**M S Ramaiah Institute of Technology**

(An Autonomous Institute, Affiliated to VTU)

MSR nagar, MSRIT post, Bangalore-54

**A Mid-Sem Evaluation Report on**

**Effective prediction and prevention of Air pollution caused due to automobiles using IoT and Data Analytics Techniques**

Submitted by

Adarsh Raj 1MS12CS005

Anvitha D Rao 1MS12CS010

Vishal Mahuli 1MS12CS134

Vishwanath 1MS12CS135

Under the guidance of

Dr. Seema S

Associate Professor

Dept. of Computer Science & Engineering

M S Ramaiah Institute of Technology, Bangalore



**DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING**

**M.S.RAMAIAH INSTITUTE OF TECHNOLOGY**

**(Autonomous Institute, Affiliated to VTU)**

**BANGALORE-560054**

[www.msrit.edu](http://www.msrit.edu)

**ABSTRACT**

Climate change and Environmental Hazards have been burning issues all around the globe. Even the super powers of the world are facing these problems and solutions to these issues are the need of the hour. Air Pollution is a major contribution to the Environmental Pollution which not only deteriorates the quality of air and surroundings but also causes catastrophic consequences on the human health. Air Pollution is the presence in the ambient atmosphere of substances that interfere in the normal health of the life forms and quality of the substances causing hazards in both.

The major contributors to the air pollution are automobiles. Today, with the growth of technology and enhancement of lifestyles automobiles are found even in the remotest places of world where the population is negligible. The use of vehicles is rapidly growing as they're being deployed almost everywhere. So, it is of paramount importance to control this pollution caused by automobiles to tackle the air pollution issue effectively.

That said, In order to formulate a solution to this burning global issue we've come up with an idea that inculcates techniques of IoT/IoE (Internet of Things) and Data Analytics to predict and prevent air pollution substantially. The project aims at developing a device that will monitor the toxicity of the vehicular emission and assist the owner in getting the vehicle serviced timely to keep the emission levels low. Another objective of the project work is to analyze the air pollution levels at different geographical locations and to derive certain useful insights out of it that would help to reduce overall air pollution levels to a considerable extent.

**INDEX**

|  |  |
| --- | --- |
| **CONTENTS** | **PG NO** |
| 1. **Abstract** | **2** |
| 1. **Project Synopsis** | **4** |
| 1. **Project Plan** | **12** |
| 1. **Software Requirement Specification** | **18** |
| 1. **Literature Survey** | **26** |
| 1. **Design** | **30** |
| 1. **Conclusion** | **37** |
| 1. **References** | **38** |

1. **PROJECT SYNOPSIS**

**1.1 PROBLEM STATEMENT**

Climate change and Environmental Hazards have been burning issues all around the globe. Even the super powers of the world are facing these problems and solutions to these issues are the need of the hour. Air Pollution is a major contribution to the Environmental Pollution which not only deteriorates the quality of air and surroundings but also causes catastrophic consequences on the human health. Air Pollution is the presence in the ambient atmosphere of substances that interfere in the normal health of the life forms and quality of the substances causing hazards in both.

The problem ahead of us mainly is to develop a technical solution for the air pollution occurred due to the usage of automobiles. Problems include:  
-How to reduce the emission of toxic gases from vehicles?  
-How to assist the owners of automobile in notifying them about the emission of their automobile and harm caused by it to the surroundings?  
-How to reduce the air pollution in the city as a whole by routing the vehicles to different routes prioritized based on the pollution of the specific areas?  
-How to frequently notify people about the increasing pollution in specific areas and provide solutions to reduce it?  
-Most importantly, how to deploy technology in solving the above mentioned problems?

Technology will be vital in formulating a solution to these problems provided some of the following barriers are overcome:  
-The solution should be eco-friendly.  
-The solution should be of low cost and judicial power consumption.  
-Since the solution consists of a hardware product, it should be made compact and user-friendly.  
-The solution should be durable and fault tolerant.  
-The solution provided must be subjected to timely update and scaling with reference to the changing parameters  
  
**1.2 SYNOPSIS**

Air pollution is the introduction of particulates, biological molecules, or other harmful materials into Earth's atmosphere, causing diseases to humans, damage to other living organisms such as animals and food crops, or the natural or built environment. Air pollution can be caused due to various human activities such as industries, automobiles and burning of fossil fuels like wood ,coal etc. or naturally. The air pollution from vehicles in urban areas, particularly in big cities, has become a serious problem. With the increase in the number of vehicles due to urbanization, air pollution has increased rapidly in the past few years. The primary pollutants emitted from these automobiles are carbon monoxide, oxides of nitrogen and unburned hydrocarbons. CO is considered to be the most dangerous among all these. The health risks of air pollution are extremely serious leading to various diseases such as cancer, asthma , **Cardiovascular Disease, diabetes, bronchitis and also putting the elderly and the kids at a higher risk. As a result various measures are taken to reduce the vehicular pollution.**

**The factors that contribute to vehicular pollution are poor fuel quality, old vehicles, inadequate maintenance, old automotive technologies and traffic management. Thus vehicles that are more fuel efficient and those that produce fewer emissions are some of the means by which we can reduce transport related air pollution..** Emission from vehicles cannot be completely avoided but, it definitely can be controlled. **In this system we are trying to control the emission of poisonous gases due to incomplete combustion** which can be achieved by restricting the fuel supply to the engine when the level of pollutant let out by the vehicle exceeds the predefined safe value.

**1.3 OBJECTIVES:**

The objective of the project lies in reducing the air pollution and finding a technical solution for the same.

The project aims at enabling facilities such as:

***1.3.1 Message intimation by the system to the owner of the vehicle for the service that is required by the vehicle on excess of the carbon monoxide and/or nitrogen oxides.***

A message will be sent to the owner of the vehicle whenever his/her vehicle emits excess of CO and/or NOx, which is decided by the threshold set for these values and also based on the area the owner resides. This message will intimate the owner to get the vehicle serviced so as to get the vehicular emissions under control. The time period provided for getting the vehicle serviced will again depend on severity of pollution in that particular area where the vehicle is registered.

***1.3.2 Warning message to be sent to the owner as a reminder to the inactivity towards servicing of the vehicle.***

After the first message intimation, if at all the owner is unable to get the vehicle serviced; this second message will serve as a reminder to the owner about the status of the vehicle’s bad condition.

***1.3.3 A device to be developed that blocks the supply of fuel to the engine thereby deactivating the vehicle if the owner fails to get the vehicle serviced after series of intimations.***

Even after both the intimations if the owner doesn’t respond towards the servicing of the vehicle, in order to stop the usage of the vehicle, this device will block the fuel supply and hence will ensure that the vehicle can’t be used until and unless it is serviced .

***1.3.4 Ranking the various areas of the city based on the air pollution levels. Graphical representations for various parts of the city based on various parameters should be provided.***

Based on the contamination level in the air for a particular region, it will be provided with a rank which will eventually be used in deciding the amount of time provided to the vehicle owners when their vehicle’s emissions are exceeding threshold. Various graphical representations for the same are to be provided based on various parameters like locality or type of area etc.

***1.3.5 Predictions should be provided based on the current levels of the pollution and certain precautionary measures to be suggested.***

Based on the available data, certain predictions will be made informing people about the adverse effects of the rising air pollution. These predictions will be basically about what all can happen if the same contamination rate of air continues for certain period of time. Along with such predictions, precautionary measures can be suggested to the general public and to spread the awareness about it. We call it “Tip of the week” and "Tip of the day". The former will alert the people about uneven increase of the pollutants in the atmosphere of the area over a week and provides air pollution preventive measures. The latter is a daily notification sent to all the registered vehicle users which contains information about the highly polluted area and asking the residents/passers of'/from that area not to use their vehicles unless there's necessity.

***1.3.6 Routing of vehicles based on Air pollution at different locations.***

The route which a vehicle takes to reach a destination from a source is usually decided based on the distance, traffic etc but the project aims at considering another important parameter: Air pollution levels, to route. The vehicles can take different routes from the usual so as to account as least as possible for the air pollution in the city.

