VISVESVARAYA TECHNOLOGICAL UNIVERSITY
BELAGAVI- 590018

A Project Phase 2 Report on

"ADVANCE TRAFFIC MANAGEMENT SYSTEM USING GOOGLE CLOUD"

Submitted to

Visvesvaraya Technological University

In the partial fulfilment of requirements for the award of degree

BACHELOR OF ENGINEERING
IN
COMPUTER SCIENCE AND ENGINEERING

Submitted by

AKASH C P	**4RA16CS002**
B S SURABHI	**4RA16CS013**
MOHANKUMAR M C	**4RA16CS049**
POORNIMA H P	**4RA16CS064**

Under the Guidance of

Mr. DHRUVA M S B. E, M. Tech
Assistant Professor,
Department of Computer Science & Engineering
Rajeev Institute of Technology
Hassan.

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING
RAJEEV INSTITUTE OF TECHNOLOGY
HASSAN-573201

2019-2020

RAJEEV INSTITUTE OF TECHNOLOGY, HASSAN

(Approved by AICTE, New Delhi and Affiliated to VTU, Belagavi.)

Plot # 1-D, Growth Center, Industrial Area, B-M Bypass Road, Hassan-573201
Ph: (08172)-243180/80/84 Fax: (08172)-243183

Department of Computer Science &Engineering

CERTIFICATE

Certified that the **project phase 2 work** entitled **"ADVANCED TRAFFIC MANAGEMENT SYSTEM USING GOOGLE CLOUD"** is carried out by **Mr. AKASH C P [4RA16CS002]**, **Ms. B S SURABHI [4RA16CS013]**, **Mr. MOHANKUMAR M C [4RA16CS049]** and **Ms. POORNIMA H P [4RA16CS064]** respectively bonafide students of **RAJEEV INSTITUTE OF TECHNOLOGY, Hassan** in partial fulfillment for the subject **"PROJECT PHASE 2"** in **COMPUTER SCIENCE AND ENGINEERING** of Visvesvaraya Technological University, Belagavi during the year 2019-2020. The project report has been approved as it satisfies the academic requirements in respect of project work prescribed for the said degree.

Mr. DHRUVA M S	**Dr. H N PRAKASH**	**Dr. A N RAMAKRISHNA**
Assistant Professor	Head of the Department	Principal
Dept. of Computer Science & Engineering	Computer Science & Engineering, RIT, HASSAN	RIT, HASSAN
RIT, HASSAN		

Name of the examiners   Signature with date

1.

2.

ACKNOWLEDGEMENT

The satisfaction and euphoria that accompany the successful of any task would be incomplete without the mention of the people who made it possible, whose constant guidance and encouragement crowned our efforts with success.

We would like to profoundly thank our Management of RIT & our President **Dr. Rajeev** for providing such a healthy environment.

We would like to express our sincere thanks to our principal **Dr. A.N Ramakrishna**, Rajeev Institute of Technology for his encouragement.

We wish to express our gratitude to **Dr. H.N Prakash**, Head of the Department of CSE for providing a good working environment and for his constant support and encouragement.

It gives us great pleasure to express our gratitude to **Mr. Dhruva M.S** Assistant Professor, Department of CSE for his expert guidance, initiative and encouragement that led us to complete this project.

We would also like to thank all our staffs of Computer Science and Engineering department who have directly or indirectly helped us in the successful completion of this project and also we would like to thank our parents.

AKASH C P
B S SURABHI
MOHANKUMAR M C
POORNIMA H P

ABSTRACT

The major goal of the project is to make traffic management system work dynamically using Internet of Things, Infrared sensor in order to make traffic system work efficiently. Traffic management automation systems in the market aims to computerized the traffic lights operates on the periodic to control the light (red / green) uses various technologies like GSM, NFC focuses on the basic operation of an electrical switch.

Our project plan to provide an automated IR-sense based solution that makes traffic signals to shift the lights (red / green) dynamically. We plan on implementing the project for one junction “proof-of-concept”, which includes traffic lights, IR-sensors, Wi-Fi transmitter and Raspberry Pi microcontroller. The sensed data gathered from IR sensor is transmitted by the Wi-Fi transmitter which is received by the raspberry-pi controller. Based on this compilation it dynamically shifts time of the red signal and user gets an intimation of status of the signal on his way. The Raspberry Pi controller works as a central console, it determines which sideways of the road signal is to get open or close. The central console gathers all the data from sensors and stores it in the cloud which intimates traffic status to a mobile device and finally providing way to ambulance.

LIST OF FIGURES

CONTENTS

CHAPTER 1

INTRODUCTION

The Advanced Traffic Management System field is a primary subfield within the Intelligent Transportation System domain. The Advanced Traffic Management System view is a top-down management perspective that integrates technology primarily to improve the flow of vehicle traffic and improve safety. Real-time traffic data from cameras, speed sensors, etc. flows into a Transportation Management Center where it is integrated and processed (e.g. for incident detection) and may result in actions taken (e.g. traffic routing, DMS messages) to improve traffic flow. The National Intelligent Transportation System Architecture defines the following primary goals and metrics for the Intelligent Transportation System.

Advanced Traffic Management system is an integrated solution to manage highway traffic through real-time information collection, processing, analysis, and finally dissemination to the users, concerned agencies, and stakeholders. To ensure round the clock safety, it is of prime importance to provide real-time and precise information to users about the road condition, traffic situations, incidents, and weather conditions on the roadway. It is also important to make interventions for smooth, safe, and efficient traffic movement by providing rescue and relief to the users to avoid distress.

1.1 Existing System

Current traffic control techniques involving magnetic loop detectors buried in the road, infra-red and radar sensors on the side provide limited traffic and require separate systems for traffic counting and traffic surveillance. The fact which encouraged us to conduct this research is that in many cities of the world, a signal allocation is still based on the timer. The timer approach has a drawback that even when there is less traffic on a road, green signal is still allocated to the road till its timer value falls to 0 while traffic on another road which is more, faces red signal at that time which causes congestion and time loss to commutators. Most of the present systems are not automated and are prone to human errors.

They provide more information, combine both surveillance and traffic control technologies, are easily installed, and are scalable with progress in image processing techniques.

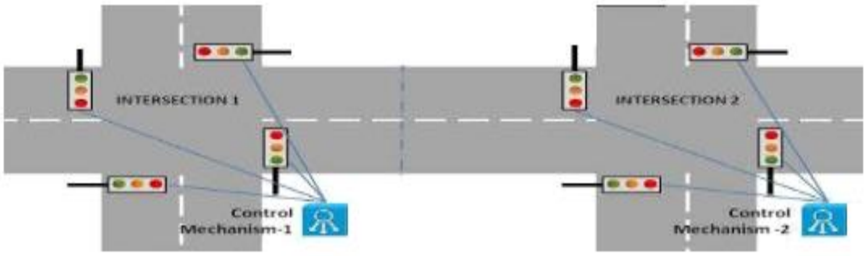

Fig 1.1: Block diagram of the existing traffic control system.

This paper tries to evaluate the process, advantages, and disadvantages of the use of image processing for traffic control. Implementation of our project will eliminate the need for traffic personnel at various junctions for regulating traffic. Thus, the use of this technology is valuable for the analysis and performance improvement of road traffic. Also, priority to emergency vehicles has been the topic of some research in the past.

1.2 Proposed System

A proposed system for the detection of these vehicles is based on Radio-Frequency Identification (RFID). However, the use of this technology necessitates unnecessary extra hardware to be installed both at every junction and in every vehicle. There have also been studies to recognize these vehicles by analysis of the count of the vehicles through sensors and the android application control system.

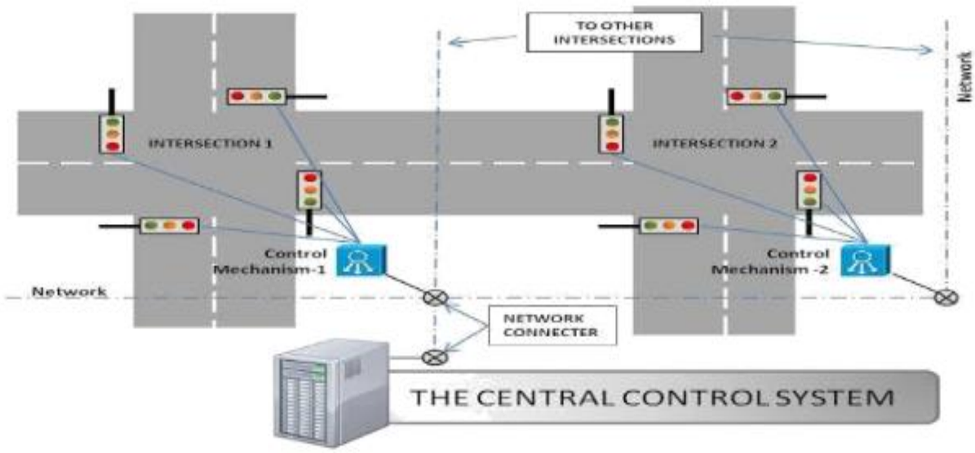

Fig 1.2: Block diagram of the proposed traffic control system.

However, this technology is also easily influenced by vehicle count and requires additional hardware at every traffic signal.

1.3 Problem Definition

The purpose of this project is to develop a series of system models for traffic passing through a four-way intersection controlled by a traffic light and road lanes is fixed and that lights switch from Red to Green to amber in a regular perspective pattern. Moreover, we assume that driver behavior constrained by road rules and the design is to avoid vehicle collisions, provide a way to an ambulance, and provide real-time traffic data to respective departments.

The fact which encouraged us to conduct this research is that in many cities of the world, a signal allocation is still based on the timer. The timer approach has a drawback that even when there is less traffic on a road, green signal is still allocated to the road till its timer value falls to 0 while traffic on another road which is more, faces red signal at that time which causes congestion and time loss to commutators. Most of the present systems are not automated and are prone to human errors.

The manual approach by traffic inspectors to give way to the ambulance and to avoid restarting the timer signal system each time after providing a way to the ambulance.

1.4 Objective of the Project

The main objective of this project is to design, to create a better road network system within the city for a smoother transition of traffic to increase the overall productivity of a city.

- **Smooth and Uninterrupted Traffic flow:** Traffic flow is the study of interactions between vehicles, drivers, and infrastructure, with the aim of understanding and developing an optimal road network with efficient movement of traffic and minimal traffic congestion problems.
- **Increase in transportation system efficiency:** The energy efficiency in transport is the useful traveled distance, of passengers, goods, or any type of load; divided by the total energy put into the transport propulsion means. The energy input might be rendered in several different types depending on the type of propulsion, and normally such energy is presented in liquid fuels, electrical energy, or food energy. The energy efficiency is also occasionally known as energy intensity. The inverse of the energy efficiency in transport is the energy consumption in transport.
- **Reduce Journey time and inconvenience:** The fact which encouraged us to conduct this research is that in many cities of the world, a signal allocation is still based on a timer. The timer approach has a drawback that even when there is less traffic on a road,

green signal is still allocated to the road till its timer value falls to 0 while traffic on another road which is more, faces red signal at that time which causes congestion and time loss to commutators. Most of the present systems are not automated and are prone to human errors.

- **Enhance Road Safety:** Road safety barriers provide many benefits. Here are some of them. Increased Safety on the Roads. The road safety barriers are specially designed to increase safety levels on the roads by protecting the drivers and vehicles in cases of accidents.
- **Smart Mobility:** The main objective of this project is to create a better road network system within the city for a smoother transition of traffic to increase the overall productivity of a city.

CHAPTER 2

LITERATURE SURVEY

Traffic congestion is a temporal condition on networks that occurs as utility increases, and is characterized by slower speeds, longer trip times, and increased queuing. When the volume of traffic is high and so heterogeneous that the interaction between vehicles slows down the speed of traffic, traffic congestion is the result. As demand approaches the capacity of a road (or of the intersections along the road), traffic congestion sets in. When vehicles are fully stopped for some time, this is colloquially known as a traffic jam.

2.1 EXISTING STUDIES ON RISK PARAMETERS

A simple model was developed by Jack Mallinckrodt, 2009 on regional average congestion delay, in a closed-form, the differentiable function of the regional transportation system with volume and capacity data. This model can be used to reduce the risk generated due to congestion. Different views were studied by Robert A. Johnston, Jay R. Lund, Paul P. Craig, 1995 on congestion generation and degeneration. Their study revealed that it is unlikely that roadway construction or vehicle automation will be able to alleviate most major urban congestion shortly i.e. for another 5 –15 years. Traffic congestion occurs when a volume of traffic or modal split generates demand for space greater than the available road capacity. Several specific circumstances cause or aggravate congestion most of them reduce the effective capacity of a road at a given point or over a certain length, or increase the number of vehicles required for a given volume of people or goods. Capacity allocation studies reveal that approaches like laissez-faire allocation, allocation by passenger load, ramp metering, road and parking pricing, allocation by trip purpose, rationing, and mixed strategies can be used for reducing congestion Quantification of congestion can be done by incorporating the volume and operational characteristics of traffic movement. Bhargab Maitra, P.K. Sikdar, and S.L. Dhingra, 1999 conducted a study on quantification of congestion. The quantified congestion level can be used as a logical and improved measure of effectiveness to account for the conceptual definition of the level of service quantitatively. Modeling congestion has provided a quantitative basis for understanding the contribution of different vehicle types in overall congestion, and it is useful for evolving the policy for congestion mitigation.

