"ULTRA-FFIC BOT"

A PROJECT REPORT

Submitted by

Bardolia Priyank (17042010501)

Mahatma Saloni (17042010530)

Mistry Vishal (17042010536)

Parmar Siddhant (17042010552)

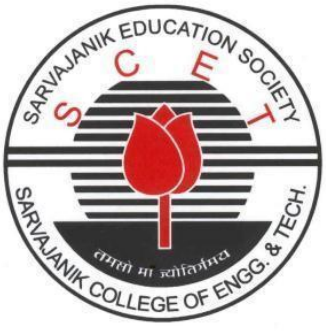
Srivastav Umang (17042010554)

In fulfillment for the award of the degree
Of

BACHELOR OF ENGINEERING

In

COMPUTER ENGINEERING



Sarvajanik College of Engineering and Technology,
Surat.

Gujarat Technological University, Ahmedabad.

Nov - Dec, 2020

SARVAJANIK COLLEGE OF ENGINEERING AND TECHNOLOGY

Dr. R.K.DESAI MARG, ATHWALINES,

SURAT-395001

DEPARTMENT OF COMPUTER ENGINEERING

CERTIFICATE

This is to certify that the project entitled *Ultra-ffic Bot* has been carried out by Bardolia Priyank (17042010501), Mahatma Saloni (17042010530), Mistry Vishal (17042010536), Parmar Siddhant (17042010552), Srivastav Umang (17042010554) students of B.E.IV (CO), Semester-VII, under my guidance in fulfilment of the degree of Bachelor of Engineering in Computer Engineering of Gujarat Technological University, Ahmedabad for the academic year Nov – Dec, 2020.

Signature of Guide	Signature of Head of the Department
---	---
Prof. Bintu Kadhiwala	Dr. Pariza Kamboj

Signature of Jury Members

--- , --- , ---

GUJARAT TECHNOLOGICAL UNIVERSITY

CERTIFICATE FOR COMPLETION OF ALL ACTIVITIES AT ONLINE PROJECT PORTAL

B.E. SEMESTER VIII, ACADEMIC YEAR 2020-2021

Date of certificate generation : 16 April 2021 (22:28:38)

This is to certify that, ***Bardolia Priyank Harshad*** (Enrolment Number - 170420107501) working on project entitled with ***ULTRA-FFIC BOT*** from ***Computer Engineering*** department of ***SARVAJANIK COLLEGE OF ENGINEERING & TECHNOLOGY, SURAT*** had submitted following details at online project portal.

Periodic Progress Reports (PPR)	**Completed**
Business Model Canvas (Image)	**Completed**
Business Model Canvas (Report)	**Completed**
Patent Drafting Exercise (PDE)	**Completed**
Final Plagiarism Report	**Completed**
Final Project Report	**Completed**

Name of Student : Bardolia Priyank Harshad

Name of Guide : Mr.Kadhiwala Bintu Bharatkumar

Signature of Student : _____

*Signature of Guide : _____

Disclaimer :
This is a computer generated copy and does not indicate that your data has been evaluated. This is the receipt that GTU has received a copy of the data that you have uploaded and submitted as your project work.

*Guide has to sign the certificate, Only if all above activities has been Completed.

GUJARAT TECHNOLOGICAL UNIVERSITY

CERTIFICATE FOR COMPLETION OF ALL ACTIVITIES AT ONLINE PROJECT PORTAL

B.E. SEMESTER VIII, ACADEMIC YEAR 2020-2021

Date of certificate generation : 16 April 2021 (22:30:45)

This is to certify that, ***Mahatma Saloni Nileshkumar*** (Enrolment Number - 170420107530) working on project entitled with ***ULTRA-FFIC BOT*** from ***Computer Engineering*** department of ***SARVAJANIK COLLEGE OF ENGINEERING & TECHNOLOGY, SURAT*** had submitted following details at online project portal.

Periodic Progress Reports (PPR)	**Completed**
Business Model Canvas (Image)	**Completed**
Business Model Canvas (Report)	**Completed**
Patent Drafting Exercise (PDE)	**Completed**
Final Plagiarism Report	**Completed**
Final Project Report	**Completed**

Name of Student : Mahatma Saloni Nileshkumar

Name of Guide : Mr.Kadhiwala Bintu Bharatkumar

Signature of Student : _____

*Signature of Guide : _____

Disclaimer :
This is a computer generated copy and does not indicate that your data has been evaluated. This is the receipt that GTU has received a copy of the data that you have uploaded and submitted as your project work.

*Guide has to sign the certificate, Only if all above activities has been Completed.

GUJARAT TECHNOLOGICAL UNIVERSITY

CERTIFICATE FOR COMPLETION OF ALL ACTIVITIES AT ONLINE PROJECT PORTAL

B.E. SEMESTER VIII, ACADEMIC YEAR 2020-2021

Date of certificate generation : 16 April 2021 (22:26:51)

This is to certify that, ***Mistry Vishal Dharmeshbhai*** (Enrolment Number - 170420107536) working on project entitled with ***ULTRA-FFIC BOT*** from ***Computer Engineering*** department of ***SARVAJANIK COLLEGE OF ENGINEERING & TECHNOLOGY, SURAT*** had submitted following details at online project portal.

Periodic Progress Reports (PPR)	**Completed**
Business Model Canvas (Image)	**Completed**
Business Model Canvas (Report)	**Completed**
Patent Drafting Exercise (PDE)	**Completed**
Final Plagiarism Report	**Completed**
Final Project Report	**Completed**

Name of Student : Mistry Vishal Dharmeshbhai

Name of Guide : Mr.Kadhiwala Bintu Bharatkumar

Signature of Student : _____

*Signature of Guide : _____

Disclaimer :
This is a computer generated copy and does not indicate that your data has been evaluated. This is the receipt that GTU has received a copy of the data that you have uploaded and submitted as your project work.

*Guide has to sign the certificate, Only if all above activities has been Completed.

GUJARAT TECHNOLOGICAL UNIVERSITY

CERTIFICATE FOR COMPLETION OF ALL ACTIVITIES AT ONLINE PROJECT PORTAL

B.E. SEMESTER VIII, ACADEMIC YEAR 2020-2021

Date of certificate generation : 16 April 2021 (22:28:26)

This is to certify that, ***Siddhant Parmar*** (Enrolment Number - 170420107552) working on project entitled with ***ULTRA-FFIC BOT*** from ***Computer Engineering*** department of ***SARVAJANIK COLLEGE OF ENGINEERING & TECHNOLOGY, SURAT*** had submitted following details at online project portal.

Periodic Progress Reports (PPR)	**Completed**
Business Model Canvas (Image)	**Completed**
Business Model Canvas (Report)	**Completed**
Patent Drafting Exercise (PDE)	**Completed**
Final Plagiarism Report	**Completed**
Final Project Report	**Completed**

Name of Student : Siddhant Parmar

Name of Guide : Mr.Kadhiwala Bharatkumar Bintu

Signature of Student : _____

*Signature of Guide : _____

Disclaimer :
This is a computer generated copy and does not indicate that your data has been evaluated. This is the receipt that GTU has received a copy of the data that you have uploaded and submitted as your project work.

*Guide has to sign the certificate, Only if all above activities has been Completed.

GUJARAT TECHNOLOGICAL UNIVERSITY

CERTIFICATE FOR COMPLETION OF ALL ACTIVITIES AT ONLINE PROJECT PORTAL

B.E. SEMESTER VIII, ACADEMIC YEAR 2020-2021

Date of certificate generation : 16 April 2021 (22:31:53)

This is to certify that, ***Srivastava Umang Vipul*** (Enrolment Number - 170420107554) working on project entitled with ***ULTRA-FFIC BOT*** from ***Computer Engineering*** department of ***SARVAJANIK COLLEGE OF ENGINEERING & TECHNOLOGY, SURAT*** had submitted following details at online project portal.

Periodic Progress Reports (PPR)	**Completed**
Business Model Canvas (Image)	**Completed**
Business Model Canvas (Report)	**Completed**
Patent Drafting Exercise (PDE)	**Completed**
Final Plagiarism Report	**Completed**
Final Project Report	**Completed**

Name of Student : Srivastava Umang Vipul

Name of Guide : Mr.Kadhiwala Bharatkumar Bintu

Signature of Student : _____

*Signature of Guide : _____

Disclaimer :
This is a computer generated copy and does not indicate that your data has been evaluated. This is the receipt that GTU has received a copy of the data that you have uploaded and submitted as your project work.

*Guide has to sign the certificate, Only if all above activities has been Completed.

Plagiarism Detector.net

Date: April, 16 2021

PLAGIARISM SCAN REPORT

Excluded Url : None

Content Checked For Plagiarism

Traffic congestion has always been a major problem of any populated city. Number of vehicles are increasing day by day with the increase in population. With a large number of vehicles, it gets more and more difficult for the city to manage the traffic. Although there are traffic signals to manage the traffic and reduce the chaos, heavy congestion is observed at the junctions. As per our observation, one of the possible reasons for the same is that the traffic lights are configured with static time intervals. This is where the concept of Dynamic/Adaptive traffic signals come into picture. The traffic signals need to be modified in such a way that it adapts itself according to the changing density of vehicles present at each lane of junction (traffic signal). CHAPTER – 5 IMPLEMENTATION 5.1 ACTUAL IMPLEMENTATION Module - 1 : Vehicle Density Estimation This module takes traffic images as an input that undergoes image pre-processing for feature extraction. We get vehicle count as an output. In addition, vehicles and other surrounding elements (such as a person(s), animal(s), etc..) are also classified into various specific classes. We have used a deep convolutional neural network called Darknet-53 architecture. It is a 53 layer neural network trained on ImageNet for classi?cation combined with detection layers making the total network 106 layers deep. Module - 2 : Emergency Vehicle Detection Step 1: Data Collection Images were downloaded from the internet Videos of the ambulance were recorded using from all angle using a drone Step 2: Data Preprocessing Video Slicing: Obtained frames (interval: 3s) by slicing the recorded video and redundant images were removed. Resizing: Because yolo requires the size of input images to be in multiples of 32, all images were resized to 412x412 Orientation: Due to resizing, annotation of all images were oriented according to resized images. Step 3: Annotation LabelImg software was used to annotate (bounding box coordinates) all the images in yolo format Step 4: Custom object training of yolov4 model We trained yolov4-tiny for 2 classes namely: Ambulance and Firetruck. These two classes are reflected in /content/darknet/data/obj.names file Step 5: Testing Module - 3 : Embedded Controller/ Simulator In this module, we started our implementation on Simulator. We started by making a virtual circuit of the proposed system on fritzing software. Later we tried various microcontrollers including Arduino Uno and raspberry pi. We created the proposed system in a virtual environment of the Proteus 8 professional software. We then modified the simulator project to integrate with module-4 (android application) of our project. The machine learning models were then deployed on Raspberry pi. To satisfy more processing and GPU needs of the machine learning model, we created a server on Google Cloud Platform (GCP). The server acts as a central processing unit for the whole project, processing is carried out in the server, also it is integrated with the Raspberry pi, Firestore database, Android application and the website. Module - 4 : Android Application The mobile android-based application is a controller which can be switched from manual mode to automatic mode and vice versa based on necessity. In static mode, time for red, yellow and green light is set. In dynamic mode the ve. The mode and time provided are sent to the simulator for further action. We created the android application login and registration page by integrating it with the firebase database. Then we coded the java application for connectivity with the microcontroller. It will be used as a control centre for the traffic signal. Module - 5 : Website The website module is an administrative module used to manage the traffic police and traffic junctions. With the help of this website, we can assign or de-assign a junction to a user, we can approve or disapprove a user, we can add or remove a user as an admin, and we can add a junction or view the allocated junctions. The website is integrated with a Firestore database that handles the login credentials and makes the admin features available only to an admin user. We've implemented all Modules of the project. With the help of computer vision and image processing, we detect the density of the traffic and make the traffic signals dynamic as per the analysed data. With this system, we not only make traffic signals intelligent but also help the emergency service vehicles with easy passage and thus helping people by saving their precious time. Hence we would like to conclude that "the new age of traffic management will begin, and being stuck at a traffic jam will no longer be an excuse to be late". Linking of multiple signals together to determine the flow of traffic and

suggesting efficient traffic routes in order to reduce the time of travelling. In a scenario when there is power cut the signal won't be able to work and communication will be interrupted. With more images in our dataset we can improve our accuracy. In case of harsh weather conditions, the model won't be able to work properly as the images fetched will not be clearer. Efficient passage of vehicles is performed at the junction and hence time is saved. Prevention of Emergency Vehicle from getting stuck in the traffic by giving them Priority access. Efficient management of traffic with an aim to reduce the pollution, wastage of fuel, cost of transportation and stress to drivers. OpenCV (Image Processing) OpenCV (Open Source Computer Vision Library) is an open-source computer vision and machine learning software library. OpenCV was built to provide a common infrastructure for computer vision applications and to accelerate the use of machine perception in commercial products. YOLOv3 (Object detection) YOLO V3 is an improvement over previous YOLO detection networks. Compared to prior versions, it features multi-scale detection, a stronger feature extractor network, and some changes in the loss function. As a result, this network can now detect many more targets from big to small.