**1.4 TECHNICAL DETAILS:**

**1.4.1 MQ7 –CO Sensor**

Sensitive material of MQ-7 gas sensor is SnO2, which with lower conductivity in clean air. It make detection by method of cycle high and low temperature, and detect CO when low temperature (heated by 1.5V). The sensor’s conductivity is more higher along with the gas concentration rising. When high temperature (heated by 5.0V), it cleans the other gases adsorbed under low temperature. Please use simple electro-circuit, Convert change of conductivity to correspond output signal of gas concentration.

MQ-7 gas sensor has high sensitity to Carbon Monoxide. The sensor could be used to detect

different gases contains CO, it is with low cost and suitable for different application.

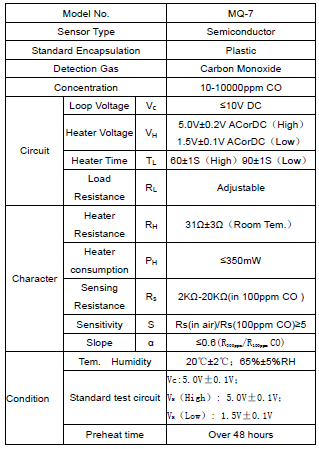


Figure1. MQ-7 Sensor

Table 1. MQ-7 Datasheet

**1.4.2 MQ135 GAS Sensor**

MQ135 Semiconductor Sensor for Air Quality Control Sensitive material of MQ135 gas sensor is SnO2, which with lower conductivity in clean air. When the target combustible gas exist, The sensor’s conductivity is more higher along with the gas concentration rising. Please use simple electro-circuit, Convert change of conductivity to correspond output signal of gas concentration. MQ135 gas sensor has high sensitivity to Ammonia, Sulfide and Benzene steam, also sensitive to smoke and other harmful gases. It is with low cost and suitable for different application.

Figure 2. MQ-135 Sensor

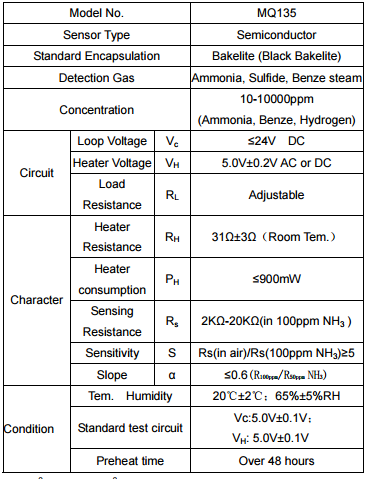


Table 2. MQ-135 Datasheet

**1.4.3 Arduino**

Arduino is an open-source prototyping platform based on easy-to-use hardware and software. Arduino is an open-source computer  hardware and software company, project and user community that designs and manufactures microcontroller-based kits for building digital devices and interactive objects that can sense and control objects in the physical world. Arduino boards are able to read inputs - light on a sensor, a finger on a button, or a Twitter message - and turn it into an output - activating a motor, turning on an LED, publishing something online. You can tell your board what to do by sending a set of instructions to the microcontroller on the board.  Arduino consists of both a physical programmable circuit board (often referred to as a microcontroller) and a piece of software, or IDE (Integrated Development Environment) that runs on your computer, used to write and upload computer code to the physical board.

The Uno board can be powered via the USB connection or with an external power supply. The power source is selected automatically. The board can operate on an external supply from 6 to 20 volts. The ATmega328 has 32 KB (with 0.5 KB occupied by the boot-loader). It also has 2 KB of SRAM and 1 KB of EEPROM (which can be read and written with the [EEPROM library](https://www.arduino.cc/en/Reference/EEPROM)).

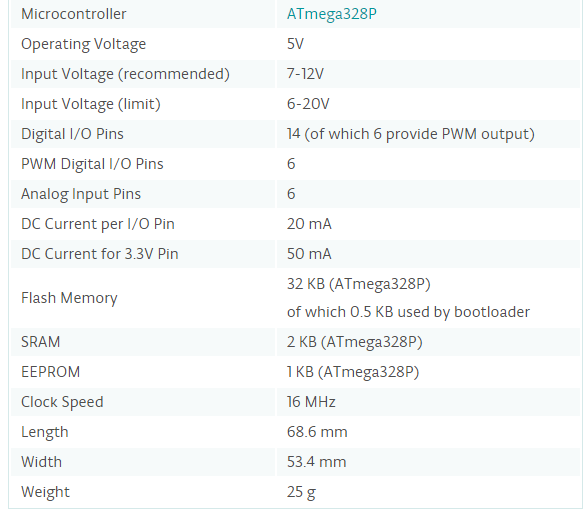


Table 3. Arduino Datasheet



Figure 3. Arduino Board

**1.5 INNOVATIVENESS AND USEFULNESS:**

* Deployment of a technical solution to an environmental issue.
* Devising a user-friendly, eco-friendly, low cost device to control the toxicity of the vehicular emissions.
* Using Data Analytics to analyze the air pollution parameters and derive insights out of the data to acknowledge people about the amount of air pollution and preventive measures of it.
* Enforcing effective solutions to restrict the usage of automobile.
* Using analytics to predict the growth of air pollution and provide solutions to avoid the hazards.
* Innovative method of routing the vehicles based on the pollution levels of different areas so as to avoid over pollution at key locations unlike the existing routing algorithms that prioritize routes based on the traffic.
* Easily implementable hardware which eventually might be ubiquitous in automobile.
* Most important use of the whole project is to tackle causes of air pollution and reduce the same.

**2. PROJECT PLAN**

**2.1 GANTT CHART:**

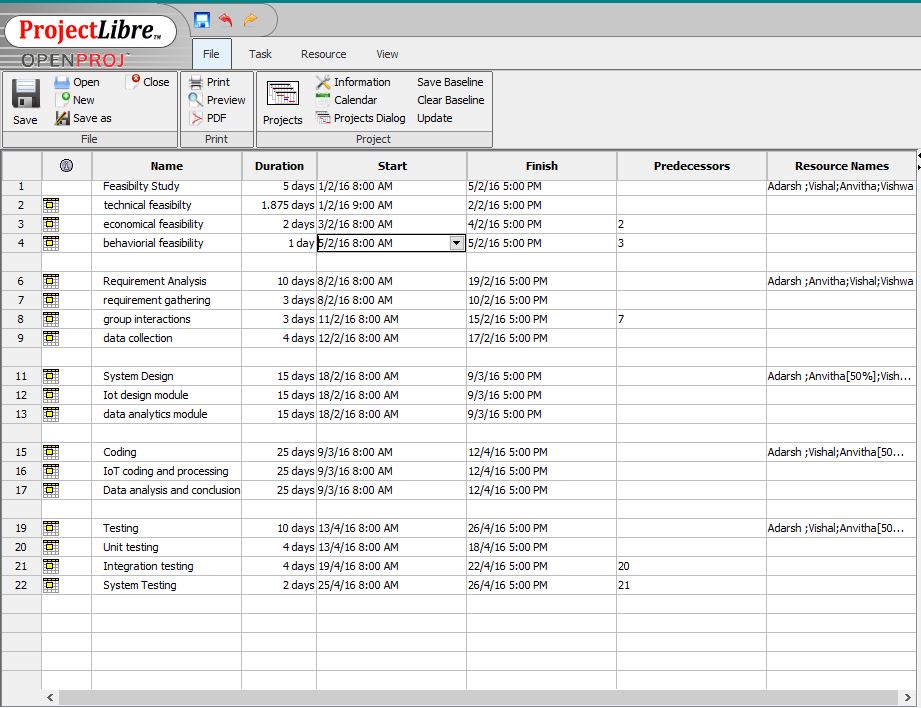
****

Figure 4. Snapshot of WBS (Work Breakdown Structure)