The principles of duration modeling can be used to find the extent of congestion. An approach is found for estimating the duration of congestion on a given road section and the probability that, given its onset, congestion will end during the identified period. It was studied by Anthony Stathopoulos and Mattew G. Karlaftis, 2001. Evaluating the efficacy of Intelligent Transportation Systems (ITS)technologies in reducing accidents that affect research development of models (such as incident delay and congestion models) that can accurately predict incident duration along with the magnitude of nonrecurring congestion, have been reported by A.Garib, A.E. Radwan and H.Al-Deek,1997. An analysis of freeway traffic flows under congestion was conducted by Do.H.Nam and Donald R.Drew, 1998, based on the principle of traffic dynamics, using the example of recurring congestion. Traditional incident-detection algorithms were developed by Chien-Hua Hsiao, Ching-Teng Lin, Michael Cassidy,2006 to distinguish between congested and uncongested operation by comparing measured traffic-stream parameters with predefined threshold values. Risk management is a key issue in project management. The first step in risk management is risk identification. It includes their cognition of potential risk causative factors and the clarification of risk. It was studied in detail by Ming-The Wang, Hui-yu Chou, 2006. Intelligent Transportation Systems are undergoing a transition from demonstration projects to becoming part of the mainstream set of options available to transportation planners. Hence, the evaluation of it is one of the most critical and important steps to be taken before any ITS technique can be deployed.

Safety has been recently emerging as an area of increased concerns, attention, and awareness within transportation engineering. Even though recent studies shed some light on driving speed factors as well as on the direction of the effects, knowledge is still insufficient to allow for specific quantifications. It was studied by Kai-ran Zhang, Guo-fangLi, 2007. Congestion leads to risk and finally may lead to accidents where urban accidents have the highest percentage impact (75%) over the entirety of accidents; therefore, they represent a crucial event which potentially may lead to disastrous consequences. Artificial Intelligence may help provide more powerful techniques to understand the main causes of accidents and congestion.

Nowadays urban transport planning is not taking into account the increasing number of motor vehicles, the growth of the city, or the environment. Given the enormous benefit to the health of every citizen, in terms of cost and terms, the benefits arising from inner-city accessibility one would think that network functionality and successive planning of public transport would be one of the provincial government’s primary areas of interest. The provincial government is interested, and committees are being set upend meetings are held.

But these seek to “solve” congestion when, in reality, it can only be dealt with and this approach to the issue, glaring oversights are common. For the present study, risk analysis has been proposed to be achieved through Principal component analysis followed by causal techniques to identify the factors contributing to risk generation and the major links which are leading to congestion.

2.2 GEOGRAPHIC INFORMATION SYSTEM (GIS)

GIS is a computer-based tool for mapping and analyzing things that exist and events that happen on earth. GIS technology integrates common database operations such as query and statistical analysis with the unique visualization and geographic analysis benefits offered by maps. These abilities distinguish GIS from other information systems and make it valuable to a wide range of public and private enterprises for explaining events, predicting outcomes, and planning strategies (Moses Santhakumar, 1998). GIS stores information about the world as a collection of thematic layers which can be linked together by geography. This is a simple but extremely powerful and versatile concept that has proven invaluable for solving many real-world problems from tracking delivery vehicles, to recording details of planning applications, to modeling global atmospheric circulation. GIS allows us to bring all types of data together based on the geographic and location component of the data. But unlike a static paper map, GIS can display many layers of information that are useful to us(www.esri.com). Using this, one will be able to integrate, visualize, manage, solve, and present the information in a new way.

Relationships between the data will become more apparent and the data will become more valuable. GIS gives us the power to create maps, integrate information, visualize scenarios, solve complicated problems, present powerful ideas, and develop effective solutions like never before.

2.3 STUDY OF RISK CRETERIONS

As discussed in the previous chapters, to identify the exact criteria and the links which lead to congestion and to study which are the real factors of risk generation, certain criteria have been identified. The broad criteria are categorized into four groups –Geometric characteristics, Traffic characteristics, Land use or roadside characteristics, and Utility characteristics. The attributes considered under each criterion are as follows.

2.3.1 Geometric characteristics

Geometric characteristics represent the geometric features of the roadway affecting the Level of service of the link. These are the static characteristics of the road infrastructure.

- Roadway width in meters (RW).
- Carriageway width in meters (CW).
- Stopping sight distance in meters (SSD).
- The number of curves on the link (NC).

Pavement Condition index determined from the rating of the pavement based on the pavement condition and riding comfort experienced by the user on a scale of 1 to 5, 5 being excellent pavement and 0 being an impassable pavement (PCI). The pavement condition index is quality and comfort promoted on a pavement with a defined mobility speed of 20 kmph.

The pavement condition is assessed with cumulative bumps that occurred in the travel with a bump integrator machine. If the bump depth in a one km length is between 0 to 50 mm it is ranked as 5 and beyond 250mm per km itis ranked as 0. The transitional ranking of 4,3,2,1 is given in proportion in the range of bump depth 50 and 250mm. The geometrics of the highway should be designed to provide optimum efficiency in traffic operations with maximum safety at a reasonable cost. The overall design of geometrics of a highway is a function of the design speed. Geometric attributes are studied from static data. They are collected from satellite data by using GPS and GIS as supportive tools and field survey data.

2.3.2 Traffic characteristics

Traffic characteristics are the dynamic characteristics of the road that influence the level of service of the link. They are

- Headway in seconds (H)
- V/C Ratio (VCR)
- The intensity of Parking (PBE), business activities, and roadside activities encroachments on a point scale.
- Speed in kmph (V)
- Delay in seconds (D)

The traffic characteristics are quite complex with various types of road users in the roads moving with different motives. The study of vehicular characteristics is an essential part and the various studies to be carried on the actual traffic include traffic flow characteristics.

2.3.3 Traffic studies

Traffic studies or surveys are carried out to analyze the traffic characteristics. These studies help in deciding the geometric design features and traffic control for safe and efficient traffic movements. The traffic surveys for collecting traffic data are also called traffic census. The various traffic studies generally carried out are.

Traffic Volume Study - One of the fundamental measures of traffic on a road system is the volume of traffic using the road in a given interval of time. It is also termed as flow and it is expressed in vehicles per hour or vehicles per day. When the traffic is composed of some types of vehicles, itis the normal practice to convert the flow into an equivalent passenger car unit (PCUs) by using certain equivalency factors. The flow is then expressed as PCUs per hour or PCUs per day. Manual methods use field personnel to count and classify traffic flowing past a fixed point. Some of the advantages of manual methods and situations where these are to be preferred are

- Details such as vehicle classification and several occupants can be easily obtained.
- The data can be collected giving the breakdown of traffic in each direction of travel.

It is more desirable to record the traffic in both directions of travel separately and post separate observations for each direction. For all-day counts, work in three shifts of 8 hours each could be organized. A separate observer is needed if the occupancy count is to be made.

Speed Survey - Speed is one of the most important characteristics of traffic and its measurements are a frequent necessity. All vehicles do not travel at the same speed at a location along a road. The amount of speed dispersion or the spread from the average speed affects both capacity and safety. The actual speed of a vehicle over a particular route may be fluctuating widely depending on several factors such as geometric features, traffic conditions, time, place, environment, and driver. Speed studies carried out occasionally give the general trend in speeds.

Speed and Delay Study - The speed and delay studies give the running speeds, overall speeds, fluctuations in speeds and the delay between two stations of a road spaced far apart. They also give information such as the amount, location, duration frequency, and causes of the delay in the traffic stream. The results of the speed and delay studies are useful in determining the spots of congestion, the cause, and in arriving at a suitable remedial measure.

Delay Studies - Delay studies along routes are best done by the moving observer method described earlier. The delays occurring due to stop can be conveniently recorded by a separate stopwatch. Special watches that can accumulate the delay time as the observer operates buttons will be found convenient for this purpose. The delays that can be measured thus are stopped delays or fixed delays that occur at intersections, railway crossing, and stop signs.

2.3.4 Land use or Roadside characteristics

These are also the static elements of the link which influence the operational efficiency of the link. They are:

- Number of access points on the link (NA)
- Commercial area along the roadside of the link in sq. km (CA)
- The residential area along the roadside of the link in sq.km (RA)
- Semi Residential area along the roadside of the link in sq. km (SRA)
- The industrial area along the roadside of the link in sq. km (IA)

These are static elements of the link which influence the operational efficiency of the link. The data is collected from satellite, field, and municipality authorities.

2.3.5 Utility characteristics

Utility characteristics are the characteristics of the link indicating the degree of utility of the link concerning the static analysis and dynamic analysis.

- Overlap size of the link from static analysis (OS)
- Trip intensity on the link (TI) in trips/day

Utility characteristics of the link indicate the degree of utility of the link concerning the static and dynamic analysis. Utility characteristics are collected from the field and OD survey.

CHAPTER 3

INTRODUCTION TO IOT

Internet of Things (IoT) has not been around for very long. However, there have been visions of machines communicating with one another since the early 1800s. Machines have been providing direct communications since the telegraph (the first landline) was developed in the 1830s and 1840s. Described as “wireless telegraphy”, the first radio voice transmission took place on June 3, 1900, providing another necessary component for developing the Internet of Things. The development of computers began in the 1950s.

The Internet, itself a significant component of the IoT, started as part of DARPA (Defense Advanced Research Projects Agency) in 1962 and evolved into ARPANET in 1969. In the 1980s, commercial service providers began supporting public use of ARPANET, allowing it to evolve into our modern internet. Global Positioning Satellites (GPS) became a reality in early 1993, with the Department of Defense providing a stable, highly functional system of 24 satellites. This was quickly followed by privately owned, commercial satellites being placed in orbit. Satellites and landlines provide basic communications for much of the IoT.

The Internet of Things, as a concept, wasn’t officially named until 1999. One of the first examples of an IoT is from the early 1980s and was a Coca-Cola machine, located at the Carnegie Melon University. By the year 2013, the IoT had evolved onto a system using multiple technologies, ranging from the Internet to wireless communication and from micro-electromechanical systems (MEMS) to embedded systems.

3.1 Internet of Things (IoT)

Internet of Things (IoT) is the network of physical devices, vehicles, home appliances, and other items embedded with electronics, software, sensors, actuators, and connectivity which enables these things to connect, collect and exchange data.

Fig 3.1: Internet of Things.

IoT involves extending Internet connectivity beyond standard devices, such as desktops, laptops, smartphones, and tablets, to any range of traditionally dumb or non-internet-enabled physical devices and everyday objects. Embedded with technology, these devices can communicate and interact over the Internet, and they can be remotely monitored and controlled. With the arrival of driverless vehicles, a branch of IoT, i.e. the Internet of Vehicles starts to gain more attention.

3.2 Architecture of IoT

Figure 1.2 shows the Architecture of IoT. There are four layers in IoT architecture such as the Interface layer, Service layer, networking layer, Sensing layer.

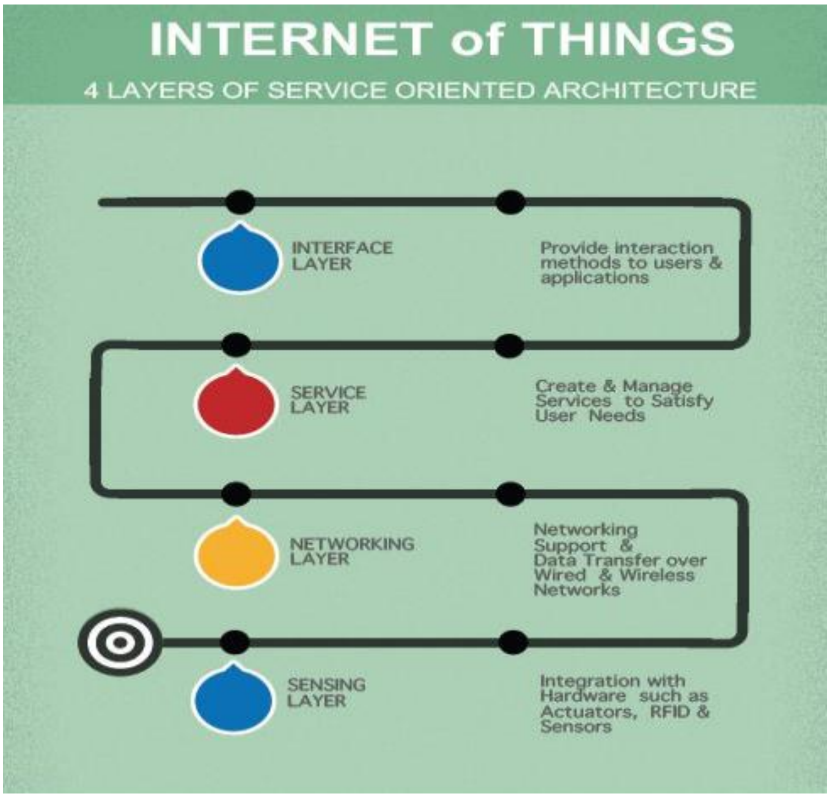

Fig 3.2: IoT Basic layer architecture.

Interface Layer - The first layer of IoT is the interface layer. This layer provides the interaction methods between users and applications. This section looks at how users can easily use the system. This includes three main approaches. Firstly, we need the ability to create web-based front-ends and portals that interact with devices and with the event-processing layer. Secondly, we need the ability to create dashboards that offer views into analytics and 11 events processing. Finally, we need to be able to interact with systems outside this network using machine-to-machine communications (APIs). The recommended approach to building the web front end is to utilize a modular frontend architecture, Web server-side technology, such as Java Servlets, JSP, PHP, Python, Ruby, etc.