3% Plagiarized

· OpenCV (Open Source Computer Vision Library) is an open source computer vision and machine learning software library. OpenCV was built to provide a …

https://medium.com/analytics-vidhya/hand-detection-and-finger-counting-using-opencv-python-5b594704eb08

3% Plagiarized

OpenCV (Open Source Computer Vision Library) is an open source computer vision and machine learning software library. OpenCV was built to provide a common infrastructure for computer vision applications and to accelerate the use of machine perception in the commercial products.

https://opencv.org/about/

6% Plagiarized

YOLO V3 is an improvement over previous YOLO detection networks. As a result, this network can now detect many more targets from big to small. ... And, of course, just like other single-shot detectors, YOLO V3 also runs quite fast and makes real-time inference possible on GPU devices.

https://towardsdatascience.com/dive-really-deep-into-yolo-v3-a-beginners-guide-9e3d2666280e

---

3% Plagiarized

Dive Really Deep into YOLO v3: A Beginner's Guide | by ...

https://towardsdatascience.com/dive-really-deep-into-yolo-v3-a-beginners-guide-9e3d2666280e#:~:text=YOLO%20V3%20is%20an%20improvement,targets%20from%20big%20to%20small.

Acknowledgement

We thank our guide Prof. Bintu Kadhiwala for guiding us and giving us his valuable time and advice. He always enriched us with his knowledge and gave us the necessary input to carry out this work. We are grateful to him for his extra efforts and for being patient with us.

We take this opportunity to thank all our lecturers who have directly or indirectly helped in our project.

Table of Contents

List of Figures

CHAPTER – 1 INTRODUCTION

1.1 PROBLEM SUMMARY

Traffic congestion has always been a major problem in populated cities. With the increase in population, vehicles are increasing day by day. With a large number of vehicles, it gets more and more difficult for the city to manage the traffic.

Although there are traffic signals to manage the traffic and reduce the chaos, heavy congestion is observed at the junctions. As per our observation, one of the possible reasons for the same is that the traffic lights are configured with static time intervals.

This is where the concept of Dynamic/Adaptive traffic signals come into the picture. The traffic signals need to be modified in such a way that it adapts itself according to the changing density of vehicles present at each lane of a junction (traffic signal).

1.2 AIMS AND OBJECTIVES

We aim to develop **ULTRA-FFIC Bot-an intelligent traffic signalling system** that takes into account the number of vehicles on each lane and gives priority to the lane that has more vehicles. Also, the presence of an Emergency vehicle (Firetruck, ambulance) is detected and priority is given to that lane.

Furthermore, an android app is developed, which can be used to set a timer for each lane so that effective management is possible in case of political rallies or wedding processions.

Additionally, a website is created for an administrative purpose that can be used to manage the users and junctions.

CHAPTER – 2 BRIEF LITERATURE REVIEW

2.1 Patent Study

1. Nellore et al. [1]

Vehicular traffic is endlessly increasing everywhere in the world and can cause terrible traffic congestion at intersections. Most of the traffic lights today feature a fixed green light sequence, therefore the green light sequence is determined without taking the presence of the emergency vehicles into account. Therefore, emergency vehicles such as ambulances, police cars, fire engines, etc. stuck in a traffic jam and delayed in reaching their destination can lead to loss of property and valuable lives. This paper presents an approach to schedule emergency vehicles in traffic. The approach combines the measurement of the distance between the emergency vehicle and an intersection using visual sensing methods, vehicle counting and time sensitive alert transmission within the sensor network. The distance between the emergency vehicle and the intersection is calculated for comparison using Euclidean distance, Manhattan distance and Canberra distance techniques. The experimental results have shown that the Euclidean distance outperforms other distance measurement techniques. Along with visual sensing techniques to collect emergency vehicle information, it is very important to have a Medium Access Control (MAC) protocol to deliver the emergency vehicle information to the Traffic Management Center (TMC) with less delay. Then only the emergency vehicle is quickly served and can reach the destination in time. In this paper, we have also investigated the MAC layer in WSNs to prioritize the emergency vehicle data and to reduce the transmission delay for emergency messages. We have modified the medium access procedure used in standard IEEE 802.11p with PE-MAC protocol, which is a new back off selection and contention window adjustment scheme to achieve low broadcast delay for emergency messages. A VANET model for the UTMS is developed and simulated in NS-2. The performance of the standard IEEE 802.11p and the proposed PE-MAC is analysed in detail. The NS-2 simulation results have shown that the PE-MAC outperforms the IEEE 802.11p in terms of average end-to-end delay, throughput and energy consumption. The performance evaluation results have proven that the proposed PE-MAC prioritizes the emergency vehicle data and delivers the emergency messages to the TMC with less delay compared to the IEEE 802.11p. The transmission delay of the proposed PE-MAC is also compared with the standard IEEE 802.15.4, and Enhanced Back-off Selection

scheme for IEEE 802.15.4 protocol [EBSS, an existing protocol to ensure fast transmission of the detected events on the road towards the TMC] and the comparative results have proven the effectiveness of the PE-MAC over them. Furthermore, this research work will provide an insight into the design of an intelligent urban traffic management system for the effective management of emergency vehicles and will help to save lives and property.

2. Bakhare et al. [2]

One or more RFID information readers, and a microcontroller including memory storage, said one or more speed and direction sensors and one or more range-sensing devices being connected to said microcontroller; route planning means for generating strategic route plans for said two or more vehicles located in said enroute passage; each vehicle including tracking and monitoring means to track and monitor its global position and orientation in real time in said enroute passage as said two or more vehicles progress along said strategic route plans.

The said vehicle tracking and monitoring means is configured to identify and communicate with the forthcoming traffic signal vide said wireless data communication system and alter the traffic signal from a prescribed distance for providing a clear passage and again altering the traffic signal after the crossover of the said two or more vehicles.

3. Eichhorst el al. [3]

Traffic control preemption systems assist authorized vehicles (police, fire and other public safety or transit vehicles) through signalized intersections by making preemption requests to the intersection controllers that control the traffic lights at the intersections. The intersection controller may respond to the preemption request from the vehicle by changing the intersection lights to green in the direction of travel of the approaching vehicle. This system improves the response time of public safety personnel, while reducing dangerous situations at intersections when an emergency vehicle is trying to cross on a red light in addition, speed and schedule efficiency can be improved for transit vehicles.

In a disclosed method of processing traffic signal priority requests, traffic signal priority requests from a vehicle are received at an intersection. A number of stopped vehicles at the intersection and on an approach to the intersection is determined in response to receiving each priority request. An activation threshold is computed as a function of an estimated-time-of-arrival (ETA) threshold and the number of stopped vehicles. A vehicle ETA of the vehicle at the intersection is determined in response to each priority request. In response to the vehicle ETA being less than the activation threshold, the priority request is submitted for

preemption service processing at the intersection. In response to the vehicle ETA being greater than the activation threshold, submission of the priority request for preemption service processing at the intersection is bypassed. A disclosed system for processing traffic signal priority requests includes a priority request receiver that is configured and arranged to receive priority requests. A data collector is configured and arranged to provide data indicative of vehicles at the intersection. A processor is coupled to the priority request receiver and to the data collector, and a memory coupled to the processor. The memory is configured with instructions that when executed by the processor cause the processor to receive traffic signal priority requests from a vehicle. The processor determines the number of stopped vehicles at the intersection and on an approach to the intersection in response to receiving each priority request and using the data indicative of vehicles at an intersection. An activation threshold is computed as a function of an estimated-time-of-arrival (ETA) threshold and the number of stopped vehicles. A vehicle ETA of the vehicle at the intersection is determined in response to each priority request. In response to the vehicle ETA being less than the activation threshold, the priority request is submitted for preemption service processing at the Page 2 intersection. In response to the vehicle ETA being greater than the activation threshold, submission of the priority request for preemption service processing at the intersection is bypassed.

4. Jerath et al. [4]

In densely populated countries, people face many issues related to traffic congestion traffic jams. On an average, people waste at least 2 hours of their day stuck in traffic. Even in case of emergencies, ambulances mostly are unable to reach hospitals on time due to mismanagement of traffic. So, there is a need to develop an effective system for traffic control and management systems.

The present invention herein discloses an emergency traffic management system using raspberry pi. The present system consists of a Raspberry Pi 3 B+, ESP32 microcontroller, I2C protocol, 16*2 LCD, blue and red Led and jumper wires. the present system allows the traffic lights to communicate with the ambulance and with each other and guide the ambulance by knowing the distance of the ambulance and the intersection and the traffic density.

In one embodiment, the present system allows the traffic lights to communicate with the ambulance and with each other and guide the ambulance by knowing the distance of the ambulance and the intersection and the traffic density. In another embodiment, the Google API continuously provides the traffic density data to the driver. In yet another embodiment, based on a predefined threshold the driver takes the decision of whether the emergency system needs to

be activated or not. In yet another embodiment, once the system is activated the ambulance sends the message to a nearby traffic light through a server which further passes the message to the next and this process continues until the ambulance reaches the desired destination. In yet another embodiment, the system introduces new blue color traffic lights for emergency services. In another embodiment, an ambulance driver is having a manual control of the traffic light system.

5. Donald et al. [5]

AI Programming is considered as a primary bridge connecting the physical and human world. AI Technique becomes one of the most important and interesting fields of research through which researchers hope to control all everyday usages via programming. AI programming includes objects different from technological environments, or even living organisms as well as devices that are already deeply embedded in the technological environments such as 4-G, 5-G phones or advanced vehicles.

By integrating computational abilities in all kinds of things and living organisms, it will be possible to provide a big leap in many sectors: Education, Training, Agriculture, research Health, military, home, entertainment and so on. Therefore, the study of AI programming technology is very important and interesting.

The invention is to A computer-implemented method of communicating information between a vehicle and a traffic signal, comprising: operatively connecting the vehicle to a network by a user device; operatively connecting an interaction module to the network, the interaction module operatively connected to the traffic signal, wherein the traffic signal is remote from the interaction module; and communicating the information between the vehicle and the traffic signal using the interaction module as an intermediary. The information includes processing the information by a controller that is also connected to the interaction module.