****

Figure 5. Snapshot of Timeline

**2.2 EFFORT ESTIMATION**

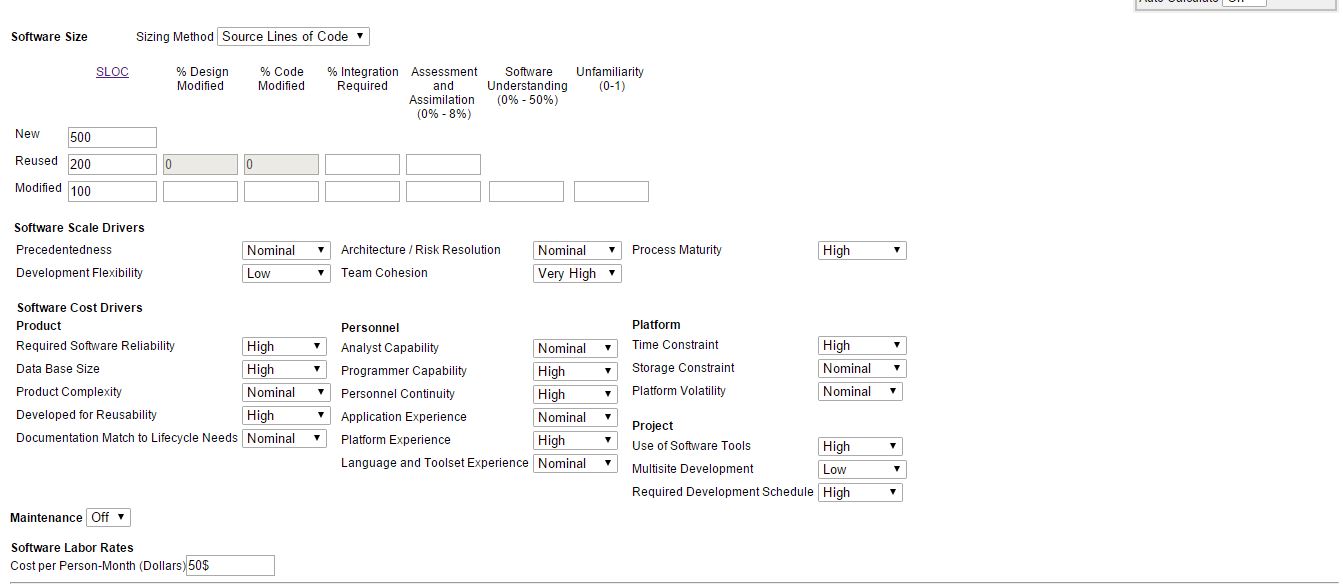
****

Figure 6. Snapshot of COCOMO

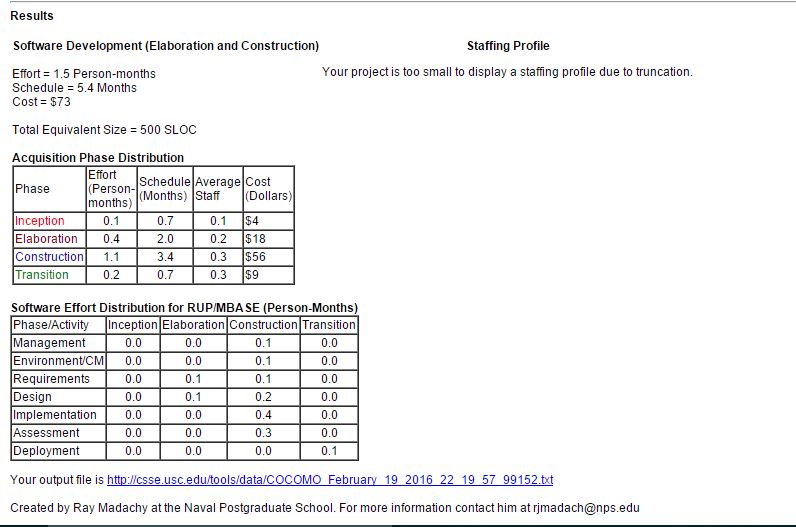
****

Figure 7. Snapshot of COCOMO

**2.3 SOFTWARE DEVELOPMENT MODEL: Agile SCRUM Methodology**

**2.3.1 Introduction**  
In earlier day’s software development activity followed “code and fix” approach. This approach worked well for small systems but failed when systems grew larger and when there was need to add new features. To avoid this Engineering methodology came into picture. The aim of these methodologies was to make the software process more predictable and efficient by having a strong emphasis on planning activity. This approach works well for other engineering fields, like lot of planning is needed to build a bridge or a house. But the software market is ever changing and brings in greater choices into market. Users and managers must deal with issues like what to include and what to exclude in the software, which technologies to use, what will give the company a competitive edge. These questions difficult to answer and trying to predict them in a rapidly changing market is even more difficult.   
As a reaction to these methodologies, lightweight methodologies like agile methodologies appeared on the scene. Agile methodologies attempt to compromise between no process and too much process.

**2.3.1.1 Agile methods are adaptive rather than predictive.**  
Engineering methods try to plan out in great detail for long span of time, this approach works well till there are no changes in design. Agile methods welcome change and try to adapt and thrive on change.

**2.3.1.2Agile methods are people oriented rather than process oriented.**  
Engineering methods define processes so that they work no matter the skill of the workers. It says that individuals are not as important as their roles. This approach is correct for a factory where workers are not the most intelligent and creative people. Agile methods state that no process will ever make up for the skill and intelligence of the development team. So role of process is to support development team in their work.  
  
**2.3.2 Agile Process Philosophy.**  
Agile software development philosophy has its roots in the reality of today’s markets. Agile software processes attempt to deal with issues introduced by rapidly changing and unpredictable markets. The “Manifesto for Agile software development” [1] the basic ideas of the philosophy are introduced through four basic values.  
  
**• Individuals and interactions over processes and tools.  
• Working software over comprehensive documentation.  
• Customer collaboration over contract negotiations.  
• Responding to change over following a plan.**  
  
The items to the right have value, however, the items on the left define the agile philosophy. In this paper, we will focus on the left hand side items to explain the agile software development. We will now look at all the four aspects in detail.

**2.3.3 Individuals and interactions**  
Adaptive process requires a very effective team of developers. The team has to work well together to be more effective. Face to face meetings have special importance in agile processes. It is believed that people respond quicker and transfer ideas more rapidly when talking face to face than they can when reading or writing documentation.. Extreme programming introduces the concept of pair programming where two developers develop a module together to provide much better and quicker output than the same job done individually. The concept of synergy (i.e. the interaction of two or more agents or forces so that the combined effect is greater than the sum of their individual effects) takes hold because a few designers, sharing a common space, working together, can produce more code quicker than can the same individuals working alone.  
  
In traditional methodologies treat people as resources that are like replaceable part. As stated earlier they say that individuals are not as important as their roles. They fail to understand that each individual is dynamic and unpredictable. When we are programming a computer, we are controlling a predictable device. But when handling human beings this approach fails. Treating individuals are replaceable resources reduces their morale and they look for much better working environments.

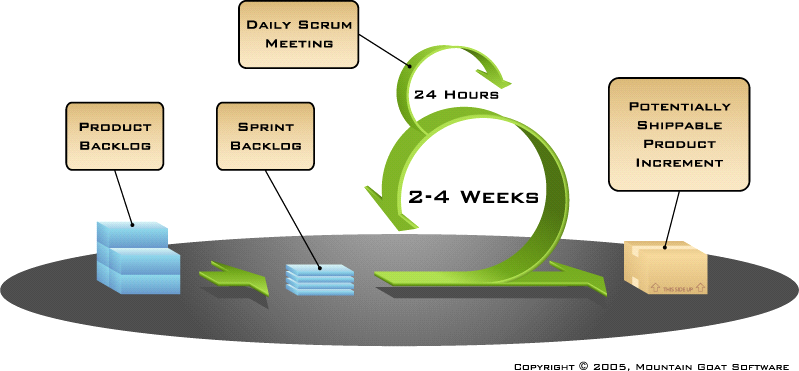


Figure 8. Agile Architecture

**2.4 RISK IDENTIFICATION AND MITIGATION**

**2.4.1 TOP 5 RISKS**

**1) Operational risk:** Operational risks are associated with the ability of the team to realize the idea into a working model. It involves correctness of the implementation techniques.  
**RiskMitigation:***Restructure the Project:* Implementation techniques need to be restructured and redesigned in a timely fashion to make sure that the end product is as per the user requirement.  
*Stage the Project*: Creating a stage to the project at different levels of development to test the features, fix the errors if exist and rebuild certain units of the project which are obsolete or dysfunctional.