Service layer - This layer is used to create and manage services to satisfy the user's needs. To do so, it processes data deep processing. To make a more user-friendly application, it provides a database with different data and divides work.

This is an important layer for three reasons:

- The ability to support an HTTP server and/or an MQTT broker to talk to the devices.
- The ability to aggregate and combine communications from different sensing devices and to route communications to a specific device (possibly via GSM/GPRS).
- The ability to bridge and transform between different protocols that is to offer HTTP based APIs that are mediated into an MQTT message going to the device.

Networking or Communication Layer - The Networking or Communication layer supports the connectivity of the devices. There are multiple potential protocols for communication between the devices and the cloud. The most well-known three potential protocols are HTTP/HTTPS.

3.3 Applications of IoT

The Internet of things (IoT) is the between systems administration of physical gadgets, vehicles, structures, and different things installed with hardware, programming, sensors, actuators, and system availability which empower these items to gather and exchange data.

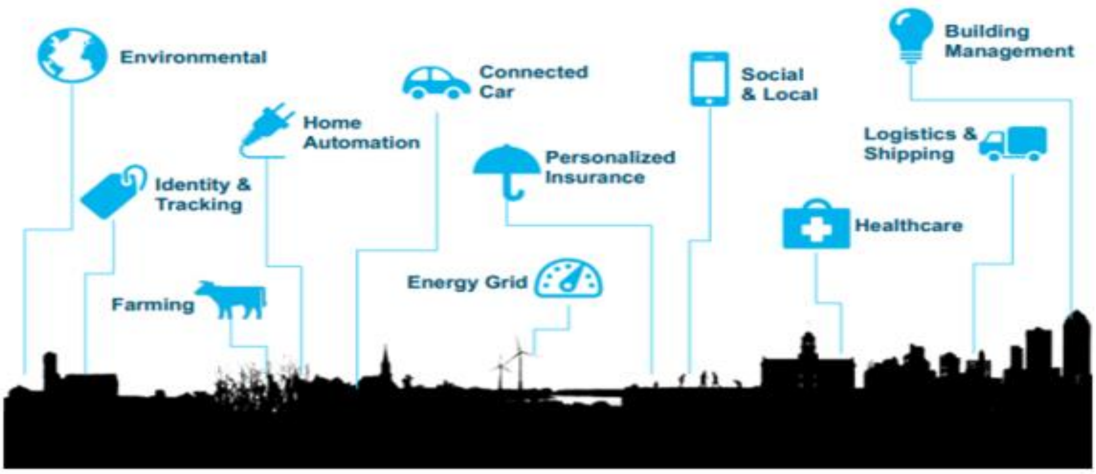

Fig 3.3: IoT connecting the physical world to the web.

The IoT enables items to be detected or controlled remotely across existing system infrastructure creating open doors for more straightforward of the physical world into computer-based systems and resulting in improved efficiency, accuracy, and economic benefit in addition to reduced human intervention. Only IoT can connect the physical world to the web.

Environmental:

The IoT is more than internet-connected consumer gadgets. Sooner or later every IT organization will need to create a framework to support it. Energy companies already use networked sensors to measure vibrations in turbines. They feed that data through the network to computing systems that analyze it to predict when machines will need maintenance and when they will fail. Jet engine manufacturers embed sensors that measure temperature, pressure, and other conditions to improve their products. Even a gift basket business can deploy sensors to constantly monitor the temperature of perishable products.

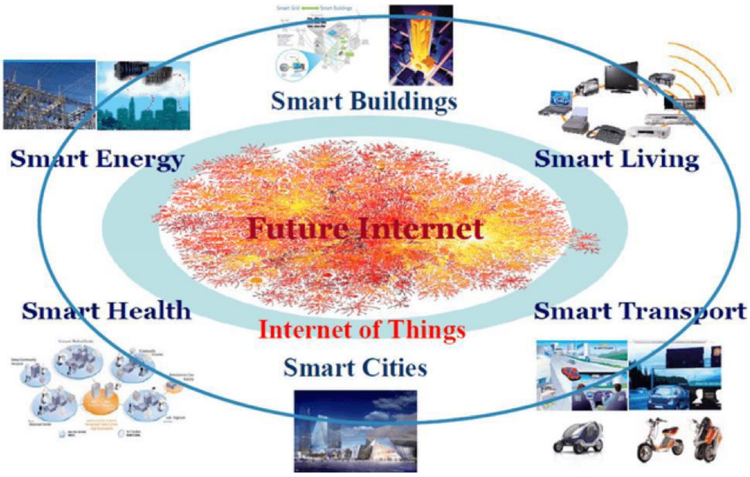

Fig 3.4: Environmental Monitoring with IoT.

Internet of Things (IoT) has a large role to play in the future of smart cities. IoT can be used in practically all scenarios for public services by governments to make cities environment friendly. Sensor-enabled devices can help monitor the environmental impact on cities, collect details about sewers, air quality, and garbage. Such devices can also help monitor woods, rivers, lakes, and oceans. Many environmental trends are so complex that they are difficult to conceptualize. IoT environmental monitoring applications usually use sensors to lend a hand in environmental protection by monitoring air or water quality, atmospheric or soil conditions, and can even include areas like monitoring the movements of wildlife and their habitats.

Medical and HealthCare:

The Internet of Medical Things (also called the internet of health things) is an application of the IoT for medical and health-related purposes, data collection and analysis for research, and monitoring. This "Smart Healthcare", as it can also be called, led to the creation of a digitized healthcare system, connecting available medical resources and healthcare services.

Fig 3.5: HealthCare Advantages using IoT

IoT devices can be used to enable remote health monitoring and emergency notification systems. These health monitoring devices can range from blood pressure and heart rate monitors to advanced devices capable of monitoring specialized implants, such as pacemakers, it can also adjust itself to ensure appropriate pressure and support is applied to the patient without the manual interaction of nurses. Moreover, the use of mobile devices to support medical follow-up led to the creation of „m-health", used "to analyze, capture, transmit and store health statistics from multiple resources, including sensors and other biomedical acquisition systems".

Transportation:

The IoT can assist in the integration of communications, control, and information processing across various transportation systems. Application of the IoT extends to all aspects of transportation systems (i.e. the vehicle, the infrastructure, and the driver or user). Dynamic interaction between these components of a transport system enables inter and intra vehicular communication, smart traffic control, smart parking, electronic toll collection systems, logistics and fleet management, vehicle control, and safety and road assistance.

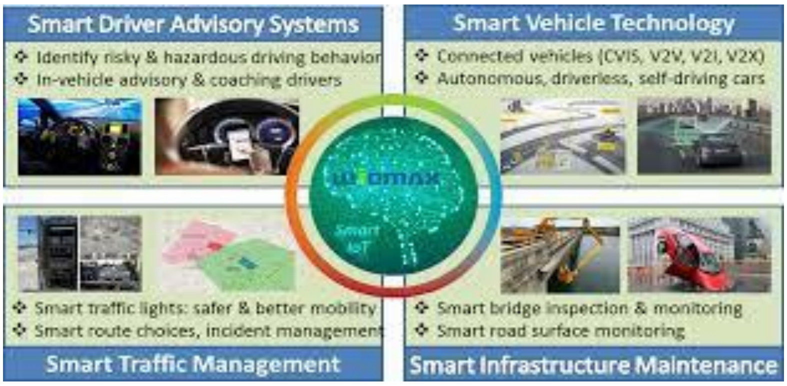

Fig 3.6: Transportation using IoT

In Logistics and Fleet Management, for example, The IoT platform can continuously monitor the location and conditions of cargo and assets via wireless sensors and send specific alerts when management exceptions occur (delays, damages, thefts, etc.). If combined with Machine Learning then it also helps in reducing traffic accidents by introducing drowsiness alerts to drivers and providing self-driven cars too.

Home automation:

The entire world is evolving with new technologies and IoT is the current trend. Not all, but those who are aware of it, are looking forward to home automation using IoT. Many IoT solution providers can help you experience IoT based on home automation. The rise of Wi-Fi's role in home automation has primarily come about due to the networked nature of deployed electronics where electronic devices have started becoming part of the home IP network and due to the increasing rate adoption of mobile computing devices.

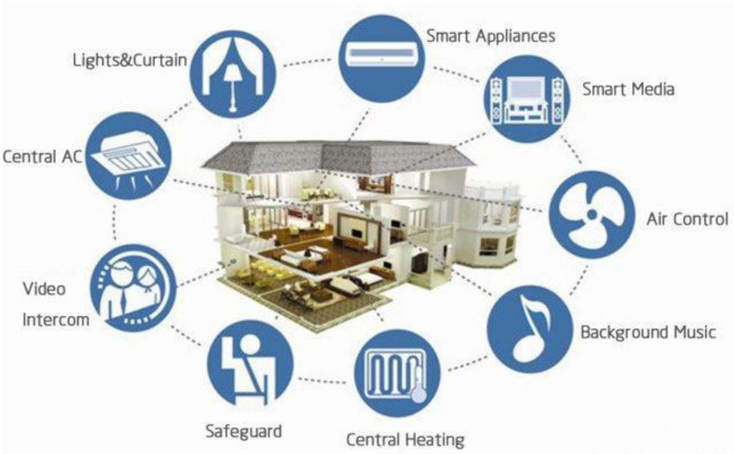

Fig 3.7: Smart Home using IoT

A smart home or automated home could be based on a platform or hub that control smart devices and appliances. For instance, using Apple's Home Kit, manufacturers can get their home products and accessories be controlled by an application in iOS devices such as the iPhone and the Apple Watch. This could be a dedicated app or iOS native applications such as Siri. This can be demonstrated in the case of Lenovo's Smart Home Essentials, which is a line of smart home devices that are controlled through Apple's Home app or Siri without the need for a Wi-Fi bridge. There are also dedicated smart home hubs that are offered as standalone platforms to connect different smart home products and these include the Amazon Echo, Apple's Home Pod, and Samsung's Smart Things Hub.

Social and Local:

The misconception that the younger generation is anti-social because their noses are buried in their smartphones doesn’t support their reasoning for attending live events. One thing they are used to though is personalized and localized information.

3.4 Advantages of IoT

Data: The more the information, the easier it is to make the right decision. Knowing what to get from the grocery while you are out, without having to check on your own, not only saves time but is convenient as well.

Tracking: The computers keep track of both the quality and the viability of things at home. Knowing the expiration date of products before one consumes them improves the safety and quality of life. Also, you will never run out of anything when you need it at the last moment.

Time: The amount of time saved in monitoring and the number of trips done otherwise would be tremendous.

Money: The financial aspect is the best advantage. This technology could replace humans who are in charge of monitoring and maintaining supplies.

CHAPTER 4

INTRODUCTION TO ANDROID

Android is a software platform and operating system for the mobile device based on the Linux operating system and developed by Google and the Open Handset Alliance. The unveiling of the Android platform on 5 November 2007 was announced with the founding of the Open Handset Alliance, a consortium of 34 hardware, software, and telecom companies devoted to advancing open standards for mobile devices. When released in 2008, most of the Android platform will be made available under the Apache free-software and open-source license.

4.1 Android Framework

Android is a software stack for mobile devices that includes an operating system, middleware, and key applications. It allows developers to write managed code in Java-like language that utilizes Google-developed Java libraries but does not support programs developed in native code.

4.1.1 The Birth of Android

Google acquires Android Inc in July 2005, Google acquired Android Inc., a small startup company based in Palo Alto, CA. Androids co-founders who went to work at Google included Andy Rubin (co-founder of Danger), Rich Miner(co-founder of wildfire communication, Int), Nick Sears (once VP at T-Mobile) and Chris White(one of the first engineers at WebTV). At the time, little was unknown about the functions of Android Inc. Other than they made software for mobile phones.

At Google, the team led by Rubin developed a Linux-based mobile device of which they marketed to handset markets and carries on the premise of providing an edible, upgradable system. It was reported that Google had already lined up a series of hardware components and software partners and signaled to carriers that it was open to various degrees of cooperation on their part.

Open Handset Alliance Founded on 5 November 2007, the Open Handset Alliance (OHA)a consortium of several companies which include GOOGLE, HTC, Intel, Motorola, Qualcomm, T-Mobile, Sprint Nextel, and NVIDIA, was unveiled to develop open standards for mobile devices. Along with the formation of the Open Handset Alliance, the OHA also unveiled their first product, Android an open-source mobile device platform based on the Linux operating system.

Hardware: Google has unveiled at least three prototypes for Android, the Mobile Congress on February 12, 2008. One prototype at the ARM both displayed several basic Google applications. A d-Pad control zooming of items in the dock with a relatively quick application. A prototype at the Google IO conference on May 28th,2008 had a 528MHz QUALCOMM processor and a synaptic capacities touch screen and used the UMTS cellular standard. It had 128 MB of RAM and 256 MB of ash, showing that Android's memory requirements are reasonable. The Demo was carried out using a 3.6 Mb HSDPA connection.