6. Sayyed et al. [6]

Traffic congestion is turning out to be a serious concern with issues causing a chain of inter-related problems such as fuel loss, and productivity loss which includes the man-hours lost on waiting in jammed roads, pollution, and accidents occurring due to road rage. The present system uses traffic signals: which are pre-timed; as a result, lanes with relatively less traffic density have the same green time when compared to lanes with increasing traffic density, which is cumbersome. Hence, a proposed system namely a smart traffic management system using image processing is a significant and efficient tool in handling such dynamic traffic conditions.

The proposed system works on two modes of operation-the "conventional mode "which represents the existing system, and the “image processing mode”. As image processing is affected by environmental factors, during these unfavourable conditions we have designed it to revert back to the "conventional mode" so that the traffic management does not get affected. The system operates in this mode until the conditions become favourable for image processing. Also, in a few cases, traffic policemen are required to manually manage the traffic. In large junctions, more personnel are required to regulate the traffic flow. This becomes a physically exhausting process. As stated earlier, where image processing is not applicable, the standard GO (GREEN) time for a particular lane will be followed. If the policemen have to manually manage the traffic, and find that the entire GO (GREEN) time need not be allocated for a particular lane then the master control button is provided for the cycle to shift to the next lane. Thereby, the exhaustion of the personnel is reduced as the entire process is being controlled at a centralized location and the number of traffic policemen required to monitor the junction is also reduced.

7. Choudekar et al. [7]

In the modern world, control and maintenance of traffic is becoming a very difficult task and that cannot be achieved by the limited resources. Even at emergency times, the traffic may get stuck. In the developed system, a camera is placed at each traffic point to monitor the density of the vehicles at each lane. The captured image is sent to the Transportation Management Centre (TMC) where the images are processed using MATLAB simulation software. The images are processed by different modes to improve the control at night times also. The MATLAB simulation software gives the number of vehicles standing in each lane. The time interval of the traffic light will be modified to clear the lane which has the highest numbers of vehicles.

In the developed system, the traffic is monitored and controlled using real-time video image processing. A camera is placed at each traffic signal pole that captures real-time images and sends to the TMC. The digital image information taken by the camera is processed by the MATLAB simulation software and which selects the appropriate timing according to the number of vehicles to reduce the traffic jam. The selected timing is sent to the Arduino board. The controller placed in the Arduino board controls the traffic signal ON and OFF with respect to the data given from the MATLAB output. The above-explained process is shown as a block diagram in Figure 1. The image processing is done by converting the analog image into digital binary images and then the noise is cancelled by removing lower pixel objects in the image. After these procedures, the block is counted and according to the number of blocks the time interval of every signal is controlled.

8. Sakahre et al. [8]

Traffic congestion is the major challenge faced by any country due to improper traffic management on roads - basic connective path. Such congestion of traffic arises due to violation of traffic rules by the public, heavy population, lack of road maintenance and also lack of technology.

As traffic rule violation is the basic reason for vehicle accidents ranging about 35 vehicles per 1000 vehicles which even leads to the death of pedestrians and drivers themselves, this invention focuses on traffic management by imparting technology to control traffic rule violation.

Any vehicle violating the rule will be identified by RFID tags and a penalty alert is sent to vehicle owners using GSM technology. This avoids any rule violator to get escaped from penalized thereby mitigating any traffic violation subsequently thereby providing efficient traffic management.

9. H. Razalli et al. [9]

In [9], H. Razalli et al. presented a system of combination of an automatic emergency vehicle recognition classification method with a real-time tracking algorithm. They proposed an algorithm to detect a moving emergency vehicle in a traffic surveillance camera using a combination of HSV (Hue Saturation and Value) color segmentation and SVM (Support Vector Machine). The application is divided into two modules, the first module detects and extracts vehicles, and the second module recognizes and classifies them. The detection of vehicles is done by using color segmentation which adopts HSV (Hue Saturation and Value) and RGB (Red, Green, and Blue) color models to characterize the emergency vehicle siren light to use as a detection feature in emergency vehicle recognition. The vehicle classification is achieved by constructing a classifier using SVM (Support Vector Machine). Overall, on videos captured during the daytime, the recognition results achieved up to 97.3% accuracy and the recognition rate was almost 92%, with low processing time.

2.2 Plan of Work

We divided our product in 5 distinct modules. Each module is described below:

Module - 1 : Vehicle Density Estimation

It includes the input of traffic images that undergo image pre-processing, feature extraction, traffic density estimation, output of vehicle count and traffic Vehicles classification.

Module - 2 : Emergency Service Vehicles Detection

It detects the presence of emergency vehicles on any lane.

Module - 3 : Embedded Controller/ Simulator

In this module, the output of Module-1 (specifically vehicle count) and Output of Module-2 will be taken as an input into the software-based Simulator with an aim to set the time of the signal dynamically.

Module - 4 : Android Application

The mobile android-based application is a controller which can be switched from manual mode to automatic mode and vice versa based on necessity.

Module - 5 : Website

The website is an administrative platform to manage the users and junctions.

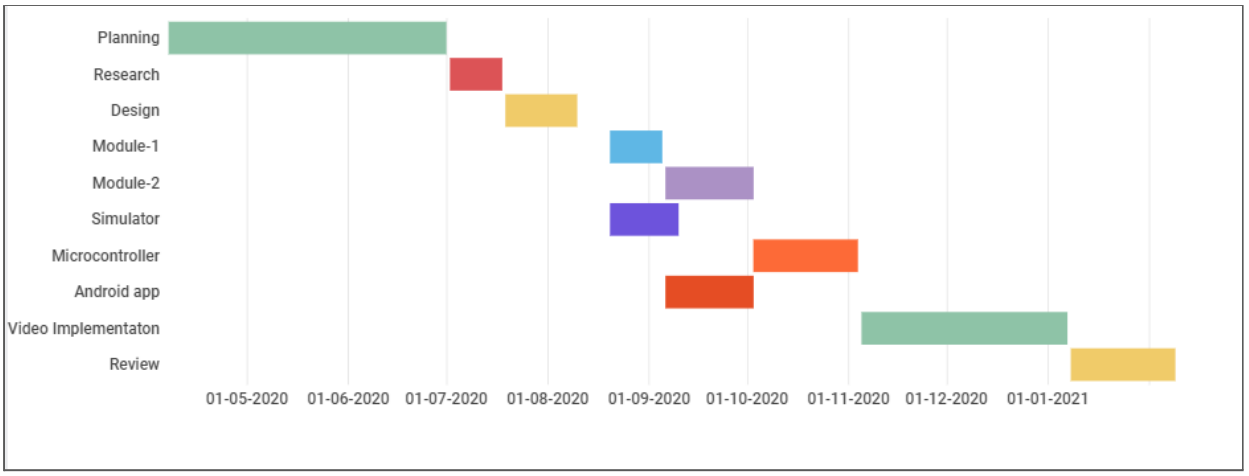


Fig. 1 : Gantt chart

CHAPTER – 3 Requirement Analysis and Design

3.1 AEIOU Canvas

AEIOU Framework: AEIOU is a heuristic to help interpret observations gathered by ethnographic practise in the industry. Its two primary functions are to code data, and to develop building blocks of models that will ultimately address the objectives and issues of a client.

Fig. 2 : AEIOU Canvas

Activity:

Activities are goal-directed sets of actions—paths towards things people want to accomplish. What are the modes people work in, and the specific activities and processes they go through?

- People driving vehicles
- Traffic police managing traffic

- Beggars begging
- Vendors selling
- Traffic light switching colours
- Pedestrian crossing road
- People waiting at the signal
- Pedestrian walking on the footpath
- Auto waiting for passengers
- Animals blocking road

Environment:

Environments include the entire arena where activities take place. What is the character and function of the space overall, of each individual's spaces, and of shared spaces?

- Divider
- Noisy environment
- Greenery
- Polluted air
- Chaotic Environment
- Public washroom
- Bus stand

Interaction:

Interactions are between a person and someone or something else; they are the building blocks of activities. What is the nature of routine and special interactions between people; between people and objects in their environment, and across distances?

- Police traffic talking among themselves
- Beggars asking people
- Traffic police catching offenders
- Auto driver calling people
- People bargaining with vendors
- People arguing and fighting

Object:

Objects are building blocks of the environment, key elements sometimes put to complex or unintended uses (thus changing their function, meaning and context). What are the objects and devices people have in their environments and how do they relate to their activities?

- Traffic light
- Camera
- Mobile phone
- Vehicle
- Reflectors
- Camera
- Street light
- Dustbin
- Solar panel
- Barricade
- Billboard

User:

Users are the people whose behaviours, preferences, and needs are being observed. Who is there? What are their roles and relationships? What are their values and prejudices?

- Traffic police
- Commuter
- Pedestrian

3.2 Empathy Canvas

Empathy Canvas contains the information of users who uses the product, stakeholders who are concerned with our project directly or indirectly. To define any user-centric problem we need to know the user properly. That was what this canvas was all about. For this, we observe the railway station properly. After then we note down some activity, some happy things and some sad things too. In this Canvas, we write about users, stakeholders, activity and story based on what we observe. An Empathy map is a collaborative tool team can use to gain a deeper insight into their customers. Much like a user persona, an empathy map can represent a group of users, such as a customer segment.

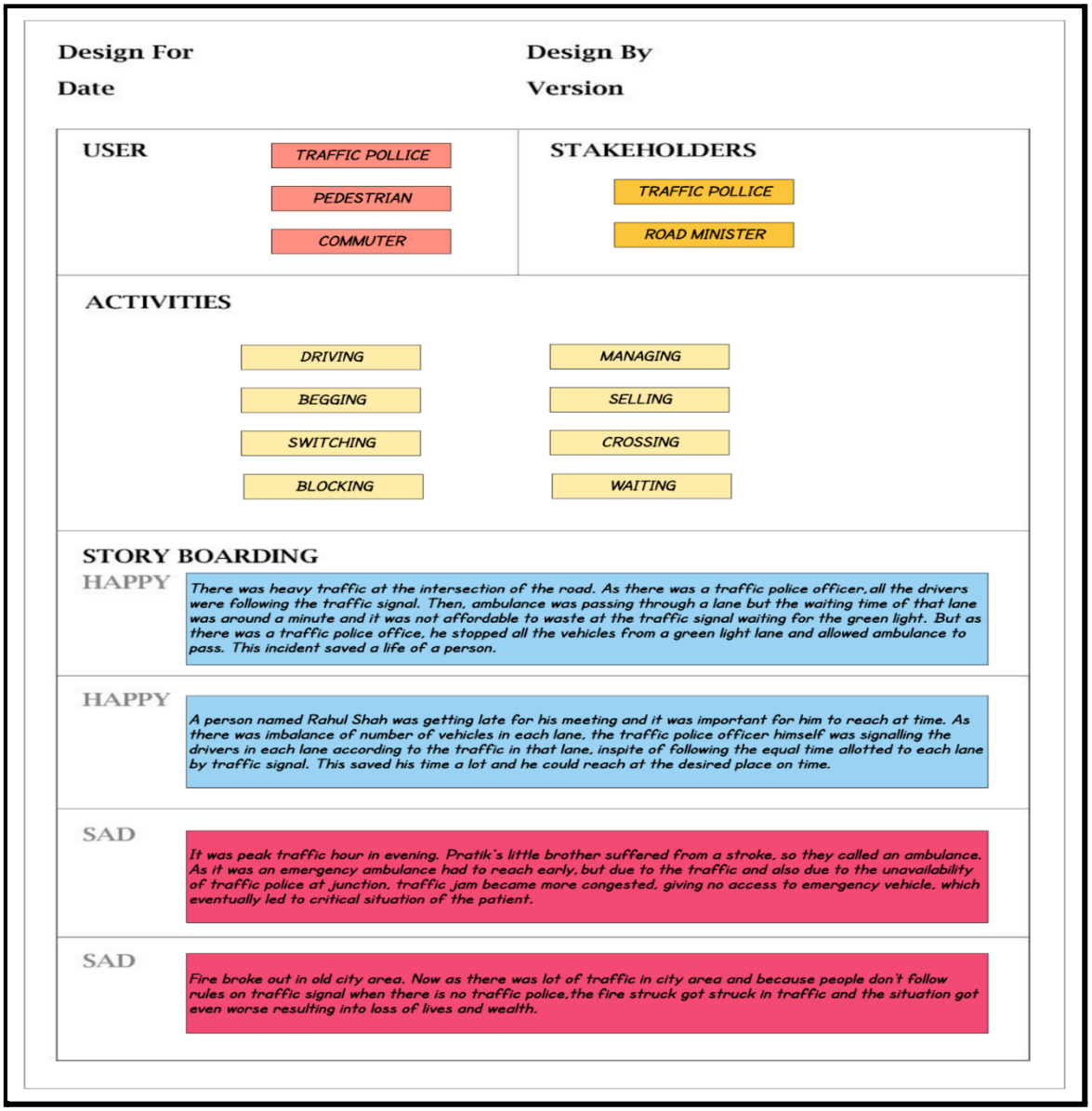


Fig. 3 : Empathy Canvas

User :

Users are the people whose behaviours, preferences, and needs are being observed. Who is there? What are their roles and relationships? What are their values and prejudices?

