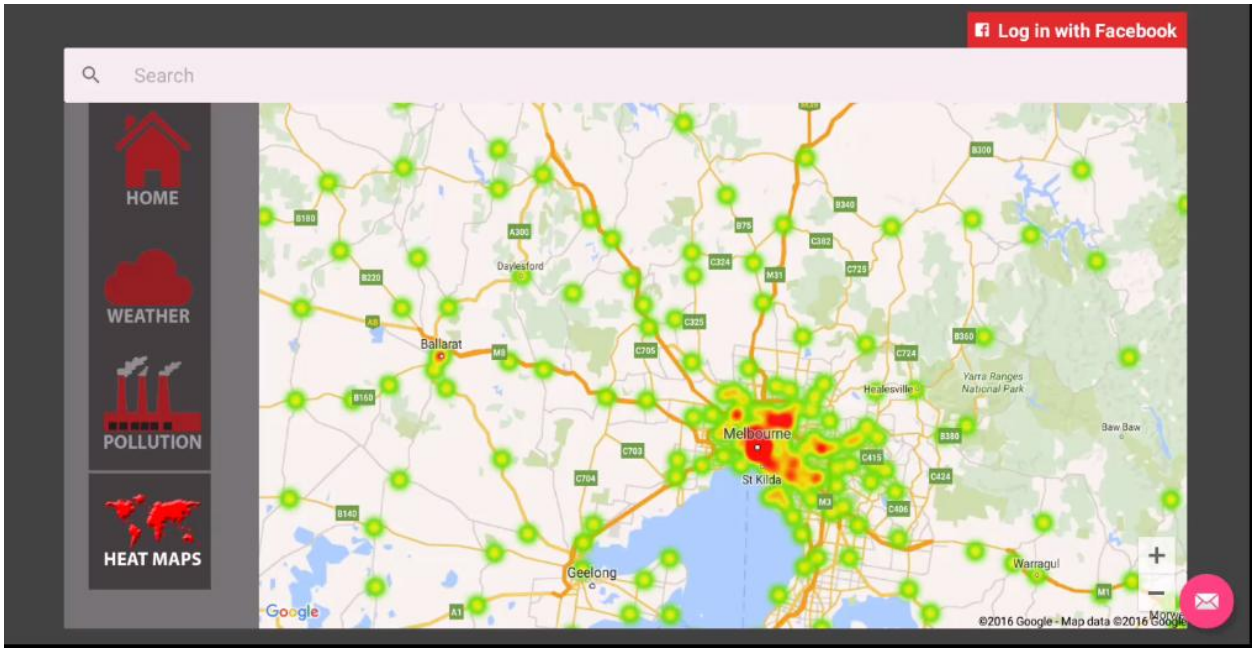
The user can check the historical pollution details by clicking on the individual pollution co-ordinates like O3, CO, SO2, etc.

Historical Pollution Information Screen for each pollutants:



From the pollution screen, the user can check the historical pollution details by clicking on the individual pollution co-ordinates like O3, CO, SO2, etc. The chart here indicates the historical values for NO2. The higher limit indicates the permissible limit for the pollutant for a particular city. The user has to click on “Current” button to go back to the pollution screen to see the real time values. From there onwards, he can again browse back to see historical values for other pollution co-ordinates like CO, O3, etc. He can also go back to Home Screen, Weather Screen or Heat Map Screen.

Pollution Heat Map:



From the home screen, the user can go to pollution heat map screen to see the places which are least polluted & the places which are mostly polluted. The red dots show the places which are mostly polluted and are unsafe where as the green dots show the places which are least polluted and comparatively safe. The user can go back to Home Screen or Weather Screen or Real time Pollution Information Screen.

Business Case for the Prototype

Navigation based on Pollution

Audi Env can offer the least polluted routes when a user searches for a location from place A to place B. It is particularly useful in developing economies where the pollution level is quite high.

City Planning based on Pollution & Weather Patterns

Audi Env can offer interesting patterns about pollution and weather; identifying zones in a city which are least and highly polluted. It is particularly important for city planning in the future context; where authorities like Government can take decisions in urban planning.

Verordnung zur Kennzeichnung der Kraftfahrzeuge mit geringem Beitrag zur Schadstoffbelastung

Audi Env can offer Audi drivers to have low emission number plates on their cars (http://www.dekra.de/feinstaub/). This can be done because Audi cars would have real time pollution data and thus Audi can take subsequent measures.

Identifying pollution & weather patterns by researchers; specifically by environmentalists

Audi Env can offer real time weather and pollution data for any location which is very important for researchers; specifically to environmentalists who can use this data to make ground breaking research when it comes to unfolding new patterns.

Performance monitoring of Audi Cars in different pollution & weather conditions

Audi Env can be used by Audi to monitor the performance of its cars by checking the performance of the car with respect to different pollution & weather conditions. The data obtained can be used for research by Audi as well.

Source Code

The repository of the prototype can be found at:

https://git.quartett-mobile.de/appchallenge/audi-appchallenge-muenchen/

Contact Persons

- Ankur Bhatia - bhatia.ankur8@gmail.com
- Rupam Bhattacharya – rupam.speaks@gmail.com
- Amit Kumar Mondal - admin@amitinside.com
- Aakash Nayyar - akash.nayyar@tum.de

Bibliography

- http://mqtt.org
- https://github.com/PhilJay/MPAndroidChart
- https://github.com/googlemaps/android-maps-utils
- https://de.wikipedia.org/wiki/Verordnung_zur_Kennzeichnung_der_Kraftfahrzeuge_mit_geringem_Beitrag_zur_Schadstoffbelastung
- http://www.dekra.de/feinstaub/
- https://www.academia.edu/20924531/Industry_4.0_Prototype