4.1.2 Features

Android includes many useful features, some of them are listed below

- **Application framework:** It is used to write applications for Android. Unlike other embedded mobile environments, Android applications are equal, for instance, which comes with the phone are no different than those that any developer writes. The framework is supported by numerous open-source libraries such as open SSL, SQLite, and Llibc. It is also supported by the Android core libraries. From the point of security, the framework is based on UNIX file system permissions that assure applications have only those abilities. That mobile phone owner gave them at install time.
- **Disk Virtual Machine:** It is extremely low-memory based virtual machine, which was designed especially for Android to run on an embedded system and work well in low power situations. It also tunes to the CPU attributes. The Dalvik VM creates a special _le format(.DEX) that is created through build time post-processing. Conversion between Java classes and. DEX format is done by including the dex tool.
- **Integrated Browser:** The possibility of linkage to YouTube in case if users are not satisfied with the information provided in the form of text and images. Users have an option for choosing the browser in which they want to watch the video. Google made the right choice on choosing Web Kit as an open-source web browser. They added a two-pass layout and frame attiring. Two-pass layout loads a page without waiting for blocking elements, such as external CSS or external JavaScript and after a while renders again with all resources downloaded to the device. Frame altering converts founded frames into a single one and loads into the browser. These features increase speed and usability browsing the internet via mobile phones.

- **Optimized Graphics:** An Android has 2D graphics libraries and 3D graphics based in OpenGL ES 1.0, possibly we will see great applications like Google Earth and spectacular like Second Life which comes on Linux version. At this moment, the shooting legendry 3D game Doom was presented using Android on the mobile phones.
- **SQLite:** Extremely small (500KB) relation able database management system, which is integrated into Android. Is based on function calls and single file, where all definitions, tables, and data are stored. This simple design is more than suitable for a platform such as Android. SQLite provides good database storage for Android applications.
- **Handset Layouts:** The platform is adaptable for both larger VGA, 2D graphics library, 3D graphics library based on OpenGL Es 1.0 specifications, traditional smartphone layouts. An underlying 2D graphics engine is also included. Surface Manager manages access to the display subsystem and seamlessly composites 2D and 3D graphic layers from multiple applications.
- **Data Storage:** SQLite is used for structured data storage; SQLite is a powerful and lightweight relational database engine available to all applications.
- **Connectivity:** Android supports a wide variety of connectivity technologies including GSM, CDMA, Bluetooth, EDGE, EVDO, 3G, and Wi-Fi.
- **Messaging:** SMS, MMS, and XAMPP are available forms of messaging including threaded text messaging.
- **Web Browser:** The web browser available in Android is based on the open-source Web Kit applications framework. It includes LibWebCore which is a modern web browser engine which powers both the Android web browser and an embedded able web view.
- **Java Virtual Machine:** Software written in Java can be compiled into Dalvik byte codes and executed in the Dalvi virtual machine, which is a specialized VM implementation designed for mobile device use, although not technically a standard Java Virtual Machine.
- **Media Support:** Android will support advanced audio/video/still media formats such as MPEG-4, h.264, MP, AAC, AMR, JPEG, GIF, PNG.
- **Additional Hardware Support:** Android is fully capable of utilizing video/still cameras, touch screens, GPS, compasses, accelerometer, and accelerator 3D graphics.
- **Development Environment:** Includes a device emulator, tools for debugging, memory and performing profiling, a Plugin for the Eclipse IDE. There are some hardware-dependent features, for instance, a huge media and connections support, GPS, improved

camera support, and simply GSM telephony. Great work was done for the developers to start work with Android using device emulator, tools for debugging, and plugin for Eclipse IDE.

4.2 Architecture

Figure 4.1 shows the major components of the Android operating system. Each section is described in more detail below.

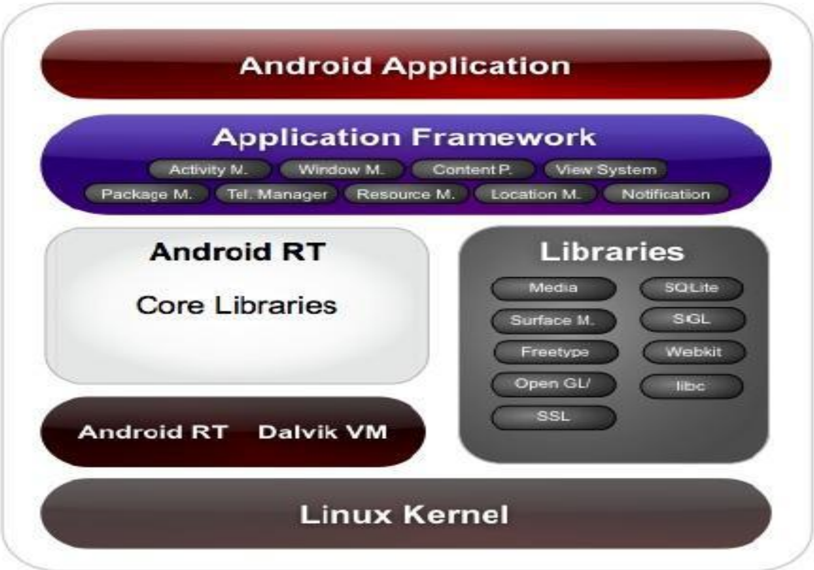

Figure 4.1: Architecture of Android

4.2.1 Linux Kernel

Android Architecture is based on the Linux 2.6 kernel. It helps to manage security, memory management, process management, network stack, and other important issues. Therefore, the user should bring Linux in his mobile as the main operating system and install all the drivers required to run it. Android provides the support for the Qualcomm MSM7K chipset quality. For instance, the current kernel tree supports Qualcomm MSM7200A chipsets, but in the second half of 2008, we should see mobile devices with stable version Qualcomm MSM 7200, which includes major features.

- WCDMA/HSUPA and EGPRS network supports,
- Bluetooth 1.2 and Wi-Fi support,
- Digital audio support for mp3 and other formats,
- Support for Linux and other third-party operating systems,
- Java Hardware acceleration and support for Java applications,

- Q camera up to 6.0 megapixels,
- Gps One solution for GPS,
- and lots of others.

4.2.2 Libraries

In the next level, there are a set of native libraries written in C/C++, which are responsible for stable performance no various components. For example, the Surface Manager is responsible for composing different drawing surfaces on the mobile screen. It manages access for different processes to compose 2D and 3D graphics layers. OpenGL ES and SGL make a core of graphics libraries and are used accordingly for 3D and 2D hardware acceleration. Moreover, it is possible to use 2D and 3D graphics in the same application in Android. The media framework was provided by pocket Video, one of the members of OHA. It gives libraries for a playback and recording support for all the bitmap and vector fonts. For data storage, Android uses SQLite.

As mentioned before, it is an extra light rational management system, which locates a single file for all operations related to the database. Web Kit, the same browser used by Apples Safari, was modified by Android to fit better in a small size screen. Android includes a set of C/C++ libraries used by various companies of the Android system. These capabilities are exposed to developers through the Android application framework. Some of the core libraries are listed below.

- System C library-a BSD-derived implementation of the standards C system library (libs), tuned for embedded Linux-based devices.
- Media Libraries-based on Packet Videos Open CORE, the libraries support playback and recording of many popular audio and video formats. As well as static image files, including MPEG4, H.264, MP3, AAC, AMR, JPG and PNG.
- Surface Manager - manages access to the display subsystem and seamlessly composite2D and 3D graphics layers from multiple applications.
- LibWebCore - a modern web browser engine which powers both the Android browser and an embeddable web view.
- SGL - the underlying 2D graphics engine.
- 3D libraries - an implementation based on OpenGL ES 1.0 APIs; the libraries use either hardware 3D acceleration or the included, highly optimized 3D software

- Free Type - bitmap and vector font rendering
- SQLite - a powerful and lightweight relational database engine available to all applications.

4.2.3 Android Runtime

At the same level, there is Android runtime, where the main component Virtual machine is located. It was designed especially for Android running in a limited environment, where the limited battery, CPU, memory, and data storage are the main issues, Android gives an integrated tool, which converts generated byte from .jar to .dex file after this byte code becomes much more efficient to run on the small processors. As a result, it is possible to have multiple instances of Dalvik Virtual Machine running on a single device at the same time. The Core libraries are written in Java language and contains the collection classes, the utilities, IO, and other tools.

4.2.4 Application Framework

After that, there is an Application Framework, written in Java language. It is a toolkit that all applications use, ones which come with mobile devices like contacts or SMS box, or applications written by Google and any Android developer. It has several components.

- The Activity Manager manages the life cycle of the applications and provides a common navigation back stack for applications, which are running in different processes.
- The Package Manager keeps track of the applications, which are installed in the device.
- The Windows Manager is Java programming language abstraction on the top of lower-level services that are provided by the Surface Manager.
- The Telephony Manager contains a set of APIs necessary for calling applications.
- Content Providers were built for Android to share data with other applications, for instance, the contacts of people in the address books can be used in another application tool.
- The Resources Manager is used to store localized strings, bitmaps, layout descriptions and other external parts of the applications.
- The View systems generate a set of buttons and lists used in UI.
- Other components like the Notification manager is used to customize display alerts and other functions.

4.2.5 Application Layer

At the top of Android Architecture, it has all the applications, which are used by the final user. By installing different applications, the user can turn his mobile phone into unique, optimized, and smart mobile phones. All applications are written using the Java Programming language.

4.3 Application Building Blocks

The major building blocks are

- Activity
- Intent Receiver
- Service
- Content Provider
- AndroidMainfest.xml

Activity: User interface component, which corresponds to one screen at a time. It means that for simple applications like Address Book, the developer should have one activity for displaying contacts, another activity component for displaying more detailed information of the chosen name, etc.

Intent Receiver: Wakes up a predefined action through the external event. For example, for the application like email, inbox, the developer should have an intent receiver and register his code through XML to wake up an alarm notification, when the user receives an email.

Services: A task, which is done in the background. It means that the user can start an application from the activity window and keep the service work while browsing other applications. For instance, he can browse Google Maps application while holding a call or listening to music while browsing other applications.

Content Provider: A component, which allows sharing some of the data with other processes and applications. Is the best way to communicate the application between each other Android will ship with a set of core application including an email client, SMS program, calendar, maps, browser, contacts and others. All applications are written using the java Programming language.

AndroidMainfest.xml: The AndroidMainfest.xml file is the control file that tells the system what with all the top-level components (specifically activities, service, intent receivers, and content providers described above) that are created. A developer should predefine and list all components to be used in the specific Android Mainfest.xml file. It is a required file for all the applications and is located in the root folder. It is possible to specify all global values for the package, all the components and its classes used, intent filters, which describe where and when the certain activity should start, permissions, and instruments like security control and testing.

4.4 Summary of Android OS

Android received a lukewarm reaction when it was unveiled in 2007. Although analysts were impressed with the respected technology companies that had partnered with Google to form the Open Handset Alliance, it was unclear whether mobile phone manufacturers would be willing to replace their existing operating systems with Android. The idea of an open-source, Linux-based development platform sparked interest, but there were additional worries about Android facing strong competition from established players in the smartphone market, such as Nokia and Microsoft, and rival Linux mobile operating systems that were in development. Since then Android has grown to become the most widely used smartphone operating system and "one of the fastest mobile experiences available." The openness and flexibility are also present at the level of the end-user: Android allows extensive customization of devices by their owners and apps are freely available from non-Google app stores and third-party websites. These have been cited as among the main advantages of Android phones over others. Despite android's popularity, including an activation rate three times that of iOS, there have been reports that Google has not been able to leverage its other products and web services successfully to turn Android into the money maker that analysts had expected. The Verge suggested that Google is losing control of Android due to the extensive customization and proliferation of non-Google apps and services Amazon's Kindle Fire line uses Fire OS, a heavily modified fork of Android which does not include or support any of Google's proprietary components, and requires that users obtain software from its competing Amazon App store instead of Play Store. In 2014, to improve the prominence of the Android brand, Google began to require that devices featuring its proprietary components display an Android logo on the boot screen.

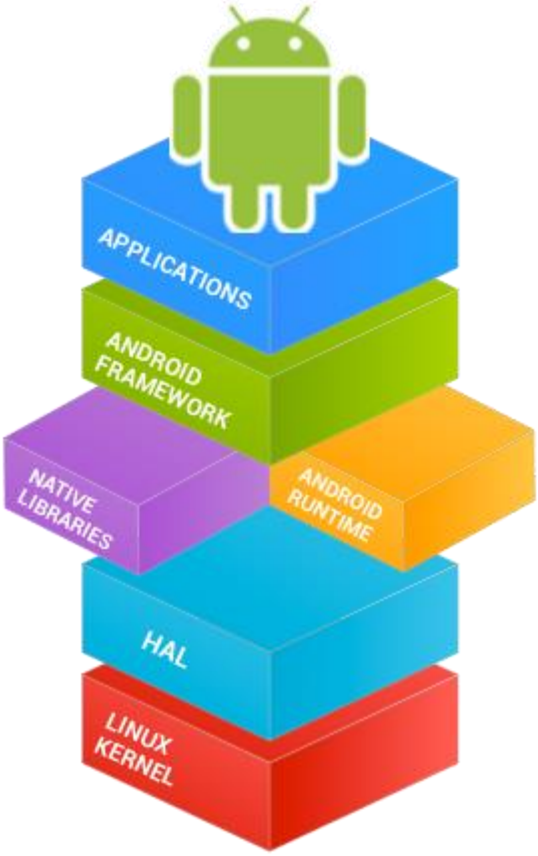

Fig 4.2: The stack of Android Open Source Project.

Android has suffered from "fragmentation", a situation where the variety of Android devices, in terms of both hardware variations and differences in the software running on them, makes the task of developing applications that work consistently across the ecosystem harder than rival platforms such as iOS where hardware and software vary less. Critics such as Apple Insider have asserted that fragmentation via hardware and software pushed Android's growth through large volumes of low end, budget-priced devices running older versions of Android. They maintain this forces Android developers to write for the "lowest common denominator to reach as many users as possible, who have too little incentive to make use of the latest hardware or software features only available on a smaller percentage of devices. However, Open Signal, who develops both Android and iOS apps, concluded that although fragmentation can make development trickier, Android's wider global reach also increases the potential reward.