- Traffic police
- Commuter
- Pedestrian

Stakeholder:

Stakeholders mean a person or organization with an interest.

- Traffic Police
- Road Minister

Activity:

Activities are goal-directed sets of actions—paths towards things people want to accomplish. What are the modes people work in, and the specific activities and processes they go through?

- Driving
- Managing
- Begging
- Selling
- Switching
- Crossing
- Blocking
- Waiting

Storyboarding:

These are the happy and sad stories of passengers which we have observed during visiting traffic signals

Happy stories :

1. There was heavy traffic at the intersection of the road. As there was a traffic police officer all the drivers were following the traffic signal. The ambulance was passing through a lane but the waiting time of that lane was around a minute and it was not affordable to waste at the traffic signal waiting for the green light. But as there was a traffic police officer he stopped all the vehicles from the green light lane and allowed the ambulance to pass. The incident saved a life of a person

2. A person named Rahul shah was getting late for his meeting and it was important for him to reach there on time. As there was an imbalance of the number of vehicles in each lane the traffic police offer himself was signalling the drivers in each lane. In spite of following the equal time allotted to each lane by a traffic signal. This saved his time a lot and he could reach the desired place on time

Sad stories :

1. It was peak traffic hour in the evening. Pratik's little brother suffered from a stroke, so they called an ambulance. As it was an emergency, ambulances had to reach early, but due to traffic and also due to unavailability of traffic police at the junction, traffic jams became more congested, giving no access to emergency vehicles, which eventually led to critical situations for the patient.

2. The fire broke out in the city area. Now as there was a lot of traffic in the city area and because people don't follow rules on traffic signals when there is no traffic police the fire truck got stuck in traffic and the situation got even worse resulting in loss of wealth and death.

3.3 Ideation Canvas

In Design ideation: the conceptual sketch in the digital age (2005) Ben Johnson defined ideation as "a matter of generating, developing and communicating ideas." His definition aligns pretty well with how I've been evangelizing and practising the discipline. However, I place a much stronger emphasis on development and communication than on idea generation.

In most organizations, there is no shortage of ideas. Ideation helps you work through ideas quickly so you know where and how to apply it.

Ideation is a process for bringing ideas to life. While ideation may include the creation of original ideas (generation), its primary focus is working through concepts (development) to gain insights, understand implications, gauge feasibility, and give teams a deeper understanding of potential product experiences (communication).

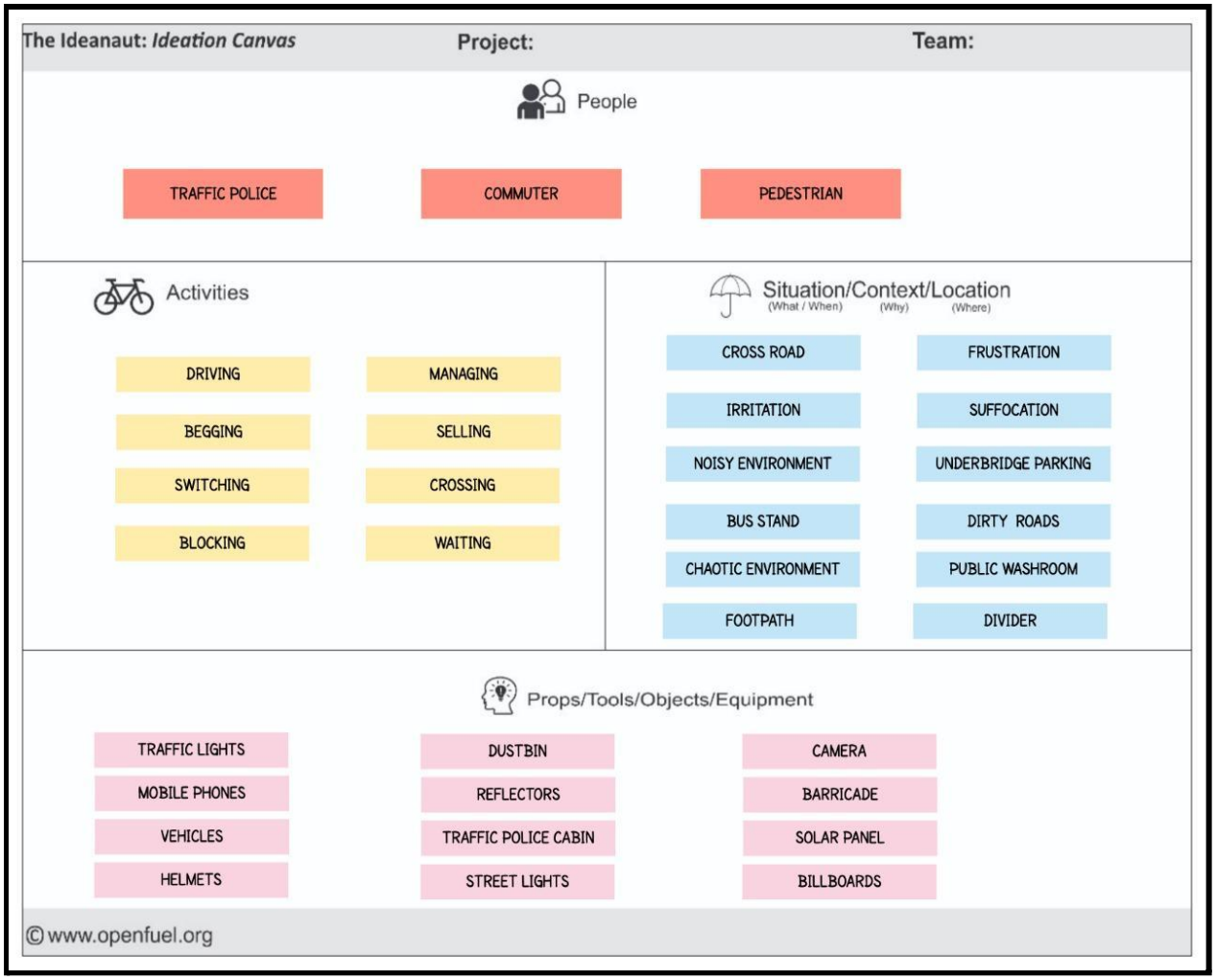


Fig. 4 : Ideation Canvas

People:

People are the people whose behaviours, preferences, and needs are being observed. Who is there? What are their roles and relationships? What are their values and prejudices?

- Traffic police
- Commuter
- Pedestrian

Activities:

Activities are goal-directed sets of actions—paths towards things people want to accomplish. What are the modes people work in, and the specific activities and processes they go through?

- Driving
- Managing
- Begging
- Selling
- Switching
- Crossing
- Blocking
- Waiting

Situation/Context/Location:

Activities in a particular situation with respect to its location is as follows:

- Cross Road
- Irritation
- Noisy Environment
- Bust stand
- Chaotic environment
- Footpath
- Frustration
- Suffocation
- Underbridge parking
- Dirty roads

- Public washroom
- Divider

Props:

Props are an integral part of the new product development process as they help validate a design, test features and conduct trials.

- Dustbin
- Camera
- Reflectors
- Traffic police cabin
- Street light
- Barricade
- Solar panel
- Billboard

3.4 PDC Canvas

Product Development Canvas (PDC) is the Canvas containing all the information about the project and product. PDC contains the purpose of making this product also users or people concerned with this product and project.

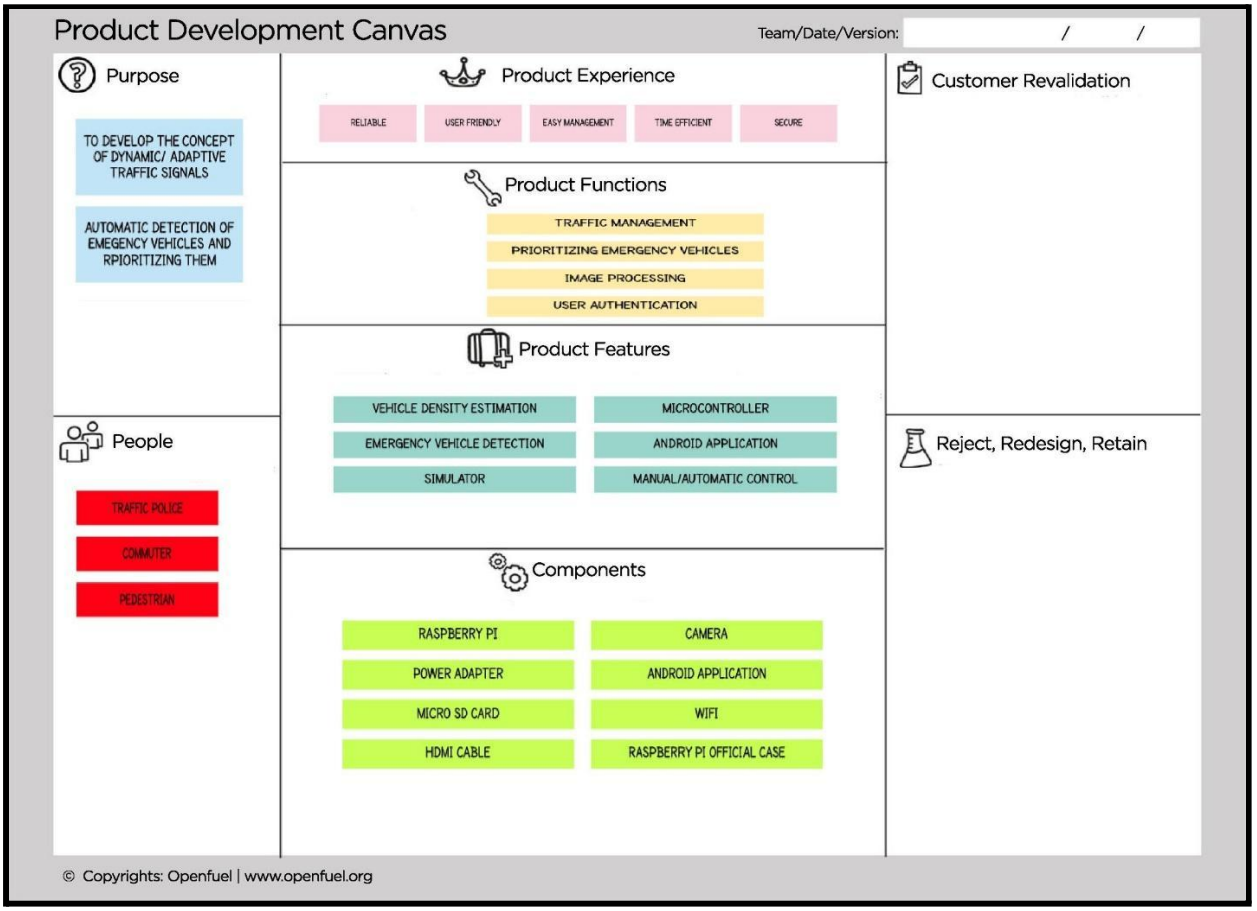


Fig. 5 : Product Development Canvas

Purpose:

The purpose is the reason for which the product is designed.