**2) Personnel Shortfall:** This risk is associated with the shortage of team members required for the project. Also, uneven investment of time imposes this risk.  
**RiskMitigation**:  
*Personnel Training:* Some other people need to be trained with the skill-set required for the project so that they can replace absentees. *Personnel Counseling:* Constant counseling sessions for the personnel is to be arranged to understand the problems and causes of leaving the team and try to overcome them and retain the member.

**3)Financial Risk:** This risk is associated with the scarcity of money required to accomplish the task. Lack of funding not only poses threat to the quality of the end product but also to the overall completion of the task.  
**Risk Mitigation:***Financial Planning:* Along with the project planning it is also important to build an appropriate financial plan according to which there should be judicial usage of the existing funds alongside taking care of the quality of the product. *Sponsorship:* Team members must also meet investors to get them into investing in the project and build for themselves a strong financial support.

**4)Market Risk:** This risk is associated with the rapidly varying trends in the market. The market risk also involves the competency of the product in the market in comparison to the other existing products of the similar type.  
**Risk Mitigation:***Understanding Market trends: Understanding the market trends related to economic variation, devaluation of currency, recession etc is very important before formulating a project plan. Based on the market trends, the type of the product, the features to be added to the product, the annual pricing of the product etc will be decided.*

**5)Technology Risk:**This risk is associated with the technology being used in the development process at every step. If the technology is obsolete then the product would not possess enough Scredibility to proliferate into all sectors of the market.**Risk Mitigation:** *Usage of modern Technology:* The team members must make sure that the technology used is new and hasn't been implemented much. It should also be appropriate to the given output of the project.  
*Optimization of an existing technology:* Team members should formulate techniques if possible to reuse the old technologies in an innovative way and optimize it if possible*.*

1. **SOFTWARE REQUIREMENT SPECIFICATION**
   1. **Product/Project Overview**

In order to formulate a solution to this burning global issue we've come up with an idea that inculcates techniques of IoT/IoE (Internet of Things) and Data Analytics to predict and prevent air pollution substantially. The project aims at developing a device that will monitor the toxicity of the vehicular emission and assist the owner in getting the vehicle serviced timely to keep the emission levels low. Another objective of the project work is to analyze the air pollution levels at different geographical locations and to derive certain useful insights out of it that would help to reduce overall air pollution levels to a considerable extent.

* 1. **External Interface Requirements**

**3.2.1 User Interfaces**

User communication with the system is limited to one of the outcomes of the project which is "Pollution based Routing". An android app will be built that takes user registration and provides an interface for the user to enter the source and destination of his travel and using Google Maps the appropriate route prioritized on pollution levels will be shown and accordingly can be used for navigation.

**3.2.2 Hardware Interfaces**

The Hardware interfaces include the following:

1. Arduino Uno 3:

|  |  |
| --- | --- |
| Microcontroller | [ATmega328P](http://www.atmel.com/Images/doc8161.pdf) |
| Operating Voltage | 5V |
| Input Voltage (recommended) | 7-12V |
| Input Voltage (limit) | 6-20V |
| Digital I/O Pins | 14 (of which 6 provide PWM output) |
| PWM Digital I/O Pins | 6 |
| Analog Input Pins | 6 |
| DC Current per I/O Pin | 20 mA |
| DC Current for 3.3V Pin | 50 mA |
| Flash Memory | 32 KB (ATmega328P) of which 0.5 KB used by bootloader |
| SRAM | 2 KB (ATmega328P) |
| EEPROM | 1 KB (ATmega328P) |
| Clock Speed | 16 MHz |
| Length | 68.6 mm |
| Width | 53.4 mm |
| Weight | 25 g |
|  |  |
|  |  |

Figure 9. Arduino Datasheet

**3.2.3. ESP8266 WiFi Module:**

* + - * + 32-bit [RISC](https://en.wikipedia.org/wiki/Reduced_instruction_set_computing) CPU: [Tensilica](https://en.wikipedia.org/wiki/Tensilica) Xtensa LX106 running at 80 MHz
        + 64 KiB of instruction RAM, 96 KiB of data RAM
        + External QSPI flash - 512 KiB to 4 MiB (up to 16MiB is supported)
        + [IEEE 802.11](https://en.wikipedia.org/wiki/IEEE_802.11) b/g/n [Wi-Fi](https://en.wikipedia.org/wiki/Wi-Fi)
        + Integrated [TR switch](https://en.wikipedia.org/wiki/Duplexer#Transmit-receive_switch), [balun](https://en.wikipedia.org/wiki/Balun), [LNA](https://en.wikipedia.org/wiki/Low-noise_amplifier), [power amplifier](https://en.wikipedia.org/wiki/RF_power_amplifier) and [matching network](https://en.wikipedia.org/wiki/Matching_network)
        + [WEP](https://en.wikipedia.org/wiki/Wired_Equivalent_Privacy) or [WPA/WPA2](https://en.wikipedia.org/wiki/Wi-Fi_Protected_Access) authentication, or open networks
        + 16 [GPIO](https://en.wikipedia.org/wiki/General-purpose_input/output) pins
        + [SPI](https://en.wikipedia.org/wiki/Serial_Peripheral_Interface_Bus), [I²C](https://en.wikipedia.org/wiki/I%C2%B2C),[I²S](https://en.wikipedia.org/wiki/I%C2%B2S) interfaces with DMA (sharing pins with GPIO)
        + [UART](https://en.wikipedia.org/wiki/Universal_asynchronous_receiver/transmitter) on dedicated pins, plus a transmit-only UART can be enabled on GPIO2.