CHAPTER 5

SYSTEM REQUIREMENTS

Whenever we purchase software or hardware for our computer, we should first make sure our computer supports the system requirements. All computer software needs certain hardware components or other software resources to be present on a computer. These prerequisites are known as system requirements and are often used as a guideline as opposed to an absolute rule. Most software defines two sets of system requirements minimum and recommended. With the increasing demand for higher processing power and resources in newer versions of software, system requirements tend to increase over time.

The requirements can be broken into two major categories namely hardware and software requirements. The formal specifications the minimal hardware facilities expected in a system in which the project has to be run. The latter specifies the essential software needed to build and run the project.

5.1 Hardware Requirements

The most common set of requirements defined by any operating system or software application is the physical computer resources, also known as hardware, A hardware requirements list is often accompanied by hardware compatibility(HCL).

5.1.1 Raspberry Pi 3 B+

The Raspberry Pi 3 Model B+ is the final revision in the Raspberry Pi 3 range. Broadcom BCM2837B0, Cortex-A53 (ARMv8) 64-bit SoC @ 1.4GHz. 1GB LPDDR2 SDRAM. 2.4GHz and 5GHz IEEE 802.11.b/g/n/ac wireless LAN, Bluetooth 4.2, BLE.

The Raspberry Pi is a low cost, credit-card sized computer that plugs into a computer monitor or TV, and uses a standard keyboard and mouse. It is a capable little device that enables people of all ages to explore computing and to learn how to program in languages like Scratch and Python.

The Raspberry Pi 3 Model B+ is the latest production Raspberry Pi 3 featuring a 64-bit quad-core processor running at 1.4 GHz. This power supply can power the Raspberry Pi 3 B+ and other USB accessories (such as most portable hard drives) without requiring an externally powered USB hub.

Fig 5.1: RASPBERRY PI 3 B+ microprocessor board.

The Raspberry Pi is a low cost, credit-card sized computer that plugs into a computer monitor or TV, and uses a standard keyboard and mouse. It is a capable little device that enables people of all ages to explore computing and to learn how to program in languages like Scratch and Python. It's capable of doing everything you'd expect a desktop computer to do, from browsing the internet and playing high-definition video, to making spreadsheets, word-processing, and playing games.

What's more, the Raspberry Pi can interact with the outside world and has been used in a wide array of digital maker projects, from music machines and parent detectors to weather stations and tweeting birdhouses with infra-red cameras. We want to see the Raspberry Pi being used by kids all over the world to learn to program and understand how computers work,

The raspberry pi boards are used in many applications like Media streamer, Arcade machine, Tablet computer, Home automation, Carputer, Internet radio, controlling robots, Cosmic Computer, hunting for meteorites, Coffee and also in raspberry pi-based projects.

5.1.2 IR Sensors

An infrared sensor is an electronic device, that emits to sense some aspects of the surroundings. An IR sensor can measure the heat of an object as well as detects the motion. These types of sensors measure only infrared radiation, rather than emitting it that is called a passive IR sensor.

Fig 5.2: IR Sensors.

IR LED Transmitter: IR LED emits light, in the range of Infrared frequency. IR light is invisible to us as its wavelength (700nm – 1mm) is much higher than the visible light range. IR LEDs have a light-emitting angle of approx. 20-60 degree and range of approx. few centimeters to several feet, it depends upon the type of IR transmitter and the manufacturer. Some transmitters have the range in kilometers. IR LED white or transparent in color, so it can give out the amount of maximum light.

Photodiode Receiver: Photodiode acts as the IR receiver as its conduct when light falls on it. The photodiode is a semiconductor that has a P-N junction, operated in Reverse Bias, means it start conducting the current in the reverse direction when Light falls on it, and the amount of current flow is proportional to the amount of Light. This property makes it useful for IR detection. Photodiode looks like a LED, with a black color coating on its outer side, Black color absorbs the highest amount of light.

Variable Resistor: The variable resistor used here is a present. It is used to calibrate the distance range at which the object should be detected. 5VDC Operating voltage. I/O pins are 5V and 3.3V compliant.

Features of IR sensor module

- 5VDC Operating voltage
- I/O pins are 5V and 3.3V compliant
- Range: Up to 20cm
- Adjustable Sensing range
- Built-in Ambient Light Sensor
- 20mA supply current.
- Mounting hole

Applications of IR sensor module

- Automatic door, using the infrared microwave of the human body to open and close the door.
- Smoke alarm, using smoke resistance to measure the smoke concentration to achieve the purpose of the alarm.
- Mobile phones, digital camera cameras, using optical sensors to capture images.

- Electronic scale, making use of mechanical sensors (conductor strain gauge technology) to measure the pressure of the object corresponding to the variable to measuring weight.
- Water level alarm, temperature alarm, humidity alarm, optical alarm, etc.

5.2 Software Requirements

Software requirements deal with defining software resource requirements and prerequisites that need to be installed on a computer to provide optimal functioning of an application. These requirements or prerequisites are generally not included in the software installation package and need to be installed separately before the software is installed.

5.2.1 Operating system

An operating system is system software that manages computer hardware, software resources and provides common services for computer programs.

Raspbian is the recommended operating system for normal use on Raspberry pi. It is a free operating system based on Debian, optimized for the raspberry pi hardware. Raspbian comes with over 35000 packages pre-compiled software bundled in a nice format for easy installation on Raspberry pi. There are several versions of Raspbian including Raspbian Stretch and Raspbian Jessie. Since 2015 it has been officially provided by the Raspberry Pi Foundation as the primary operating system for the family of Raspberry Pi single-board computers. Raspbian was created by Mike Thompson and Peter Green as an independent project. The initial build was completed in June 2012. Raspbian is highly optimized for the Raspberry Pi line's low-performance ARM CPUs.

Raspbian uses PIXEL, Pi Improved X-Window Environment, Lightweight as its main desktop environment as of the latest update. It is composed of a modified LXDE desktop environment and the Open box stacking window manager with a new theme and few other changes. It is not affiliated with the Raspberry Pi Foundation. Raspbian was created by a small, dedicated team of developers that are fans of the Raspberry Pi hardware, the educational goals of the Raspberry Pi Foundation, and, of course, the Debian Project.

5.2.2 Languages Used

In computer science, a high-level programming language is a programming language with strong abstraction from the details of the computer. In contrast to low-level programming languages, it may use natural language elements, be easier to use, or may automate (or even hide entirely) significant areas of computing systems (e.g. memory management), making the process of developing a program simpler and more understandable than when using a lower-level language. The amount of abstraction provided defines how "high-level" a programming language

Java: Java is an object-oriented, cross-platform, multi-purpose programming language produced by Sun Microsystems. First released in 1995, it was developed to be a machine-independent web technology. It was based on C and C++ syntax to make it easy for programmers from those communities to learn. Since then, it has earned a prominent place in the world of computer programming. Java has many characteristics that have contributed to its popularity:

- Platform independence - Many languages are compatible with only one platform. Java was specifically designed so that it would run on any computer, regardless if it was running Windows, Linux, Mac, Unix, or any of the other operating systems.
- Simple and easy to use - Java's creators tried to design it so code could be written efficiently and easily.
- Multi-functional - Java can produce many applications from command-line programs to applets to Swing windows (basically, sophisticated graphical user interfaces).

Python: Python is a very useful programming language that has an easy to read syntax, and allows programmers to use fewer lines of code than would be possible in languages such as assembly, C, or Java. The Python programming language started as a scripting language for Linux. Python programs are similar to shell scripts in that the files contain a series of commands that the computer executes from top to bottom.

Python also has a large collection of libraries, which speeds up the development process. There are libraries for everything you can think of – game programming, rendering graphics, GUI interfaces, web frameworks, and scientific computing. Many of the things you can do in C can be done in Python. Python is generally slower at computations than C, but its ease of use makes Python an ideal language for prototyping programs and designing applications that aren't computationally intensive.

5.2.3 Application software

Application software (app for short) is a program or group of programs designed for end-users. Examples of an application include a word processor, a spreadsheet, an accounting application, a web browser, an email client, a media player, a file viewer, an aeronautical flight simulator, a console game or a photo editor. The collective noun application software refers to all applications collectively. This contrasts with system software, which is mainly involved with running the computer.

Applications may be bundled with the computer and its system software or published separately and may be coded as proprietary, open-source, or university projects. Apps built for mobile platforms are called mobile apps.

Android Studio: Android Studio is the official Integrated Development Environment (IDE) for Android app development, based on IntelliJ IDEA. On top of IntelliJ's powerful code editor and developer tools, Android Studio offers even more features that enhance your productivity when building Android apps, such as:

- A flexible Gradle-based build system
- A fast and feature-rich emulator
- A unified environment where you can develop for all Android devices
- Apply Changes to push code and resource changes to your running app without restarting your app
- Code templates and GitHub integration to help you build common app features and import sample code
- Extensive testing tools and frameworks
- Lint tools to catch performance, usability, version compatibility, and other problems
- C++ and NDK support
- Built-in support for Google Cloud Platform, making it easy to integrate Google Cloud Messaging and App Engine

This page provides an introduction to basic Android Studio features. For a summary of the latest changes, see Android Studio release notes.

Each project in Android Studio contains one or more modules with source code files and resource files. Types of modules include:

- Android app modules
- Library modules
- Google App Engine modules

By default, Android Studio displays your project files in the Android project view. This view is organized by modules to provide quick access to your project's key source files

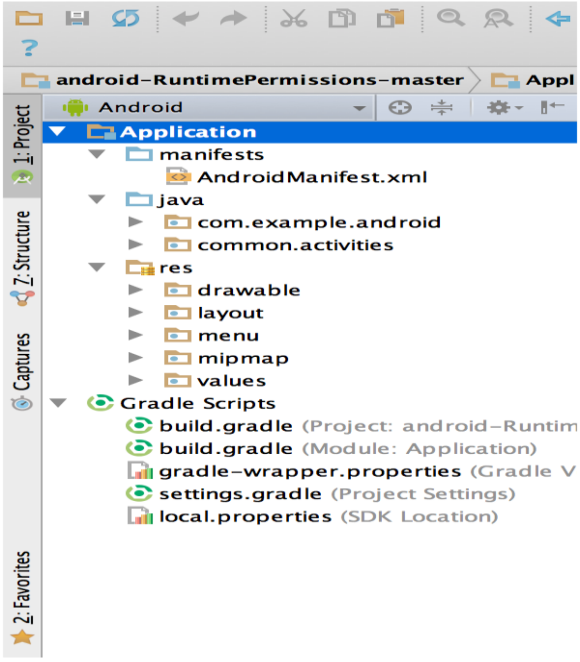

Fig 5.3: Tree structure of android studio documents.

All the build files are visible at the top level under Gradle Scripts and each app module contains the following folders:

- manifests: Contains the AndrioidManifest.xml file.
- java: Contains the Java source code files, including JUnit test code.
- res: Contains all non-code resources, such as XML layouts, UI strings, and bitmap images.

The Android project structure on disk differs from this flattened representation. To see the actual file structure of the project, select Project from the Project dropdown (in figure 1, it's showing as Android).

You can also customize the view of the project files to focus on specific aspects of your app development. For example, selecting the Problems view of your project displays links to the source files containing any recognized coding and syntax errors, such as a missing XML element closing tag in a layout file.

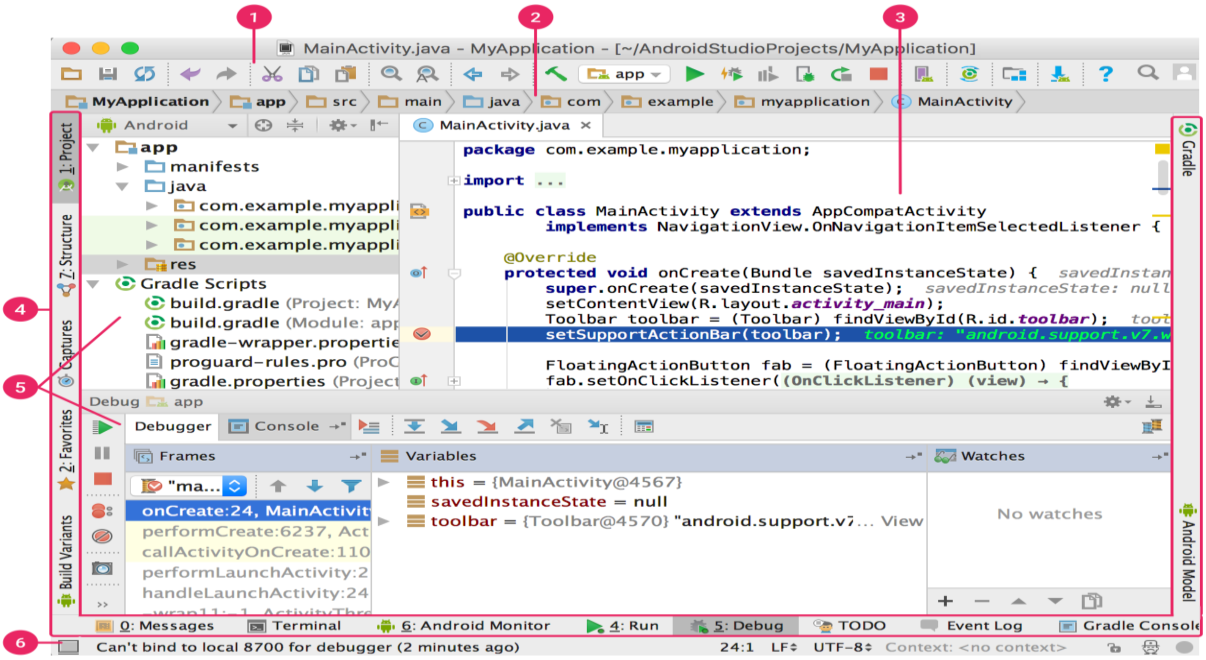

Fig 5.4: Android studio IDE structure.