- To develop the concept of dynamic/adaptive traffic signal
- Automatic detection of emergency vehicles and prioritizing them

Product experience:

Product experience is the entire set of effects that are elicited by the interaction between a user and a product, including the degree to which all our senses are gratified; the meanings we attach to the product; the feelings and emotions that are elicited (emotional experience).

- Traffic Management
- Prioritising emergency vehicles
- Image processing
- User authentication

Product Functions:

Product functions are the activities or tasks which are performed by the product.

- Traffic management
- Prioritizing emergency vehicles
- Image processing
- User authentication

Product Features:

Product features are characteristics of the product that describes its appearance, components, and capabilities.

- Vehicle Density estimation
- Emergency vehicle detection
- Simulator
- Microcontroller
- Android application
- Manual/automatic control

People:

People enlist all groups of individuals who will be affected or benefitted by the use of the product.

- Traffic Police
- Commuter
- Pedestrian

Components:

Components are all the elements used to design a particular service or product.

- Raspberry pi
- Camera
- Power adapter
- Android application
- Micro sd card wifi
- HDMI cable
- Wifi
- Raspberry pi official case

3.5 DIAGRAMMATIC REPRESENTATION OF FLOW OF IMPLEMENTATION

3.5.1 E-R Diagram

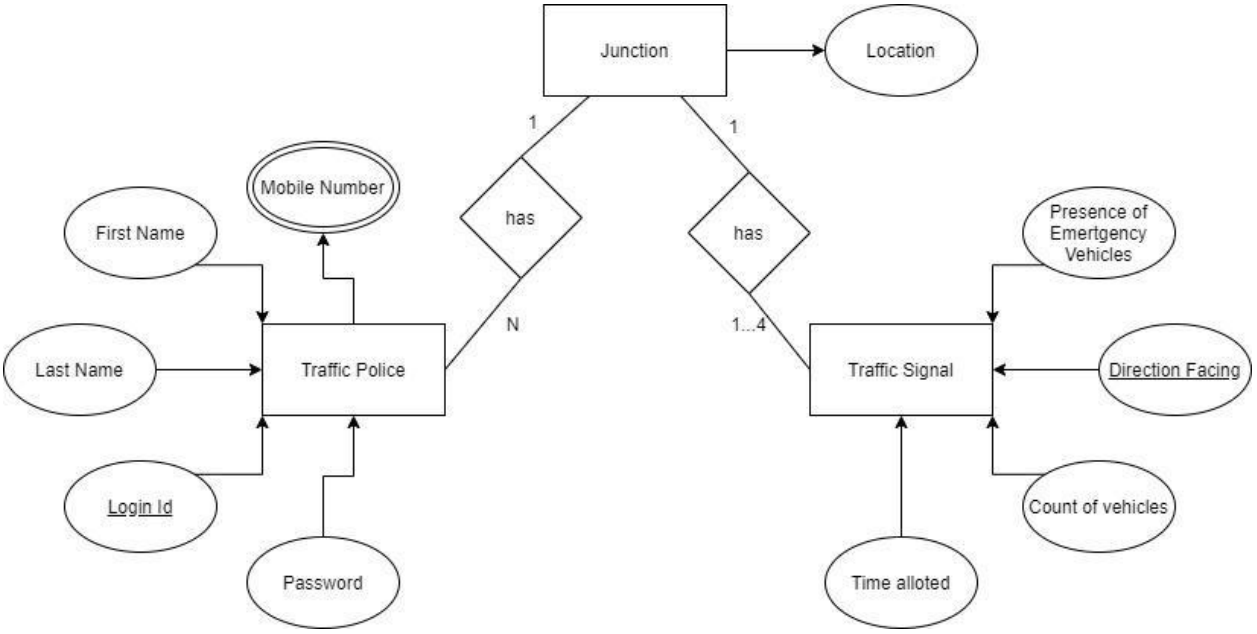


Fig. 6 : ER Diagram

3.5.2 Class Diagram

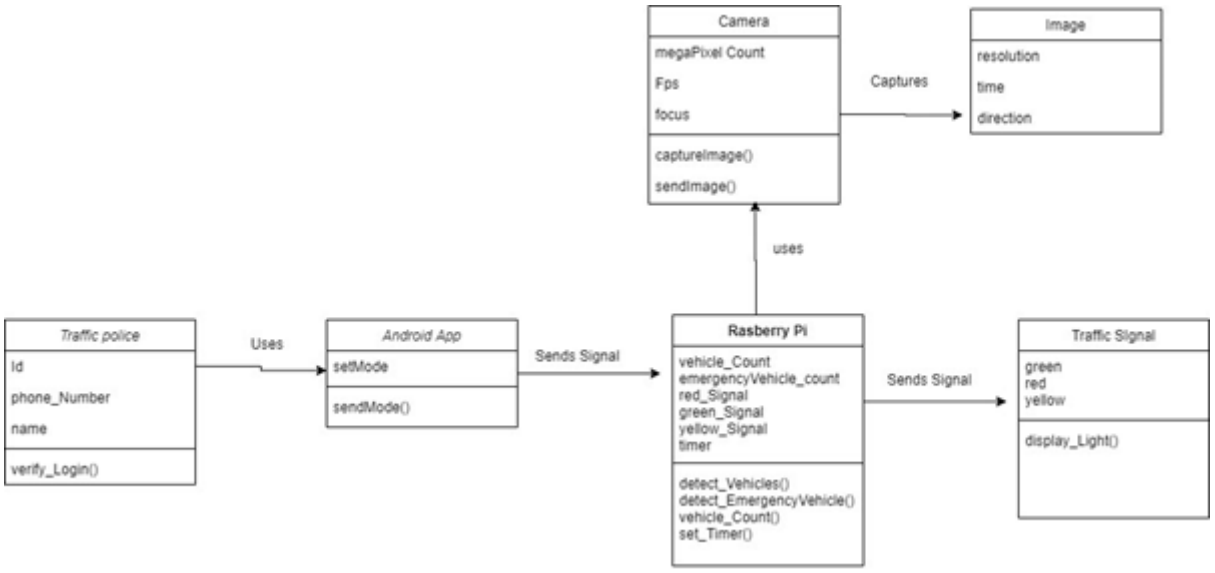


Fig. 7 : Class Diagram

3.5.3 Sequence Diagram

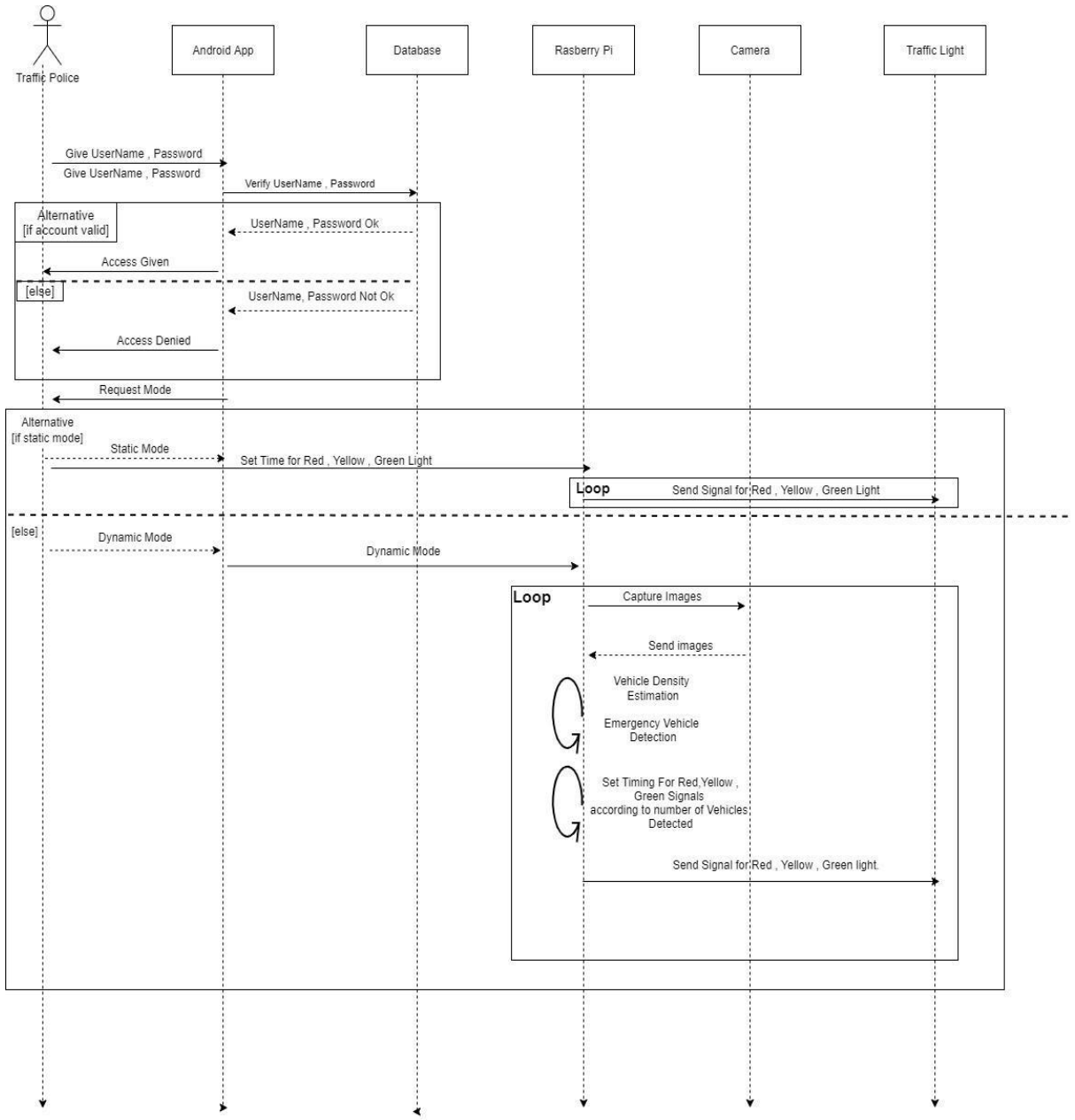


Fig. 8 : Sequence Diagram

3.5.4 Use-Case Diagram

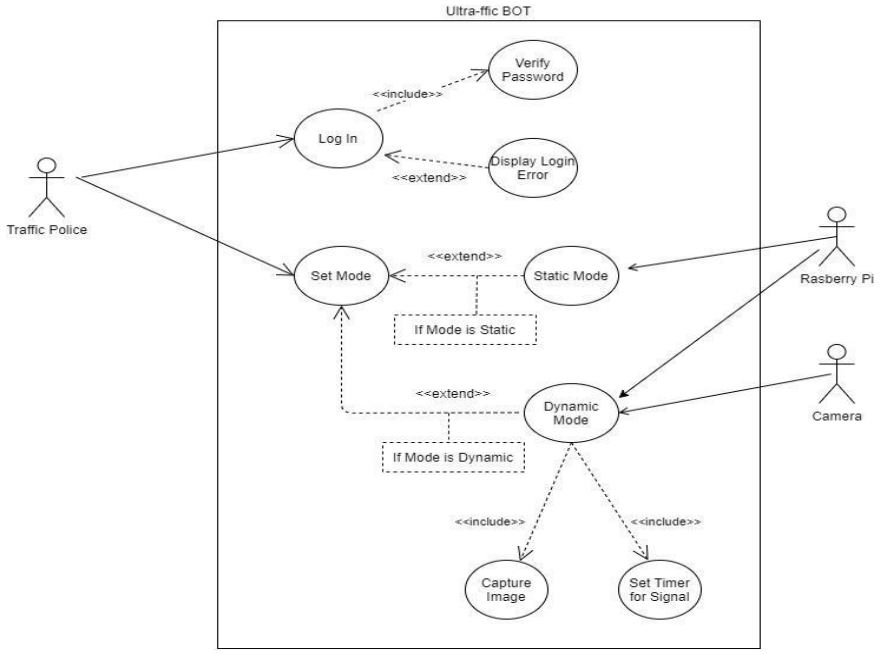


Fig. 9 : Use Case Diagram

3.5.5 Activity Diagram

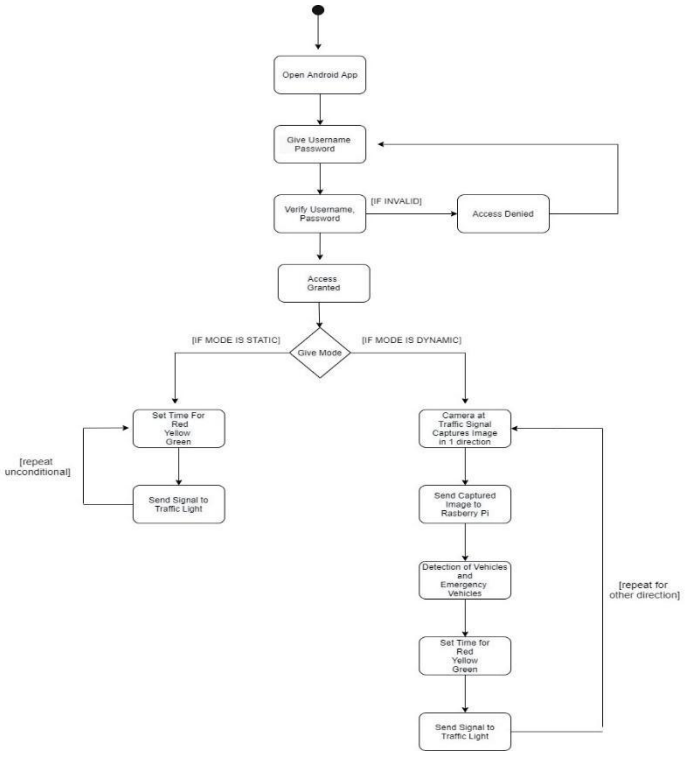


Fig. 10 : Activity Diagram

CHAPTER – 4 TOOLS AND TECHNOLOGY USED

OpenCV (Image Processing)

OpenCV (Open Source Computer Vision Library) is an open-source computer vision and machine learning software library. OpenCV was built to provide a common infrastructure for computer vision applications and to accelerate the use of machine perception in commercial products.

YOLOv3 (Object detection)

YOLO V3 is an improvement over previous **YOLO** detection networks. Compared to prior versions, it features multi-scale detection, a stronger feature extractor network, and some changes in the loss function. As a result, this network can now detect many more targets from big to small.

LabelImg (Annotation tool)

LabelImg is a graphical image **annotation** tool. It is written in Python and uses Qt for its graphical interface. **Annotations** are saved as XML files in PASCAL VOC format, the format used by ImageNet. Besides, it also supports the YOLO format.

Proteus 8 Professional (For Simulator)

Proteus 8 Professional is software that can be used to draw schematics, PCB layout, code and even simulate the schematic. It is developed by Labcenter Electronics Ltd

ISIS Proteus 7 Professional (For Simulator)

Proteus 7.0 is a Virtual System Modelling (VSM) that combines circuit simulation, animated components and microprocessor models to co-simulate the complete microcontroller based designs. This is the perfect tool for engineers to test their microcontroller designs before constructing a physical prototype in real-time

Fritzing (For Simulator)

Fritzing is an open-source initiative to develop amateur or hobby CAD software for the design of electronics hardware, to support designers and artists ready to move from experimenting with a prototype to building a more permanent circuit.

Android Studio 4.0.1 (For Mobile Application)

Android Studio is the official integrated development environment for Google's Android operating system, built on JetBrains' IntelliJ IDEA software and designed specifically for Android development.

Vysor (For Mobile Application)

Vysor lets you view and controls your Android on your computer. Use apps, play games, control your Android with your mouse and keyboard. Go wireless, and mirror your Android to your desktop; great for presentations. Vysor Share also lets you share your screen with others for remote assistance.

Yolov4 tiny (Emergency vehicle detection model)

YOLOv4-tiny is the compressed version of YOLOv4. YOLOv4-tiny model weights are 16MB. YOLOv4-tiny trains on 350 images in 1 hour on a Tesla P100.

Darknet (Neural network framework)

Darknet is an open-source neural network framework written in C and CUDA. It is fast, easy to install, and supports CPU and GPU computation.

Roboflow (Transform raw images to CV model)

Roboflow provides the key infrastructure between labelling and training that developers previously had to build themselves.

Google Colab GPU (Faster computation)

Google Colab allows anybody to write and execute arbitrary python code through the browser

and is especially well suited to machine learning, data analysis and education.

Raspberry Pi (For Hardware)