**3.2.4 MQ-7 Gas Sensor:**

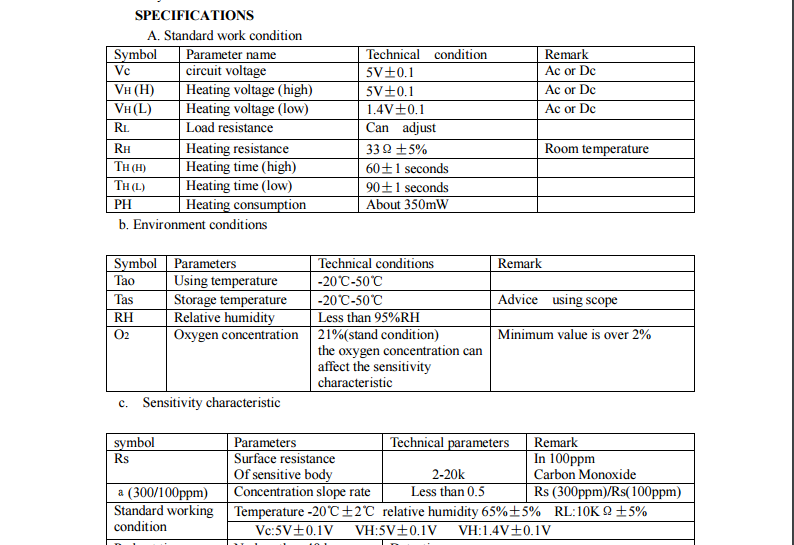


Figure 10. Snapshot of MQ-7 Datasheet

3.2.5 MQ-135 Gas Sensor

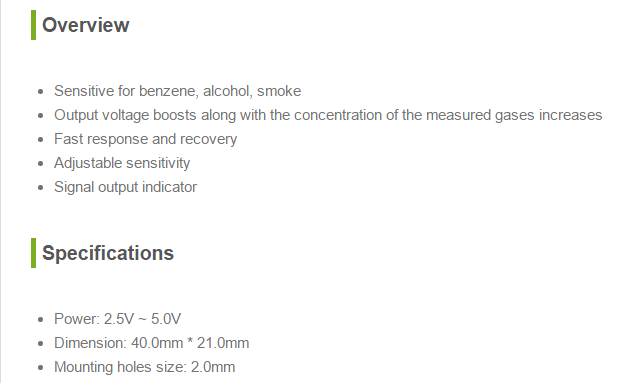


Figure 11. Snapshot of MQ-135 Datasheet

3.2.6 GSM Board:

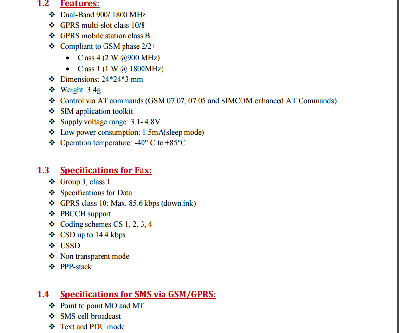


Figure 12. Snapshot of GSM Board Datasheet

**3.3 Software Interfaces**

3.3.1 **Arduino Platform:**

The open-source Arduino Software (IDE) makes it easy to write code and upload it to the board. It runs on Windows, Mac OS X, and Linux. The environment is written in Java and based on Processing and other open-source software.  
This software can be used with any Arduino board.

**3.3.2 Communication Interfaces**

1. ESP8266 WiFi Module:

* + - * + 32-bit [RISC](https://en.wikipedia.org/wiki/Reduced_instruction_set_computing) CPU: [Tensilica](https://en.wikipedia.org/wiki/Tensilica) Xtensa LX106 running at 80 MHz
        + 64 KiB of instruction RAM, 96 KiB of data RAM
        + External QSPI flash - 512 KiB to 4 MiB (up to 16MiB is supported)
        + [IEEE 802.11](https://en.wikipedia.org/wiki/IEEE_802.11) b/g/n [Wi-Fi](https://en.wikipedia.org/wiki/Wi-Fi)
        + Integrated [TR switch](https://en.wikipedia.org/wiki/Duplexer#Transmit-receive_switch), [balun](https://en.wikipedia.org/wiki/Balun), [LNA](https://en.wikipedia.org/wiki/Low-noise_amplifier), [power amplifier](https://en.wikipedia.org/wiki/RF_power_amplifier) and [matching network](https://en.wikipedia.org/wiki/Matching_network)
        + [WEP](https://en.wikipedia.org/wiki/Wired_Equivalent_Privacy) or [WPA/WPA2](https://en.wikipedia.org/wiki/Wi-Fi_Protected_Access) authentication, or open networks
        + 16 [GPIO](https://en.wikipedia.org/wiki/General-purpose_input/output) pins
        + [SPI](https://en.wikipedia.org/wiki/Serial_Peripheral_Interface_Bus), [I²C](https://en.wikipedia.org/wiki/I%C2%B2C),[I²S](https://en.wikipedia.org/wiki/I%C2%B2S) interfaces with DMA (sharing pins with GPIO)
        + [UART](https://en.wikipedia.org/wiki/Universal_asynchronous_receiver/transmitter) on dedicated pins, plus a transmit-only UART can be enabled on GPIO2.

3.3.3 GSM Board

.

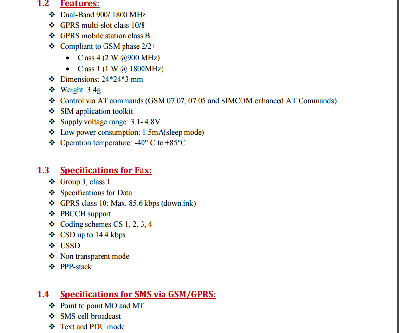


Figure 13. Snapshot of GSM Specifications

**3.4 Functional Requirements**

1. **Sensing the gaseous value for CO2 from the Exhaust emission**

The MQ7gas sensor senses the presence of CO2 from the emission and determines the level of content of CO2 gas in the emission.

1. **Sensing the gaseous value for NO2 from the Exhaust emission**

The MQ135 gas sensor senses the presence of NO2 from the emission and determines the level of content of NO2 gas in the emission.

1. **Sending the data to the main server using Wi-Fi**

Once the values are sensed from the emission, it has to be sent to the main server for processing on the value collected and hence requires a Wi-Fi Module.

1. **Processing the data obtained from sensors**

Once the data has been received from the sensors by the board and it has sent the data to the server, it has to be processed further. Based on the data received from the sensor, if the values cross the threshold for those gaseous levels, a first warning message will be sent about the condition of emission and the vehicle service period will be intimated to the owner.

1. **Second warning message to be sent as a reminder to the owner**

Once in the first message the owner is intimated about the period for servicing of his vehicle. Again a second reminder is sent to the owner if he still hasn’t got his vehicle serviced. Thisreminder is only sent if the owner has still notgot hisvehicle serviced even after first notification, which is generally sent in after half the time period allotted to him for getting his vehicle serviced.

1. **Choking mechanism**

Even after the second intimation, if the owner doesn’t responds and doesn’t get his vehicle serviced, our mechanism will choke the fuel pipe and hence until and unless he services his vehicle, he won’t be able to use it.

1. **Ranking the various areas of the city based on the air pollution levels. Graphical representations for various parts of the city based on various parameters should be provided.**

Based on the contamination level in the air for a particular region, the region will be provided with a rank which will eventually be used in deciding the amount of time provided to the vehicle owners when their vehicle’s emissions are exceeding threshold. Various graphical representations for the same are to be provided based on various parameters like locality or type of area etc.

1. **Predictions should be provided based on the current levels of the pollution and certain precautionary measures to be suggested.**

Based on the available data, certain predictions will be made informing people about the adverse effects of the rising air pollution. These predictions will be basically about what all can happen if the same contamination rate of air continues for certain period of time. Along with such predictions, precautionary measures can be suggested to the general public and to spread the awareness about it. We call it “Tip of the week” and "Tip of the day". The former will alert the people about uneven increase of the pollutants in the atmosphere of the area over a week and provides air pollution preventive measures. The latter is a daily notification sent to all the registered vehicle users which contains information about the highly polluted area and asking the residents/passers of'/from that area not to use their vehicles unless there's necessity.

1. **Routing of vehicles based on Air pollution at different locations.**

The route which a vehicle takes to reach a destination from a source is usually decided based on the distance, traffic etc but the project aims at considering another important parameter: Air pollution levels, to route. The vehicles can take different routes from the usual so as to account as least as possible for the air pollution in the city.

**3.4 Software System Attributes**

**3.4.1 Reliability**

* This system is available all time and mainly the reliability of the system depends on the sensors and the board. As the lifespan of board and sensors are more, the reliability factor is also more.

**3**.**4.2 Availability**

* It has to be available all the time 24x7 , provided users have Internet to access the web page and also for sensors to push data to the server.

**3.4.3 Security**

* The catalogue is made available only to the admin and the users. A separate

Login is made available to the user through which he can access information about his vehicle’s air pollution level and also about various preventive measures, which requires authentication. Also the routing service will be accessed by him on that particular app. Thus the data is made secure by making it available only to the concerned people. Any modification to the database shall be synchronized and be done only by the database admin.

**3.4.4 Portability**

* The system developed is portable, as small and light components with wireless connections are used .It can run over different platforms.

**3.4.5 Maintainability**

* This system is easily maintainable as it has fewer components and its design is not complicated. As the lifespan of sensors and the board is high, so it requires less maintenance. It is also well documented thus making it easy to use.

**3.4.6 Performance**

* It depends on computer hardware specifications like RAM, GPU etc. Since we are handling huge data, it should be processed in parallel. The system shall give responses in 1 second after collecting the emitted gases using sensors. The user-interface screen shall respond within 5 seconds. The System supports 1 vehicle at a time.