The toolbar lets you carry out a wide range of actions, including running your app and launching Android tools.

1. The navigation bar helps you navigate through your project and open files for editing. It provides a more compact view of the structure visible in the Project window.
2. The editor window is where you create and modify code. Depending on the current file type, the editor can change. For example, when viewing a layout file, the editor displays the Layout Editor.
3. The tool window bar runs around the outside of the IDE window and contains the buttons that allow you to expand or collapse individual tool windows.
4. The tool windows give you access to specific tasks like project management, search, version control, and more. You can expand them and collapse them.
5. The status bar displays the status of your project and the IDE itself, as well as any warnings or messages.

Firebase: Firebase gives you the tools to develop high-quality apps, grow your user base, and earn more money. We cover the essentials so you can monetize your business and focus on your users.

Firebase is a mobile and web app development platform that provides developers with a plethora of tools and services to help them develop high-quality apps, grow their user base and earn more profit.

Fig 5.5: Firebase applications.

Firebase is a mobile and web app development platform that provides developers with a plethora of tools and services to help them develop high-quality apps, grow their user base and earn more profit. Back in 2011, before Firebase was Firebase, it was a start-up called Evolve. As Evolve, it provided developers with an API that enabled the integration of online chat functionality into their website. What's interesting is that people used to Evolve to pass application data that was more than just chat messages. Developers were using Evolve to sync application data such as a game state in real-time across their users. This led the founders of Evolve, James Templin and Andrew Lee, to separate the chat system and the real-time architecture.

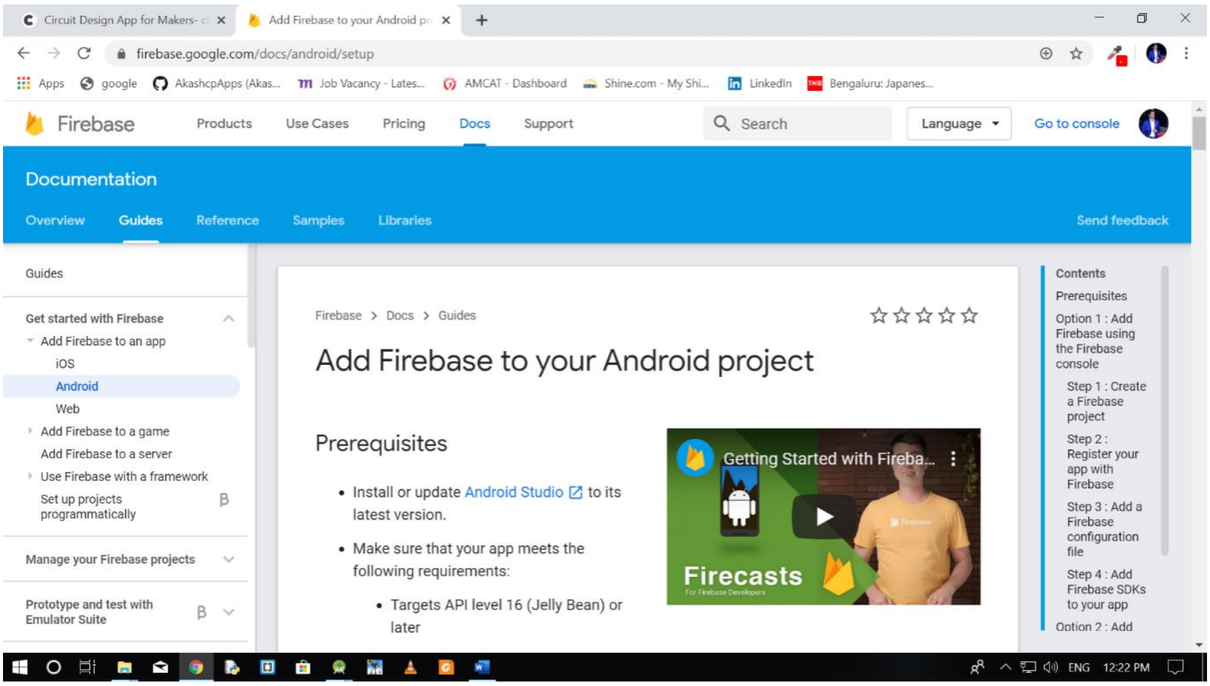

Fig 5.6: Firebase Documentation.

In April 2012, Firebase was created as a separate company that provided Backend-as-a-Service with real-time functionality.

After it was acquired by Google in 2014, Firebase rapidly evolved into the multifunctional behemoth of a mobile and web platform that it is today.

The Firebase Realtime Database is a cloud-hosted NoSQL database that lets you store and sync between your users in real-time. The Realtime Database is just one big JSON object that the developers can manage in real-time.

CHAPTER 6

SYSTEM DESIGN

The project consists of mainly six phases in the initial stage we gone gather the vehicle density information using the IR sensor for every 10 seconds then the sensors data transmit to the google cloud using real-time database bucket id in the google cloud traffic. Density is updated as signal1, signal2, signal3, and signal4. It has child density below that node the traffic density information is updated in this experiment we mainly consider three cases they are loose, moderate, and congested.

6.1 System Architecture

The module thus implements Density based traffic controlling system using IR technology. PIC is a very efficient architecture used for low-end security systems and IR technology is widely adapted technology for communication. Current work focuses on the effective usage of IR and PIC controller for digital security systems.

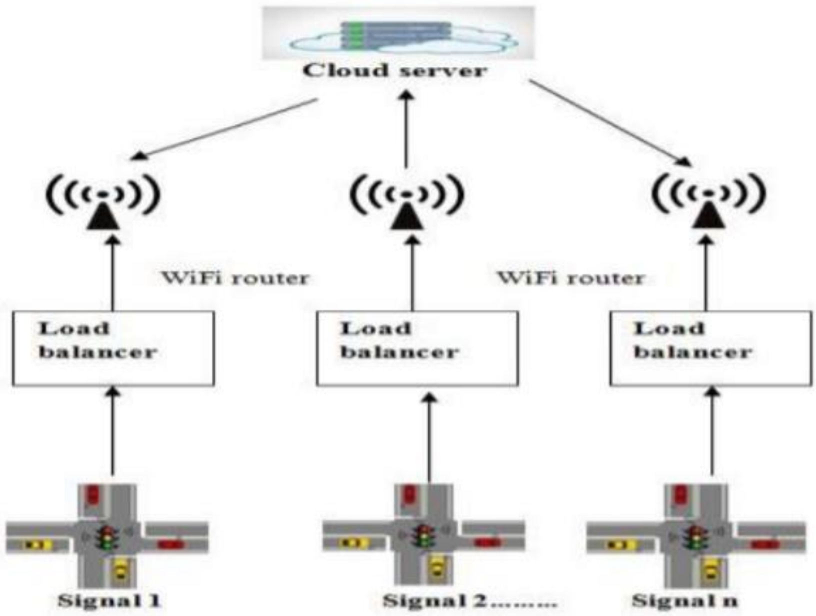

Fig 6.1: Block diagram of the advanced traffic management system.

The proposed system provides the count of the vehicles at either side of the junction while those vehicles are near the particular junction. Once the circuit is connected and coded, the circuit is sensed through the IR sensor for testing. This IR sensor is mainly used to describe the optoelectronic means for sensing the things, commonly known for photodetection. While testing keeps the transmitter & receiver aligned in a straight position facing each other. It deals with the proposed work and its Hardware & software requirements of the Traffic control system. And what would be the output for the given input? It also explains the process of managing the traffic in a city by making a connection to the cloud server.

6.2 Flow Diagram

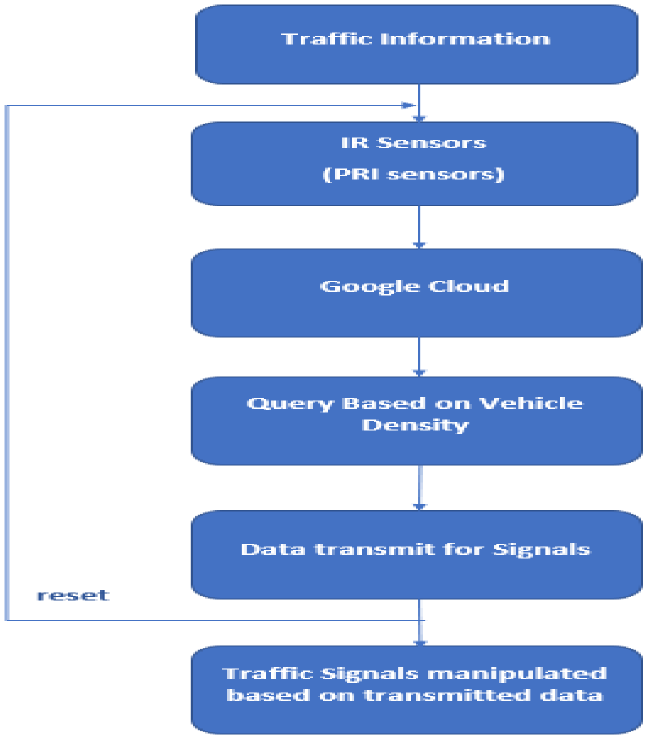

Fig 6.2: Flow chart of an advanced traffic management system.

Based upon these three conditions we go query in the google cloud itself. Then signals manipulate based on the information or signals transmitted by the google cloud. Finally, at the last stage, the cycle will repeat then control will be transmitted to the top of the flow chart.

- Loose: In this case, the vehicle density is too less.
- Moderate: In this case, the vehicle density in more compared to loosen case.
- Congested: In this case, the vehicle density is too high or above the limit.

The module thus implements Density based traffic controlling system using IR technology. PIC is a very efficient architecture used for low-end security systems and IR technology is widely adapted technology for communication. Current work focuses on the effective usage of IR and PIC controller for digital security systems. The proposed system provides the count of the vehicles at either side of the junction while those vehicles are near the particular junction. Once the circuit is connected and coded, the circuit is sensed through the IR sensor for testing. This IR sensor is mainly used to describe the optoelectronic means for sensing the things, commonly known for photodetection. While testing keeps the transmitter & receiver aligned in a straight position facing each other. It deals with the proposed work and its Hardware & software requirements of the Traffic control system. And what would be the output for the given input? It also explains the process of managing the traffic in a city by making a connection to the cloud server.

CHAPTER 7

IMPLEMENTATION

A proposed system for the detection of these vehicles is based on Radio-Frequency Identification (RFID). However, the use of this technology necessitates unnecessary extra hardware to be installed both at every junction and in every vehicle. There have also been studies to recognize these vehicles by analysis of the sound of their siren. However, this technology is also easily influenced by noise and requires additional hardware at every traffic signal. Personal Computer.

Traffic management automation systems in the market aim to computerized the traffic lights operates on the periodic to control the light (red/green) uses various technologies like GSM, NFC focuses on the basic operation of an electrical switch. Our project plan to provide an automated IR-sense based solution that makes traffic signals to shift the lights (red/green) dynamically. we plan on implementing the project for one junction "proof-of-concept" for this paper, which includes traffic lights, IR-sensors, Wi-Fi transmitters, and Raspberry Pi microcontrollers. The sensed data gathered from the IR sensor is transmitted by the Wi-Fi transmitter which is received by the raspberry-pi controller. Based on this compilation it dynamically shifts the time of the red signal and the user gets an intimation of the status of the signal on his way. The Raspberry Pi controller works as a central console, it determines which sideways of the road signal is to get open or close. The central console gathers all the data from sensors and stores it in the cloud which intimates traffic status to a mobile device.

7.1 Module description

The module thus implements Density based traffic controlling system using IR technology. PIC is a very efficient architecture used for low-end security systems and IR technology is widely adapted technology for communication. Current work focuses on the effective usage of IR and PIC controller for digital security systems. The proposed system provides the count of the vehicles at either side of the junction while those vehicles are near the particular junction. Once the circuit is connected and coded, the circuit is sensed through the IR sensor for testing. This IR sensor is mainly used to describe the optoelectronic means for sensing the things, commonly known for photodetection. While testing keeps the transmitter & receiver aligned in a straight position facing each other. It deals with the proposed work and its Hardware & software requirements of the Traffic control system. And what would be the output for the given input? It also explains the process of managing the traffic in a city by making a connection to the cloud server.

7.1.1 Vehicle Sensing

Vehicles can be identified using suitable sensors for the real-world scenario, but for demonstration purposes, we have used IR sensors for the identification of vehicles. There are two types of infrared (IR) detectors, active and passive. Active infrared sensors operate by transmitting energy from either a light-emitting diode (LED) or a laser diode. An LED is used for a non-imaging active IR detector, and a laser diode is used for an imaging active IR detector. If you are talking about a general IR sensor used in college projects then its range is maxing 4–5 meters.