The Raspberry Pi is a credit card-sized computer that plugs into your TV and a keyboard. It is a capable little computer that can be used in electronics projects, and for many of the things that your desktop PC does, like spreadsheets, word processing, browsing the internet, and playing games. It also plays high-definition video. We want to see it being used by kids all over the world to learn to program.

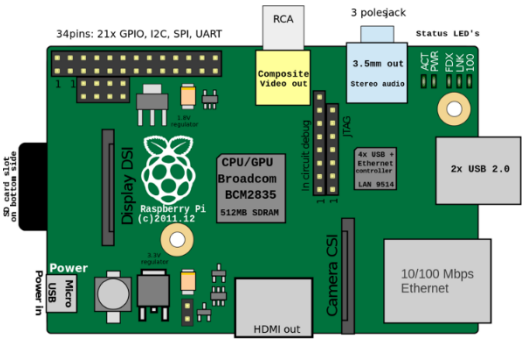


Fig. 11 : Raspberry pi

Raspberry Pi Camera (For Hardware)

The Raspberry Pi camera module can be used to take high-definition video, as well as still photographs. Supports 1080p30, 720p60 and VGA90 video modes.

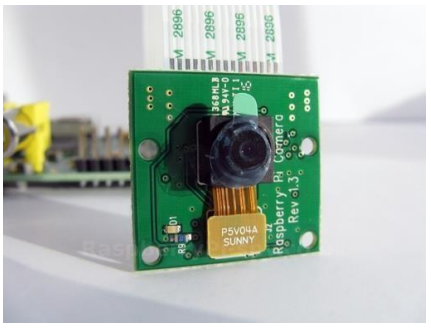


Fig. 12: Raspberry pi Camera

Bootstrap v4.0.1 (For Website)

The Bootstrap is a free and open-source CSS framework directed at responsive, mobile-first front-end web development. It contains CSS- and JavaScript-based design templates for typography, forms, buttons, navigation, and other interface components.

Firestore (for handling data of android app and website)

Cloud Firestore is a flexible, scalable database for mobile, web, and server development from Firebase and Google Cloud. Like Firebase Realtime Database, it keeps your data in-sync across client apps through real-time listeners and offers offline support for mobile and web so you can build responsive apps that work regardless of network latency or Internet connectivity. Cloud Firestore also offers seamless integration with other Firebase and Google Cloud products, including Cloud Functions.

Google Cloud Platform (for deployment on the server)

Google Cloud Platform, offered by Google, is a suite of cloud computing services that runs on the same infrastructure that Google uses internally for its end-user products, such as Google Search, Gmail, file storage, and YouTube.

CHAPTER – 5 IMPLEMENTATION

5.1 ACTUAL IMPLEMENTATION

Module - 1 : Vehicle Density Estimation

This module takes traffic images as an input that undergoes image pre-processing for feature extraction. We get vehicle count as an output.

In addition, vehicles and other surrounding elements (such as a person(s), animal(s), etc..) are also classified into various specific classes.

We have used a deep convolutional neural network called Darknet-53 architecture. It is a 53 layer neural network trained on ImageNet for classification combined with detection layers making the total network 106 layers deep.

Module - 2 : Emergency Vehicle Detection

Step 1: Data Collection

- Images were downloaded from the internet
- Videos of the ambulance were recorded using from all angle using a drone

Step 2: Data Preprocessing

- Video Slicing: Obtained frames (interval: 3s) by slicing the recorded video and redundant images were removed.
- Resizing: Because yolo requires the size of input images to be in multiples of 32, all images were resized to 412x412
- Orientation: Due to resizing, annotation of all images were oriented according to resized images.

Step 3: Annotation

- LabelImg software was used to annotate (bounding box coordinates) all the images in yolo format

Step 4: Custom object training of yolov4 model

- We trained yolov4-tiny for 2 classes namely: Ambulance and Firetruck.
- These two classes are reflected in /content/darknet/data/obj.names file

Step 5: Testing

Module - 3 : Embedded Controller/ Simulator

In this module, we started our implementation on Simulator. We started by making a virtual circuit of the proposed system on fritzing software. Later we tried various microcontrollers including Arduino Uno and raspberry pi. We created the proposed system in a virtual environment of the Proteus 8 professional software.

We then modified the simulator project to integrate with module-4 (android application) of our project. The machine learning models were then deployed on Raspberry pi. To satisfy more processing and GPU needs of the machine learning model, we created a server on Google Cloud Platform (GCP).

The server acts as a central processing unit for the whole project, processing is carried out in the server, also it is integrated with the Raspberry pi, Firestore database, Android application and the website.

Module - 4 : Android Application

The mobile android-based application is a controller which can be switched from **manual mode** to **automatic mode** and vice versa based on necessity.

In static mode, time for red, yellow and green light is set. In dynamic mode the ve. The mode and time provided are sent to the simulator for further action.

We created the android application login and registration page by integrating it with the firebase database. Then we coded the java application for connectivity with the microcontroller. It will be used as a control centre for the traffic signal.

Module - 5 : Website

The website module is an administrative module used to manage the traffic police and traffic junctions.

With the help of this website, we can assign or de-assign a junction to a user, we can approve or disapprove a user, we can add or remove a user as an admin, and we can add a junction or view the allocated junctions.

The website is integrated with a Firestore database that handles the login credentials and makes the admin features available only to an admin user.