3.5. **Performance Requirements**

* If any excess of emission of those gases are detected, the owner should get the intimation of the excess levels and time period for servicing his vehicle within next 1 minute.
* Every morning by 8 a.m., “Tip of the day” should be sent to the owners.
* Second intimation should be sent to the owner in the halfway of the time period for servicing (Only if he has not got his bike serviced).
* The data analysis of the air pollution data has to be closely monitored and every month, the threshold should be updated.
* Routing should be highly responsive and suggest best routes possible based on air pollution levels.

3.6. **Database Requirement**

* A table has to be maintained about the air pollution level and has to be continuously updated as and when data is received from authoritative bodies.
* There are two separate tables i.e. one will contain owner-specific data(owner’s personal information and vehicular emission levels and servicing records.) and other will have data about the city’s pollution level and will be updated at regular intervals.

**3.7 Design Constraints**

1. **Hard drive space:** The amount of space an application needs for storage and execution purpose is major design limitation as Arduino Uno 3 uses a Flash Memoryfor storage.
2. **Application memory Usage:** The amount of memory space needed for an application to run is an important limitation in the design as Arduino Uno 3 has only 256MB RAM and it needs to effectively utilize it.
3. **Budget:** The amount of money that can be spent in the overall development of the project restricts the design of the system.
4. **Application Quality:** If the quality of the application is high then it means that the design is highly effective. Thus the quality of the application is an important deciding factor in the design of an application.

3.8. **Other Requirements (if any)**

Knowledge of Web programming, Python programming, R programming and other Data Anaytics Techniques, C programming with Arduino and connection of sensors to Arduino Uno 3 Board.

**4. LITERATURE SURVEY**

**4.1 INTRODUCTION**

The aim of this project is to effectively prevent and predict the air pollution caused by the automobiles using Arduino board, two gas sensors (MQ-7 and MQ-135) and a Wi-Fi module.

These components serve the IOT part of the project. The remaining half of the project is all about data analytics on data collected from various sources primarily, Karnataka State Pollution Control Board. The sensors sense the gaseous levels present in the effluents and emissions coming out of the exhaust and based on the threshold value set, the processor board decides whether an alert is to be sent to the owner or not about the emission level exceeding. These values are also sent by Wi-Fi module from the board to a specific IP so as to collect and analyze the data from every specified owner. An android application is also built to support the most important feature of the data analytics part i.e. Routing based on pollution levels in the cities.

We propose this system as this idea was brainstormed by referring to few IEEE papers and by using ideas of our own about the prevention of air pollution and how it is to be controlled. This system has various features which on implementation will affect the strategy of curbing the air pollution to a huge extent. We have the feature in which the system chokes/cuts down the fuel supply once the vehicle is not serviced in the allotted time period. The android application is basically for the routing feature which is based on suggesting you route based on pollution levels.

IOT is an emerging field and is the technology which is helping in automating almost everything. Data analytics always provide clear insights out of humongous data sets and helps to derive conclusions out of it. We try to use the benefits of both these fields into one concept which in today’s world is a burning topic i.e. AIR POLLUTION. This system is going to bring a revolutionary change to the entire concept of prevention to the air pollution and its control measures.

The Android application also provides the history of the emission records and also provides tips which we call “Tip of the day/week”. This tip is a small fact which can help spread awareness about the air pollution prevention and how some preventive measures can prove effective in this regard. Routing of the areas are based on the ranks assigned to them after the data from that area is being analyzed. Arduino is a cheap and compact but highly effective board, and has an A-to-D convertor in-built which is not there in RasberryPi board also. Hence we have chosen this board. Internet connectivity is very important in this system as every communication is dependent on it.

**4.2 Survey**

As our concentration is on prevention and prediction of air pollution from automobiles, we use two gas sensors i.e. MQ-7 and MQ-135. MQ-7 is a basically as CO gas sensor as CO is one of the primary content in the vehicular exhaust. [1] It is a real time work where a demo application has been made in which Arduino processor is used and a controller board is made where all these devices get integrated and work accordingly. The vehicle is controlled by this circuit. When a vehicle attains certain threshold pollution level then an SMS is generated and sent to the pre-defined number stored in the memory through the GSM module about the time period which he has been allotted for the servicing of the vehicle. The GPS module is used to locate the vehicle position where it is halted. This paper demonstrates an effective utilization of technology by which we save our environment by controlling the pollution of vehicles.

The aim of [2] is that the system is based on a smart sensor microcontroller equipped with a network capable application processor that downloads the pollutants level to a personal computer for further processing. The system monitors and transmits parameters atmospheric environment to a command center (admin’s server). From this paper, we got the idea of transferring all the data collected from the sensors to the admin’s server, for further analysis and this would also help to keep track about the owner’s vehicular emissions. This data collected from the sensors will be helpful in data representations and rankings for the particular regions/areas.

Further [3] gives us insights about the power management for the system. Any sensor system must be highly effective in terms of power usage and management. The lesser the system uses power, the longer time it can serve its purpose. Also from this paper, we came up with the thought of having an android application which will provide you with routing between areas not based on traffic but based on pollution levels. This idea of having an android application not only provides an edge to this system but also will help the owner to have a record of his own vehicle’s emission records. This android application will also be the means of spreading the awareness about the control of air pollution by our “Tip of the day/week”feature of the system.

The system that we have proposed is shown in Figure1. It efficiently takes ideas from different papers and integrates them to achieve maximum benefit and also ensures that the design is cost effective but at the same time efficiently solves the problem at hand. Thus the tradeoff between performance and cost has been effectively managed.

Message alert

Google Maps API

Android Application

GSM Module

MQ-7

(CO sensor)

Arduino board

(Processor)

Wi-Fi module

Admin’s server and database

MQ-135

(NOxsensor)

Fuel

Injector

Figure 14 .Proposed Architecture for the Air pollution control system (IOT and DA)

**5. DESIGN**

**Modular design**, or "modularity in design", is a design approach that subdivides a system into smaller parts called modules or skids, which can be independently created and then used in different systems. A modular system can be characterized by functional partitioning into discrete scalable, reusable modules, rigorous use of well-defined modular interfaces, and making use of industry standards for interfaces.

Thus this document initially describes the different modules present in the project. It is then followed by the various algorithms that are used to implement the modules. Later the system architecture is described which is followed by the description of the GUI and then various UML diagrams like class diagram, sequence diagram and data flow diagram are given. Finally, the document ends with the list of references used in developing this document.

The different modules needed for the project.

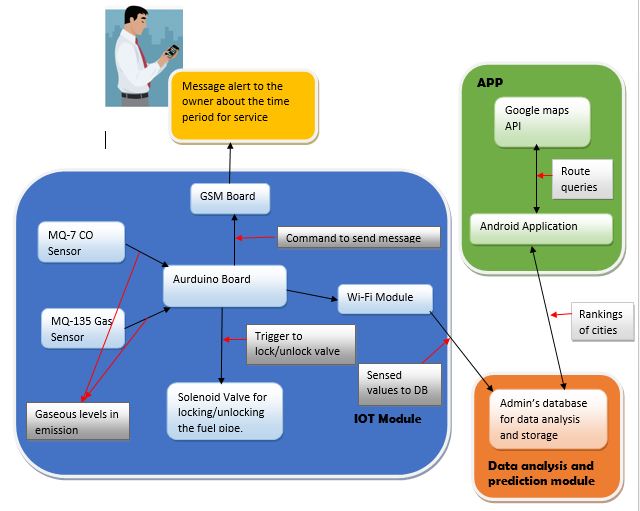
* **Gas Sensor Module**: This module is responsible for identifying the change in the NO2 and CO2 values in vehicles. In this module all the values are recorded and monitored.

* **Arduino Module:** This module is the most important module in the project. An embedded c program is written and flashed to the Arduino. This program ensures that the data is being read from the Gas sensors and also make sure that this analog voltage is converted to a digital value and is sent to the Database application using the Wi-fi module. Further suitable messages are sent to users and Air Pollution Control Board.
* **Wifi Module**: This module is responsible for interfacing a Wifi board with the Arduino to ensure that data communication takes place between the Android Application running on the user’s phone and the Arduino.
* **Database Module**: This module’s function is to insert the data into Online cloud and maintain structured data. This is responsible reducing redundancy and to Provide data for data analytics.
* **Data Analytics Module**: This module’s function is for Air Pollutoin control Boad to manage the Air Pollution Causing vehicals and to even Analyse the Pollution data with graphs and charts.
* **Android Application Module**: This Module is for the user to analyse thier vehicle Emission values periodically and to have notification when it exeeds the emmission control value.It will also provide and Map accordingly from data collected through aurdino which will be resulted in providing Alternative Route accordingly with less Air Pollution.