7.1.2 Signal Updating

Our project mainly works in the round robin in which control flow through signals s in Circle format. At initial stage the control is passed to signal one based upon density signal is manipulated after execution the control flow through the next signal ,the value updated to the firebase based on density in which number of vehicles detected in an interval of time then based upon the count The density value updated to firebase node the value stored in the format of JSON (java script object notation) in key value pairs .

7.2 Pseudocode

```
from signal import pause
import RPi.GPIO as GPIO
import time
from threading import Timer
import datetime
from firebase import firebase
GPIO.setmode(GPIO.BCM)

IR_PIN=2
IR_PIN2=3
IR_PIN3=14
IR_PIN4=4
```

```
GPIO.setup(IR_PIN,GPIO.IN)
GPIO.setup(IR_PIN2,GPIO.IN)
GPIO.setup(IR_PIN3,GPIO.IN)
GPIO.setup(IR_PIN4,GPIO.IN)

count1=0
count2=0
count3=0
count4=0
sub1=0
sub2=0
sub3=0
sub4=0

while 1:
    something=GPIO.input(IR_PIN)
    something1=GPIO.input(IR_PIN2)
    something2=GPIO.input(IR_PIN3)
    something3=GPIO.input(IR_PIN4)

    if not something:
        sub1+=1
        print("signal 1 detected".format(sub1))
        #print("signal format1"+str(sub1))
        start_time=datetime.datetime.now().time().strftime('%H:%M:%S')
        time.sleep(3)
        end_time=datetime.datetime.now().time().strftime('%H:%M:%S')
        tital_time=(datetime.datetime.strptime(end_time,'%H:%M:%S')-
datetime.datetime.strptime(start_time,'%H:%M:%S'))
       # print(sub1)

        if(tital_time):
            if sub1<2:
                print('loose')
```

```
            traffic='loose'
            fb=firebase.FirebaseApplication('https://m1signal.firebaseio.com/',None)
            result=fb.put('signal1','density',traffic)
            # sub1=0

        if sub1>2 and sub1<5:
            print('moderate')
            traffic='moderate'
            fb=firebase.FirebaseApplication('https://m1signal.firebaseio.com/',None)
            result=fb.put('signal1','density',traffic)
            # sub1=0

        if sub1>5:
            print('Conjested')
            traffic='conjested'
            fb=firebase.FirebaseApplication('https://m1signal.firebaseio.com/',None)
            result=fb.put('signal1','density',traffic)
            sub1=0

  if not something1:
      sub2+=1
      start_time=datetime.datetime.now().time().strftime('%H:%M:%S')
      time.sleep(3)
      end_time=datetime.datetime.now().time().strftime('%H:%M:%S')
      tital_time=(datetime.datetime.strptime(end_time,'%H:%M:%S')-
datetime.datetime.strptime(start_time,'%H:%M:%S'))
      print("Signal 2 Detected")
     # print("signal format1"+str(sub2))

      if(tital_time):
          if sub2<2:
              print('loose')
              traffic='loose'
              fb=firebase.FirebaseApplication('https://m1signal.firebaseio.com/',None)
```

```
            result=fb.put('signal2','density',traffic)
            # sub2=0

        if sub2>2 and sub2<5:
            traffic='moderate'
            fb=firebase.FirebaseApplication('https://m1signal.firebaseio.com/',None)
            result=fb.put('signal2','density',traffic)
            print('moderate')
            # sub2=0

        if sub2>5:
            traffic='conjested'
            fb=firebase.FirebaseApplication('https://m1signal.firebaseio.com/',None)
            result=fb.put('signal2','density',traffic)
            print('conjested')
            sub2=0

if not something2:
    sub3+=1
    print("Signal 4 Detected")
    # print("signal format3 "+str(sub3))
    start_time=datetime.datetime.now().time().strftime('%H:%M:%S')
    time.sleep(3)
    end_time=datetime.datetime.now().time().strftime('%H:%M:%S')
    tital_time=(datetime.datetime.strptime(end_time,'%H:%M:%S')-
datetime.datetime.strptime(start_time,'%H:%M:%S'))

    if(tital_time):
        if sub3<2:
            print('loose')
            traffic='loose'
            fb=firebase.FirebaseApplication('https://m1signal.firebaseio.com/',None)
            result=fb.put('signal4','density',traffic)
            # sub3=0
```

```
        if sub3>2 and sub2<5:
            print('moderate')
            traffic='moderate'
            fb=firebase.FirebaseApplication('https://m1signal.firebaseio.com/',None)
            result=fb.put('signal4','density',traffic)
            # sub3=0
        if sub3>5:
            print('Conjested')
            traffic='conjested'
            fb=firebase.FirebaseApplication('https://m1signal.firebaseio.com/',None)
            result=fb.put('signal4','density',traffic)
            sub3=0

    if not something3:
        sub4+=1
        print("Signal 3 Detected")
        start_time=datetime.datetime.now().time().strftime('%H:%M:%S')
        time.sleep(3)
        end_time=datetime.datetime.now().time().strftime('%H:%M:%S')
        tital_time=(datetime.datetime.strptime(end_time,'%H:%M:%S')-
datetime.datetime.strptime(start_time,'%H:%M:%S'))
        # print("Signal 2 Detected")

        if(tital_time):
            if sub4<2:
                print('loose')
                traffic='loose'
                fb=firebase.FirebaseApplication('https://m1signal.firebaseio.com/',None)
                result=fb.put('signal3','density',traffic)
                # sub4=0

            if sub4>2 and sub2<5:
                print('moderate')
                traffic='moderate'
```

```
    fb=firebase.FirebaseApplication('https://m1signal.firebaseio.com/',None)
    result=fb.put('signal3','density',traffic)
    # sub4=0

if sub4>5:
    print('conjested')
    traffic='conjested'
    fb=firebase.FirebaseApplication('https://m1signal.firebaseio.com/',None)
    result=fb.put('signal3','density',traffic)
    sub4=0
```

CHAPTER 8

RESULTS AND SNAPSHOTS

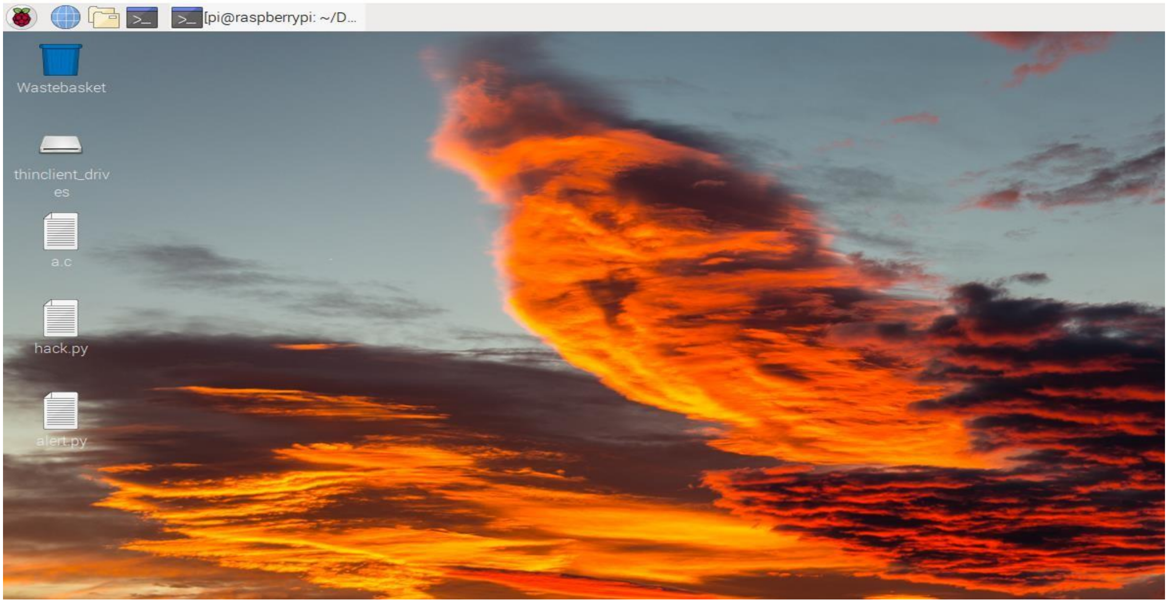

Fig 8.1: Desktop view of Raspberry pi Operating System.

Fig 8.1 represents the desktop directory of the Raspberry pi Operating System.

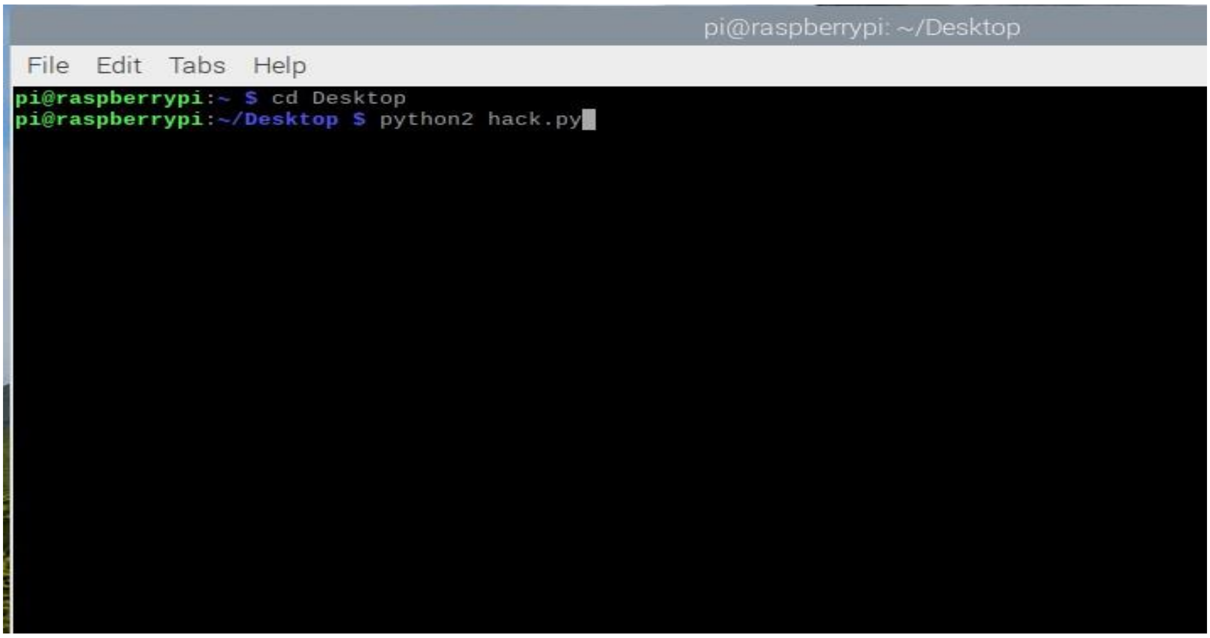

Fig 8.2: Run Python code using Terminal.

Fig 8.2 shows Terminal by clicking ctr + alt + T for opening the command prompt then change the directory to the desktop in this directory having a python file called hack.py run this file by using command python2 hack.py.

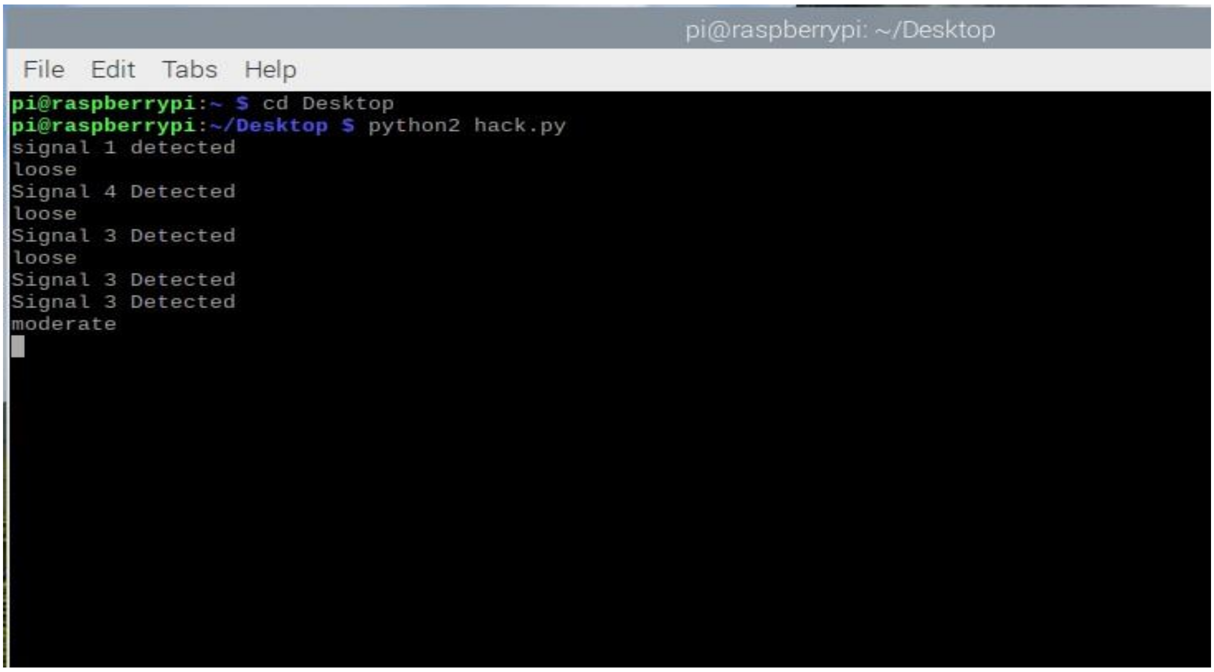

Fig 8.3: Signal Detected by IR Sensor displaying in Terminal.