5.3 SNAPSHOTS / TESTING AND VERIFICATION

Module - 1 : Vehicle Density Estimation

Fig. 13 (a) : Input image of Module1

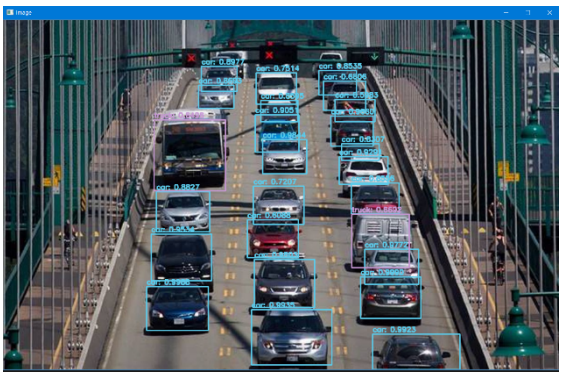


Fig. 13 (b) : Output image of Module1

- The image shown in fig 13(a) is fed as an input for module 1(vehicle density estimation) on which the Darknet-53 framework will undergo object detection through various layers of the neural network through that car and other vehicles will be identified.
- The output of module 1 we obtained after applying these operations is shown in fig 13(b) with bounding boxes formed around the vehicles such as cars and trucks.

```
PS C:\Users\bardo\Desktop\yolo-object-detection\yolo-object-detection> python yolo.py -i images\traffic-4.jpg -y yolo-coco
[INFO] loading YOLO from disk...
[INFO] YOLO took 2.769729 seconds
Cars: 23
Truck: 2
```

Fig. 14 : Output (Count of vehicles etc.)

- The output of module 1 obtained on the command prompt is shown in fig 14 with the time taken to detect cars and trucks and even the count of cars and trucks.

Fig. 15 (a) : Input image

Fig. 15 (b) : Output image

- The image shown in fig 15(a) is fed as an input for module-1(vehicle density estimation) on which the Darknet-53 framework will undergo object detection through various layers of the neural network through that car, motorbike and person will be identified.
- The output of module 1 we obtained after applying these operations is shown in the fig 15(b) with bounding boxes formed around the vehicles such as cars, persons and motorbike.

```
PS C:\Users\bardo\Desktop\yolo-object-detection\yolo-object-detection> python yolo.py -i images\Anuvrat-dwar2.jpg -y yolo-coco

[INFO] loading YOLO from disk...
[INFO] YOLO took 3.235758 seconds
Cars: 7
Motorbike: 1
Truck: 0
Person: 2
```

Fig. 16 : Output(Count of vehicles etc.)

- The output of module 1 obtained on the command prompt is shown in fig 16 with the time taken to detect the entities and even the count of cars, trucks, motorbikes and persons.

Fig. 17 (a) : Input image

Fig. 17 (b) : Output image

- The image shown in fig 17(a) is fed as an input for module 1(vehicle density estimation) on which the Darknet-53 framework will undergo object detection through various layers of the neural network through that car and person will be identified.
- The output of module 1 we obtained after applying these operations is shown in fig 17(b) with bounding boxes formed around the vehicles such as cars and persons.

```
PS C:\Users\bardo\Desktop\yolo-object-detection\yolo-object-detection> python yolo.py -i images\Anuvrat-dwar4.jpg -y yolo-coco

[INFO] loading YOLO from disk...
[INFO] YOLO took 2.601432 seconds
Cars: 9
Person: 2
```

Fig. 18 : Output (Count of vehicles etc.)

- The output of module 1 obtained on the command prompt is shown in fig 18 with the time taken to detect the entities and even the count of cars and persons.

Fig. 19 (a) : Input image

Fig. 19 (b) : Output image

- The image shown in fig 19(a) is fed as an input for module 1(vehicle density estimation) on which the Darknet-53 framework will undergo object detection through various layers of the neural network through that motorbike, animals such as cow and person will be identified.
- The output of module 1 we obtained after applying these operations is shown in fig 19(b) with bounding boxes formed around the vehicles such as motorbike and animals such as cows.

```
PS C:\Users\bardo\Desktop\yolo-object-detection\yolo-object-detection> python yolo.py -i images\cow-1.jpg -y yolo-coco
[INFO] loading YOLO from disk...
[INFO] YOLO took 3.430862 seconds
Motorbike: 4
Person: 5
Cow: 4
```

Fig. 20 : Output (Count of vehicles etc.)

- The output of module 1 obtained on the command prompt is shown in fig 20 with the time taken to detect the entities and even the count of motorbikes, persons and animals such as cows.

Module - 2 : Emergency Vehicle Detection

- **Using binary classifier method**

  Dataset: https://www.kaggle.com/abhisheksinghblr/emergency-vehicles-identification

Fig. 21 : Example of images in dataset

- The image shown in fig 21 is an example of the images in the dataset taken for the binary classifier method.

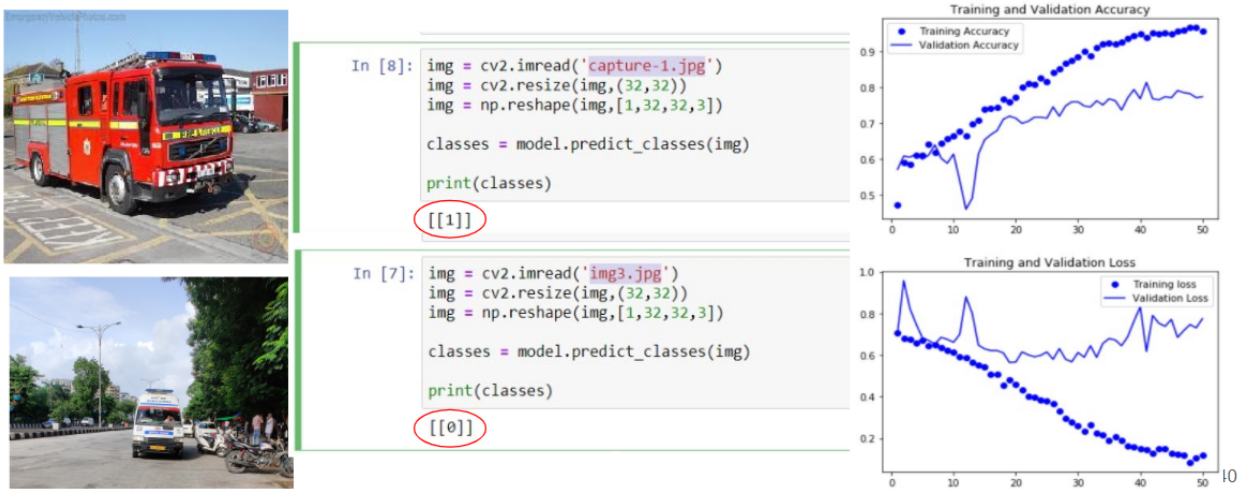


Fig. 22 : Output for binary classifier

- The output shown in fig 22 is in the form of 0 and 1 where 0 means no emergency vehicle detected and 1means an emergency vehicle is detected.
- On the right side of fig 22 the graph of training and validation accuracy and training and validation loss is obtained after training the model on an efficient net.

- **Using yolov3 model**

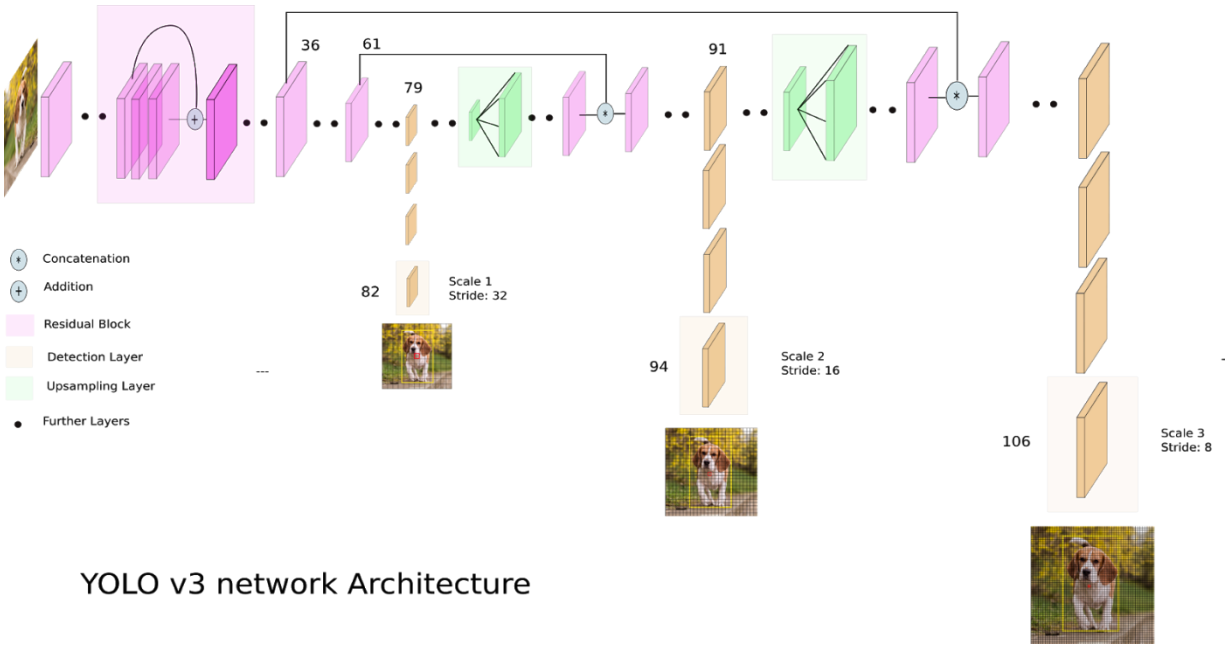


Fig. 23 : YOLO v3 network Architecture

- The image shown in fig 23 suggests the YOLOv3 makes detection at three different scales. In YOLOv3, the detection is done by applying 1 x 1 detection kernels on feature maps of three different sizes at three different places in the network.

Fig. 24 : YOLOv3 output

- The output images shown in the fig 24 is obtained by undergoing the neural network framework of YOLO v3 architecture

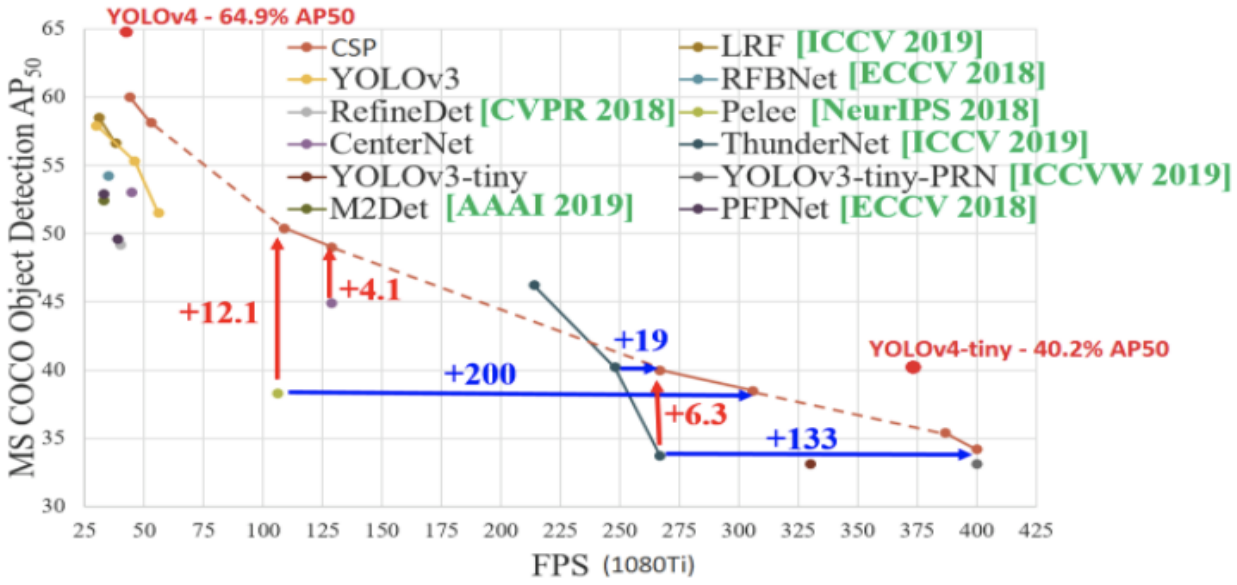


Fig. 25 : Performance of different models

- The image shown is fig 25 is a comparison between different models used for training in machine learning.