**5.1 ARCHITECTURE DESIGN**

A **system architecture** or **systems architecture** is the conceptual model that defines the structure, behavior, and more views of a system. An architecture description is a formal description and representation of a system, organized in a way that supports reasoning about the structures and behaviors of the system.

System architecture can comprise system components, the externally visible properties of those components, the relationships (e.g. the behaviour) between them. It can provide a plan from which products can be procured, and systems developed, that will work together to implement the overall system. The different modules present in this project has been described in the introduction. The system architecture is given in Figure.



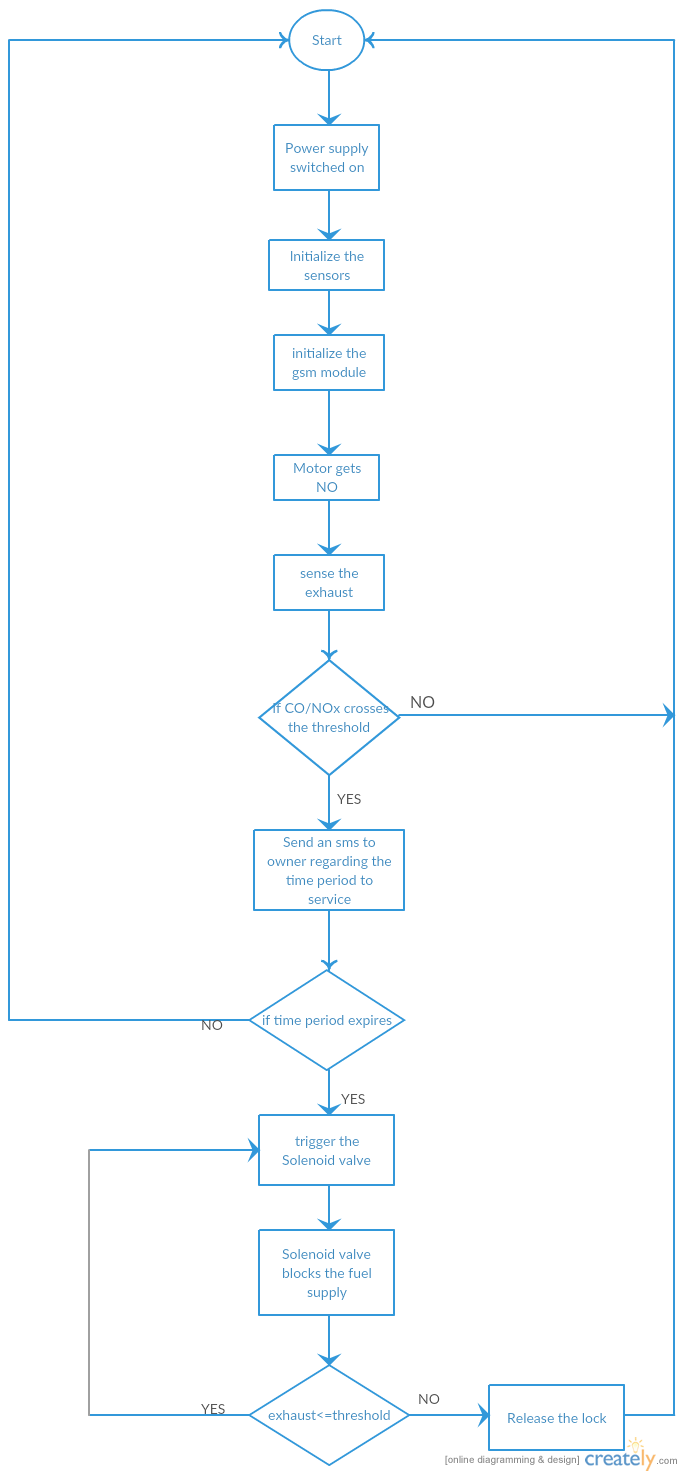
**Figure 15 System Architecture**

**5.2 GRAPHICAL USER INTERFACE**

The main graphical user interface is an Android Application which communicates with Cloud which intern communicates with Arduino via Wi-Fi Module. This application initially has a welcome login page for authentication with two buttons, namely login and Register. When the user enters the correct username and the password the login succeeds and redirects the application to the main page where a His vehicles Recent Emission data will be is displayed. If either the username or the password is invalid, then it prompts the user to enter the correct credentials. The Emission Data is generated depending on the voltages values of the 2 Sensor's CO2 and NO2 received via Bluetooth from the Aurdino. Thus the user’s Vehicals activities are identified on the basis of the voltage values a bar chart is generated to indicate the values of emission test. Further the Android application also has a button to view the activities performed by the user in the past to provide persistent storage. The GUI also has a button to genarate direction from one location to another location with minimunThe GUI has been planned to designed very efficiently so that it’s user friendly and at the same time has a good look and feel. The different functions used are ***loginButton, registerButton, viewData, processData, generatePie and genarateDirection***. Thus the Android application acts as the software graphical user interface.

**5.3 Data Flow Diagram:**

* A **data flow diagram** (**DFD**) is a graphical representation of the "flow" of data through an information system, modelling its processaspects. A DFD is often used as a preliminary step to create an overview of the system, which can later be elaborated. DFDs can also be used for the visualization of data processing (structured design).
* A DFD shows what kind of information will be input to and output from the system, where the data will come from and go to, and where the data will be stored. It does not show information about the timing of process or information about whether processes will operate in sequence or in parallel (which is shown on a flowchart).
* The entire flow of the system has been shown in the data flow diagram and every step has been decided based on the system modules and their respective functionality. The flow of the main process of the system has been shown in the figure (IoT Module).

****

Flowchart 1.

**5.4 CLASS DIAGRAM:**

* In software engineering, a class diagram in the Unified Modelling Language (UML) is a type of static structure diagram that describes the structure of a system by showing the system's classes, their attributes, operations (or methods), and the relationships among objects.
* The figure in the next slide represents the class diagram for wireless temperature monitoring system which uses association, aggregation, composition and generalization and many more features of the class model.
* The relationship among various classes and their interdependencies are effectively modeled using the class diagram.