Fig 8.3 shows the terminal running a hack.py file we can find the above results which mean objects came into contact with the IR sensors we can find the resultant of each IR sensor.

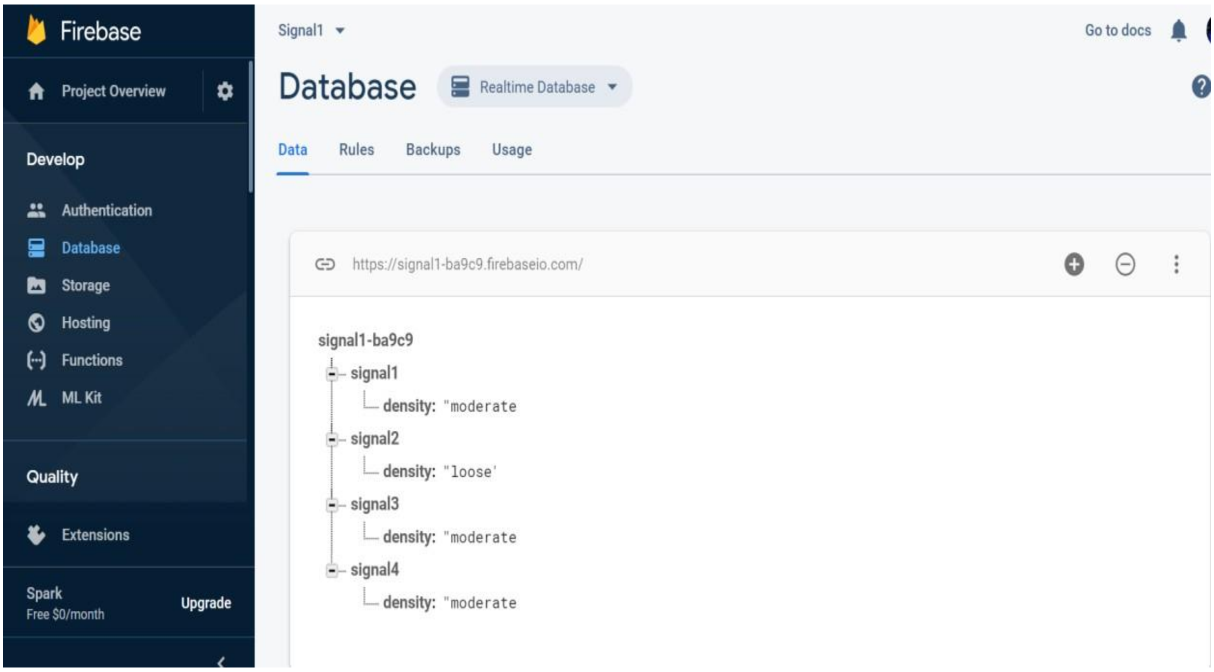

Fig 8.4: Dashboard of the Firebase Realtime Database.

Fig 8.4 represents the tree structure of each IR sensors data stored in terms of JSON format signal name is the main root node and density is sub-node which contains IR sensors density value

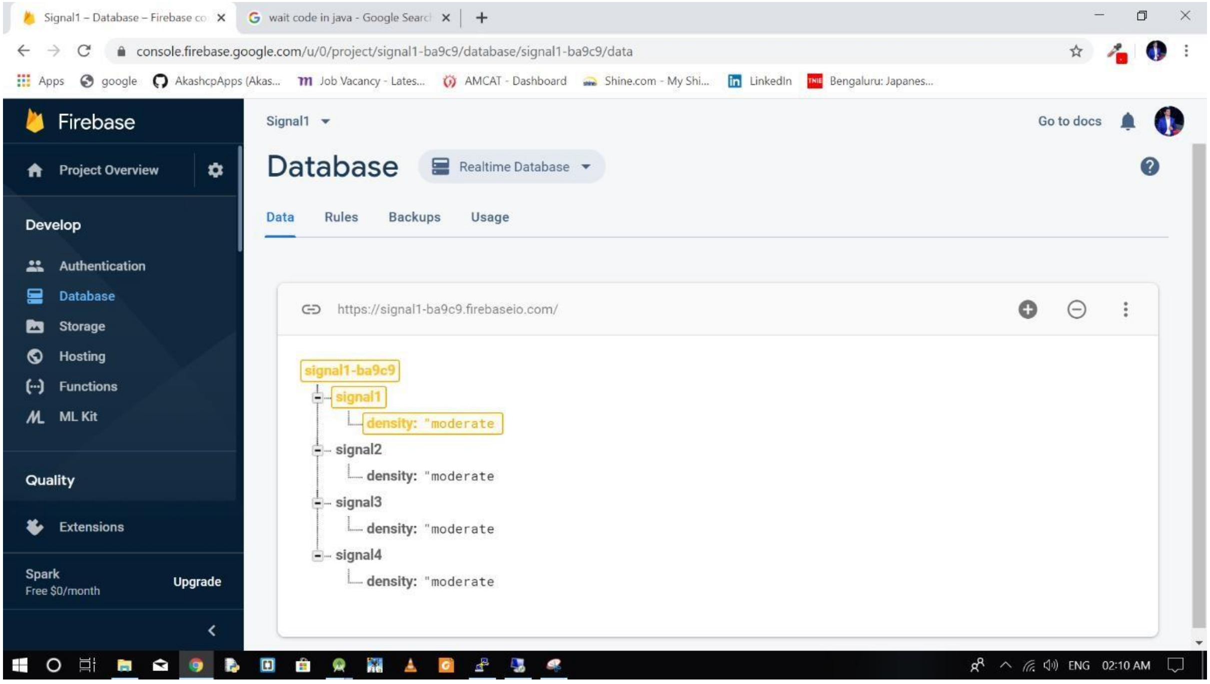

Fig 8.5: Value Updated by the Raspberry pi to Firebase Realtime Database.

Fig 8.5 represents the tree structure of each IR sensor data stored in terms of JSON format signal name is the main root node and density is sub-node which contains IR sensors density value Yellow color repents the value updated in real-time.

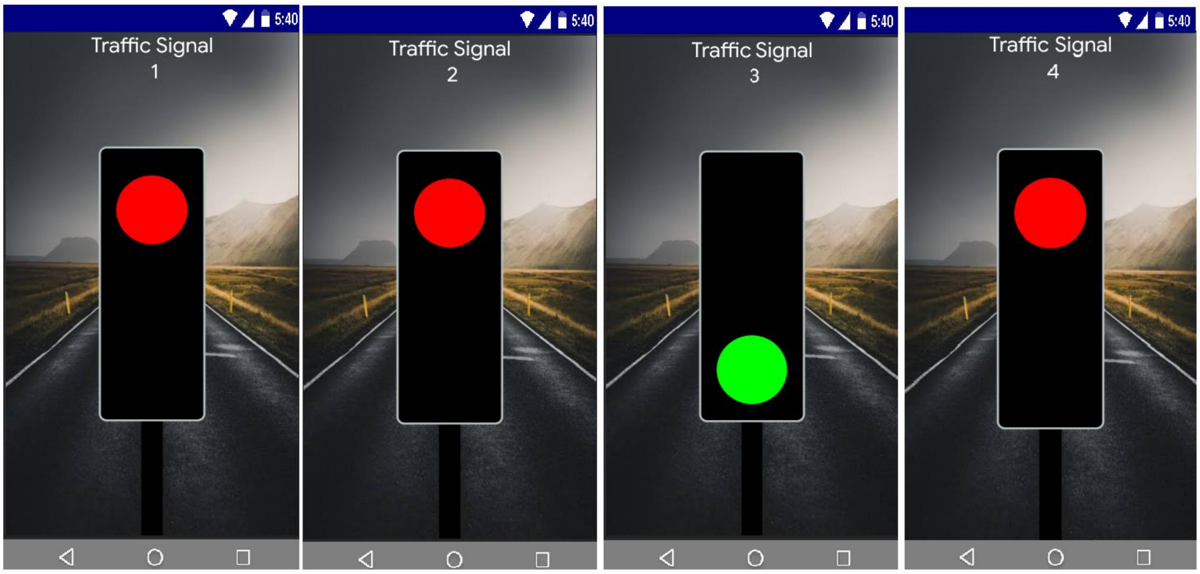

Fig 8.6: Android UI Provided for Signal Demonstration

Fig 8.6 represents the traffic signals as the screenshots control is present at traffic signal 3 based on the density value traffic signal will be manipulated and control flow to the next signal.

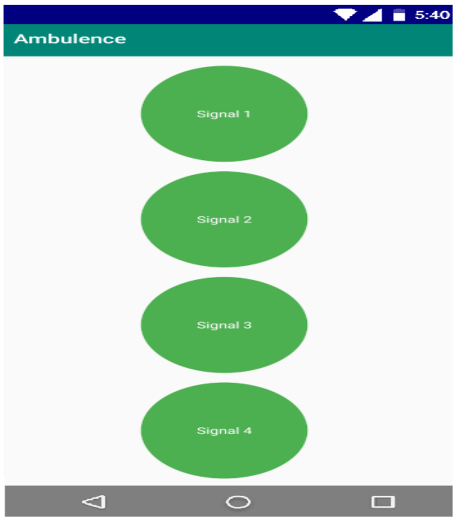

Fig 8.7: Android UI provided for ambulance admin.

Fig 8.7 represents the UI of the ambulance app which can be controlled by the traffic police or ambulance driver for clearing the traffic in the path of the ambulance car or emergency by clicking the signal 1 button which means signal 1 became green and other signals will become the red color for a particular period.

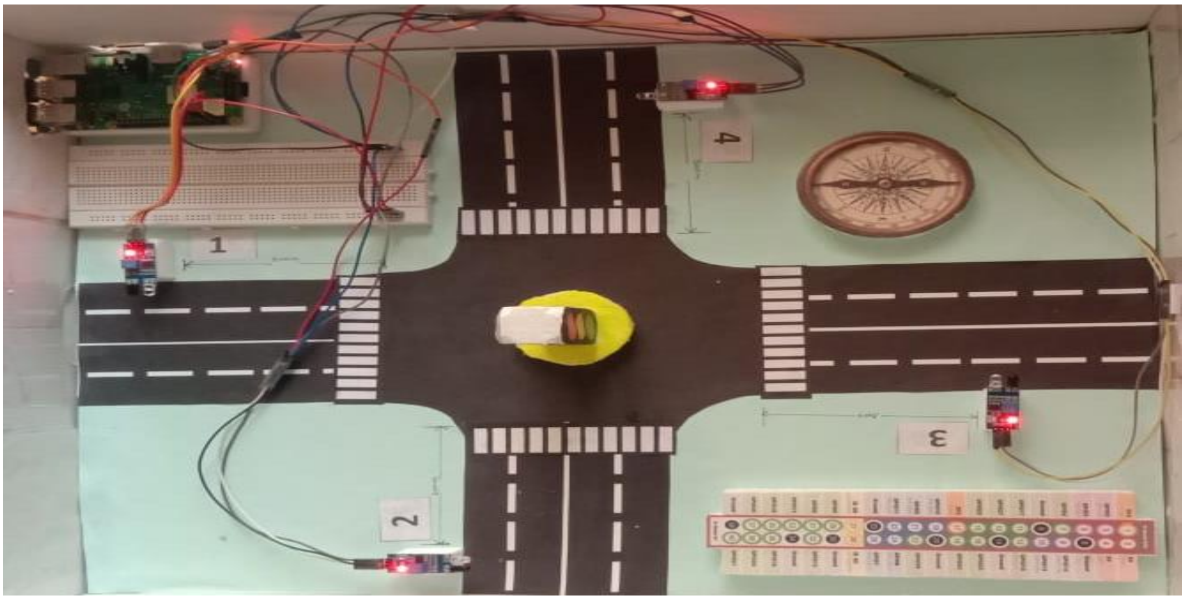

Fig 8.8: Advanced traffic management system using a google cloud model.

Fig 8.8 is the working model of our project Advanced traffic management system using google cloud.

REFERENCES

[1] Rajasheswari sunder, SanthoshsHebber, and varaprasadGolla, Implementing intelligence Traffic control system for congestion control, Ambulance Clearness, and Stolen Vehicle Detection " IEEE Sensors Journal, Vol. 15, No.2 February 2015.

[2] M.Vidhyia, K.Paramasivam, S.Elayaraja, S.Bharathiraja Reordering of Test vectors using weighting factor based on Average power for test power Minimization, Vol.4, No.2, 2015, pp.10-15.

[3] Raspberry pi Web page: https://en.wikipedia.org/wiki/Rasspberry-pi [NOV 2015].

[4] K.Vidhya, A.Bazila Banu, "Density Based Traffic signal System", Volume 3, Special Issue 3, March 2014.

[5] Priyanka Khanke, prof. P.S.Kulkarni, "A Technique on Road Traffic Analysis using Image Processing", Vol. 3 Issue 2, February 2014.

[5] Ms. Pallavi Choudekar, Ms. Sayanti Banerjee Prof. M.K. Muju, Real-Time Traffic Light Control Using Image processing " Vol.2, No. March 2014.

[6] R. Nithin Goutham1, J. Sharon Roza2, M. Santhosh3, Intelligent signal control system, Vol.3, Special Issue 4, May 2014.

[7] S. Lokesh, T. Prahald Reddy, An Adaptive Traffic Control SystemUsing Raspberry pi, Vol 3(6): June 2014