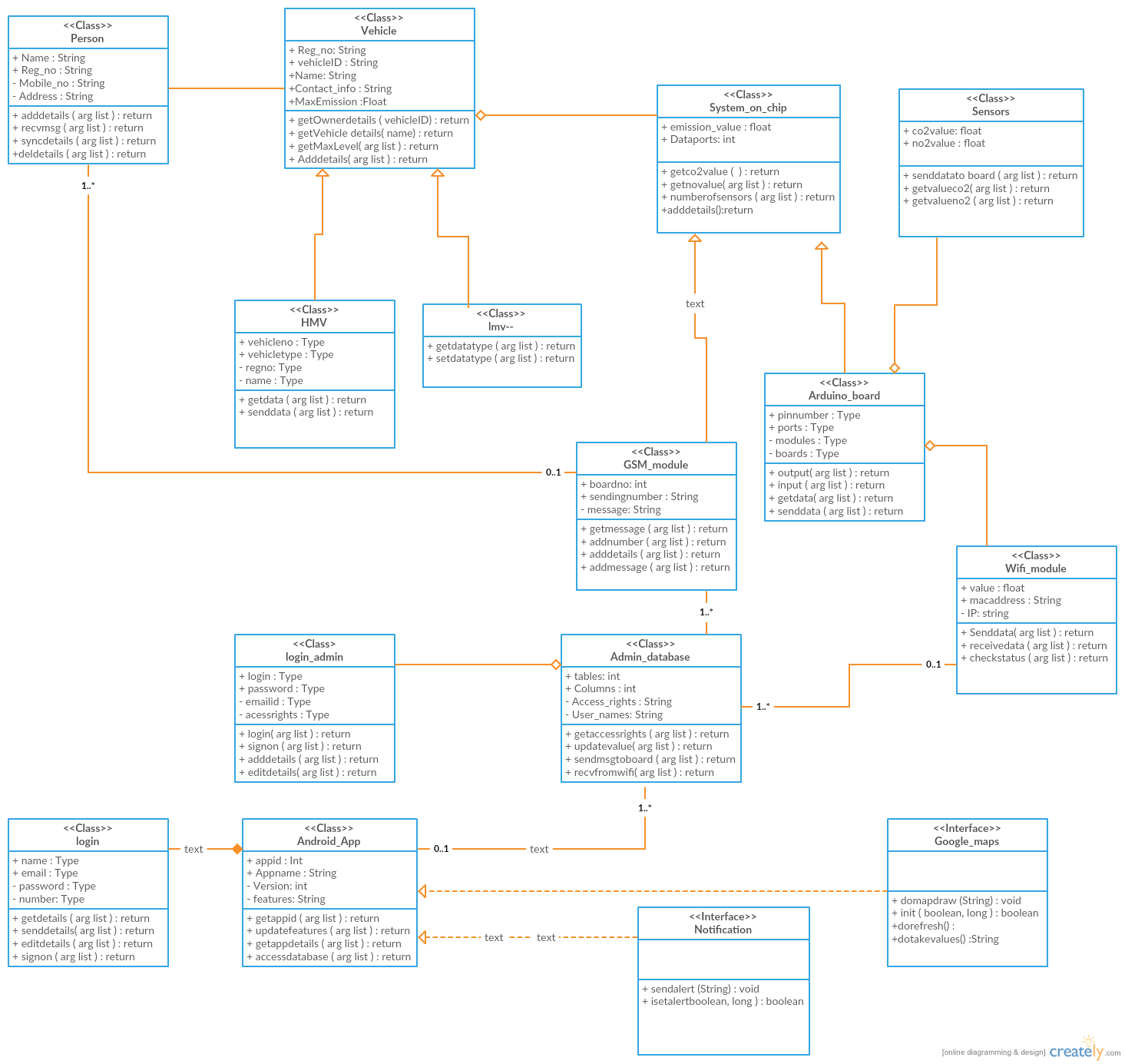
****

Figure 16. Class Diagram for the air pollution prevention and prediction system

**5.5 SEQUENCE DIAGRAM:**

* **Sequence diagram** is an interaction diagram that shows how processes operate with one another and in what order. It is a construct of a Message Sequence Chart. A sequence diagram shows object interactions arranged in time sequence. It depicts the objects and classes involved in the scenario and the sequence of messages exchanged between the objects needed to carry out the functionality of the scenario. Sequence diagrams are typically associated with use case realizations in the Logical View of the system under development. Sequence diagrams are sometimes called **event diagrams** or **event scenarios**.
* A sequence diagram shows, as parallel vertical lines (*lifelines*), different processes or objects that live simultaneously, and, as horizontal arrows, the messages exchanged between them, in the order in which they occur. This allows the specification of simple runtime scenarios in a graphical manner. The sequence diagrams for this project are as follows.
* There are various objects involved in the sequence diagrams and are clearly shown in the diagram.
* The required conditionals are mentioned in the comment box in the diagram.

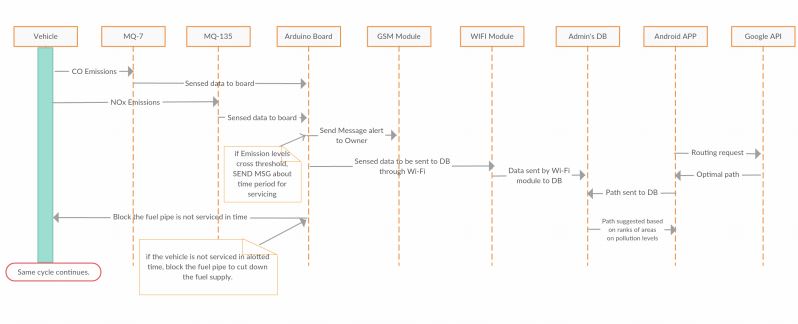
****

Figure 17. Sequence Diagram

**6. CONCLUSION**

The system which we have proposed is lightweight and compact and hence can be implemented on the vehicles. This system on implementation will surely help in reduction of current rate of generation of air pollution due to vehicular emissions. This entire concept is new and will serve for a very big change in the prevention and control measures for air pollution. The system is provided with a UI in the form of an android application which is very easy to use and can be easily deployed in any smart phone. The sensors and the other components used in building this system cost very less and hence the system is cost effective. Apart from being cost effective, this system is high in performance as it uses low power consumption sensors and also for the fact that these sensors are highly precise in detection. The routing facility to be provided in the android application is a totally new feature which will make use of Google maps API and will suggest the best suited path between two stations based on the pollution levels. This app will serve also as a record for the owner of the vehicle for his vehicle’s emissions. Also awareness tips will be provided on this app itself.

**7. REFERENCES**

1. Anita kulkarni , T. Ravi Teja

“**Automated System for Air Pollution Detection and Control in Vehicles**” in International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering, Vol. 3, Issue 9, September 2014, ISSN (Print): 2320 – 3765, ISSN (Online): 2278 – 8875, page no 12196,12197,12199.

1. A. R. Al-Ali, Member, IEEE, Imran Zualkernan, and Fadi Aloul

“**A Mobile GPRS-Sensors Array for Air Pollution Monitoring**”.

On IEEE SENSORS JOURNAL, VOL. 10, NO. 10, OCTOBER 2010, page no 1666,1667,1668,1670.

1. R.A.Roseline , Dr.M.Devapriya ,Dr.P.Sumathi

**“Pollution Monitoring using Sensors and Wireless Sensor Networks: A Survey”** in International Journal of Application or Innovation in Engineering & Management (IJAIEM), Volume 2, Issue 7, July 2013 ISSN 2319 – 4847, page no 119,120,121,123.

1. Sandip S. Patil, Prof. Jaykaran Singh

“**Monitoring and Controlling of Hazardous Gases inside Vehicle and Alerting Using GSM Technology**” in International Journal of Advanced Research in Computer Science and Software Engineering. Volume 5, Issue 1, January 2015, ISSN: 2277 128X, page no.1002, 1003, 1004, 1005.

1. <http://wiki.eprolabs.com/index.php?title=Gas_Sensor_MQ7>
2. <http://www.waveshare.com/wiki/MQ-135_Gas_Sensor>
3. <http://kspcb.gov.in/>
4. <http://www.eolss.net/sample-chapters/c09/e4-14-00-00.pdf>
5. <http://www.ardiono.com>

[10] http://wiki.eprolabs.com/index.php?title=Gas\_Sensor\_MQ7

[11] http://www.waveshare.com/wiki/MQ-135\_Gas\_Sensor

[12] http://kspcb.gov.in/

[13] http://www.eolss.net/sample-chapters/c09/e4-14-00-00.pdf

[14] http://www.ijareeie.com/upload/2014/september/46\_Automated.pdf

[15] http://davidhoulding.blogspot.in/2014/03/co-carbon-monoxide-gas-sensor-using.html

[16] http://wiring.org.co/learning/basics/airqualitymq135.html

[17] http://forum.arduino.cc/index.php?topic=172368.0

[18] http://www.ijarcsse.com/docs/papers/Volume\_5/1\_January2015/V5I1-0502.pdf

[19] http://events.awma.org/files\_original/ControlDevicesFactSheet07.pdf

[20] http://www.4cleanair.org/APTI/452combined.pdf

[21] http://www.waveshare.com/wiki/MQ-135\_Gas\_Sensor

[22] http://www.geeetech.com/wiki/index.php/Carbon\_Monoxide\_Sensor\_-\_MQ-7www.creately.com

[23] http://www.ijaiem.org/volume2issue7/IJAIEM-2013-07-14-041.pdf