- **Using YOLOv4-tiny on darknet framework**

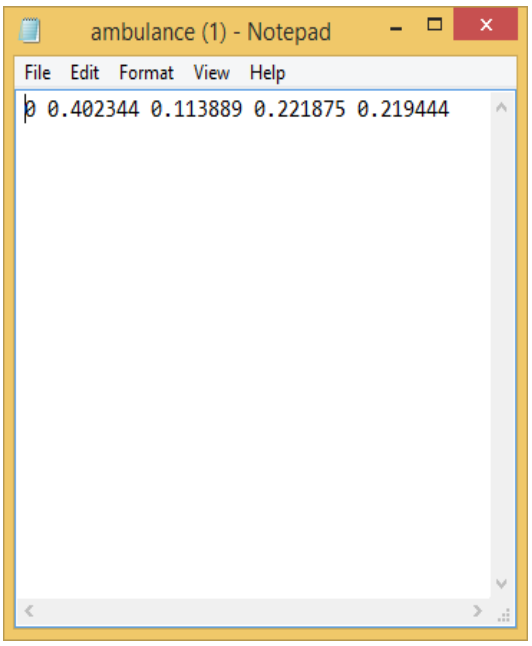


Fig. 26 : Annotation format for YOLO architecture

- Fig 26 depicts the annotation format for the images with Object class depicted by 0 or 1, followed by x and y object coordinates respectively and in the end height and width of the image is mentioned.

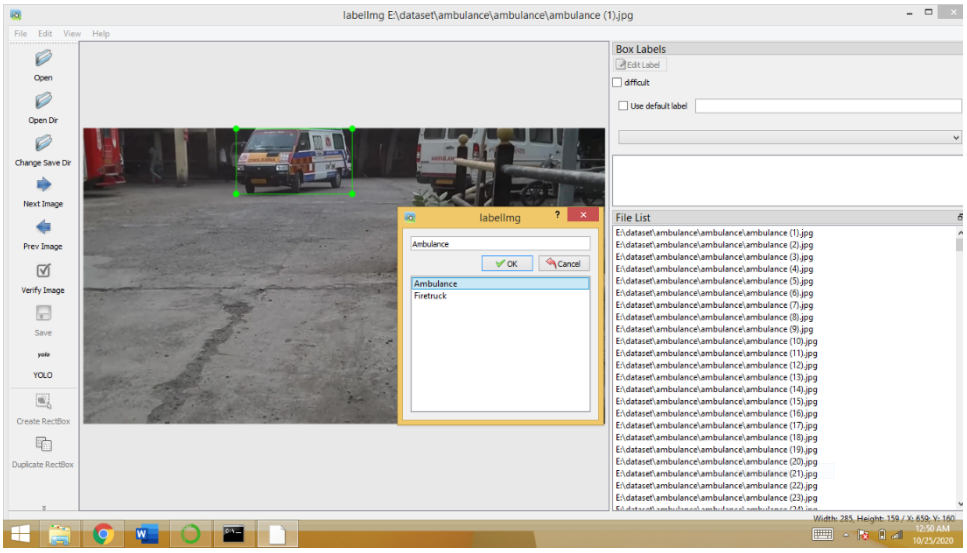


Fig. 27 : Label Image and Annotation

- The image shown in fig 27 is the depiction of the labelImg tool which is used for giving annotations to the images in the dataset.

Fig. 28 : Output (Emergency vehicles detected)

- The output images shown in fig 28 are obtained after training the model for 2 classes(ambulance and firetruck) on the darknet framework.

Fig. 29 : Output

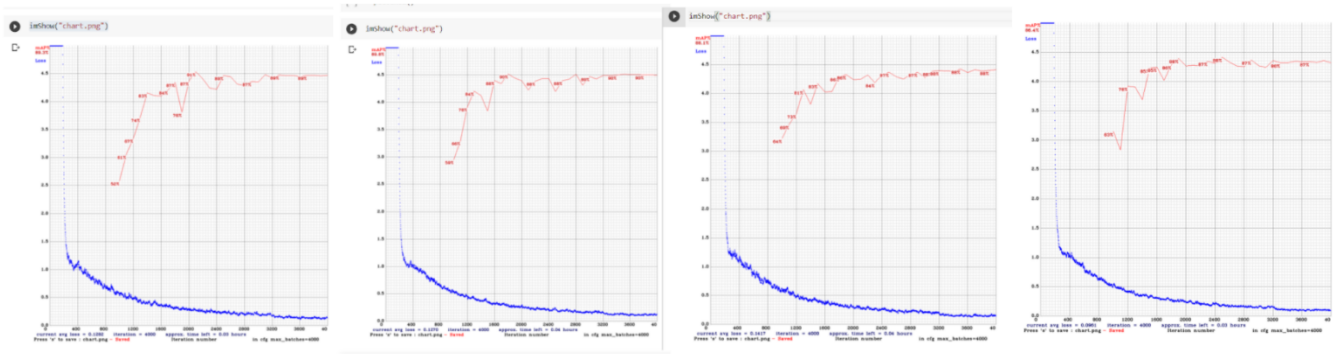


Fig. 30 : Training and validation accuracy graph

- The 4 images in fig 30 are obtained after training the model for two classes on different ratios of training and testing sets i.e. 90:10, 80:20, 60:40 and 50:50 respectively.

- **Improving Accuracy after Augmenting dataset**

- Performed data augmentation on the previous dataset.
- **Data augmentation** in data analysis are techniques used to increase the amount of data by adding slightly modified copies of already existing data or newly created synthetic data from existing data.
- It adds variance without losing the information the data carries.
- It acts as a regularizer and helps reduce overfitting when training a machine learning model as a result accuracy of the model increases.

- Following are the data augmentation steps:
    - **Rotation** - Adds variability to rotations to help a model be more resilient to camera roll.
      Parameter used : **+30° to -30°**.
    - **Shear** - Adds variability to perspective to help a model be more resilient to camera and subject pitch and yaw.
      Parameter used : ± **15° Horizontal,** ± **15° Vertical**
    - **Saturation** - Randomly adjusts the vibrancy of the colors in the images.
      Parameter used : **-19% to +19%**
    - **Exposure** - Adds variability to image brightness to help a model be more resilient to lighting and camera setting changes.
      Parameter used : **-14% to + 14%**
    - **Blur** - Adds random Gaussian blur to help a model be more resilient to camera focus.
      Parameter used : **upto 1.5px**
    - **Noise** - Adds noise to help a model be resilient to camera artefacts.
      Parameter used : **upto 6% of pixels**

Fig. 31 : Dataset Before and After Data Augmentation

Fig. 32(a) : (Module-2) Output before data augmentation

Fig. 32(b) : (Module-2) Output after data augmentation

Fig. 33(a) : (Module-2) Output before data augmentation

Fig. 33(b) : (Module-2) Output after data augmentation

Fig. 34(a) : (Module-2) Output before data augmentation

Fig. 34(b) : (Module-2) Output after data augmentation

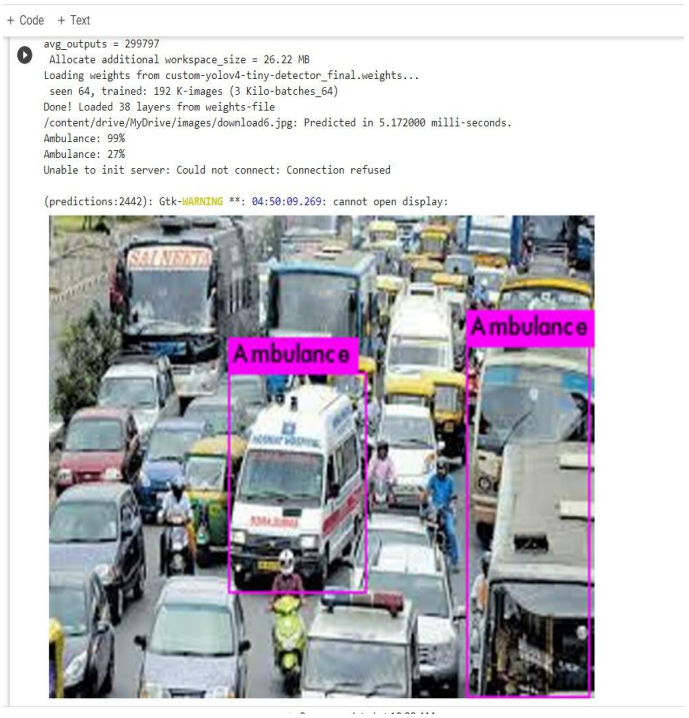


Fig. 35(a) : (Module-2) Output before data augmentation

Fig. 35(b) : (Module-2) Output after data augmentation

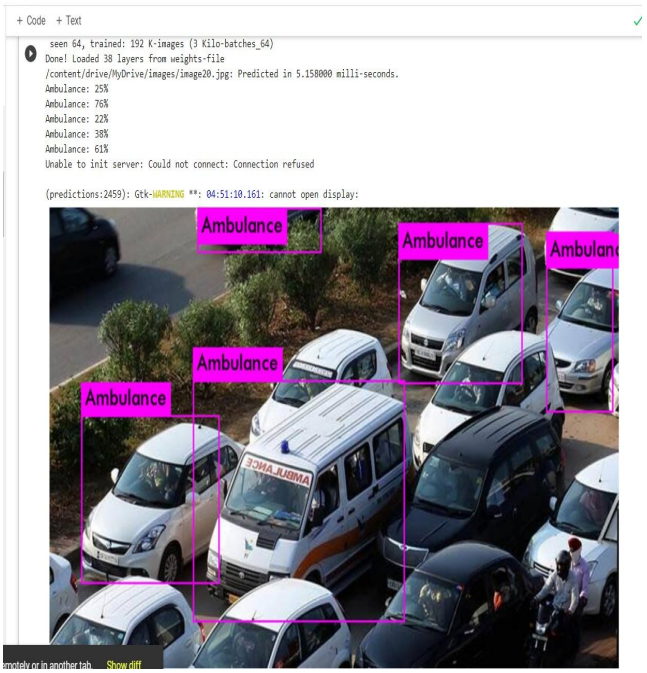


Fig. 36(a) : (Module-2) Output before data augmentation

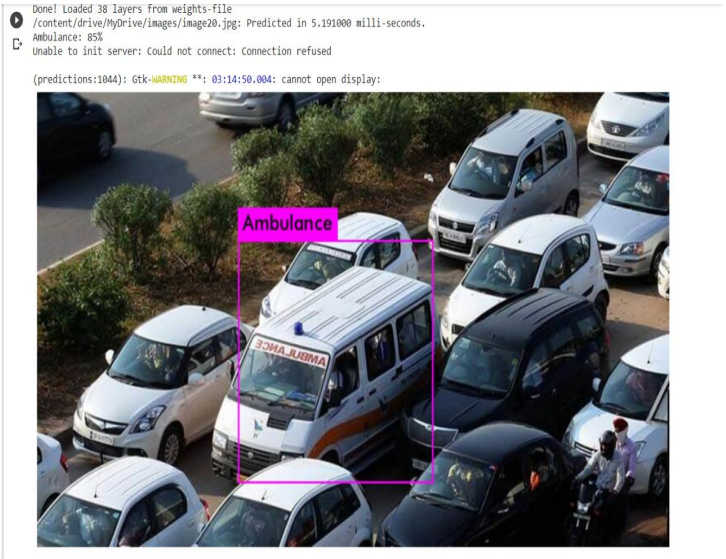


Fig. 36(b) : (Module-2) Output after data augmentation

Fig. 37(a) : Output after Augmentation

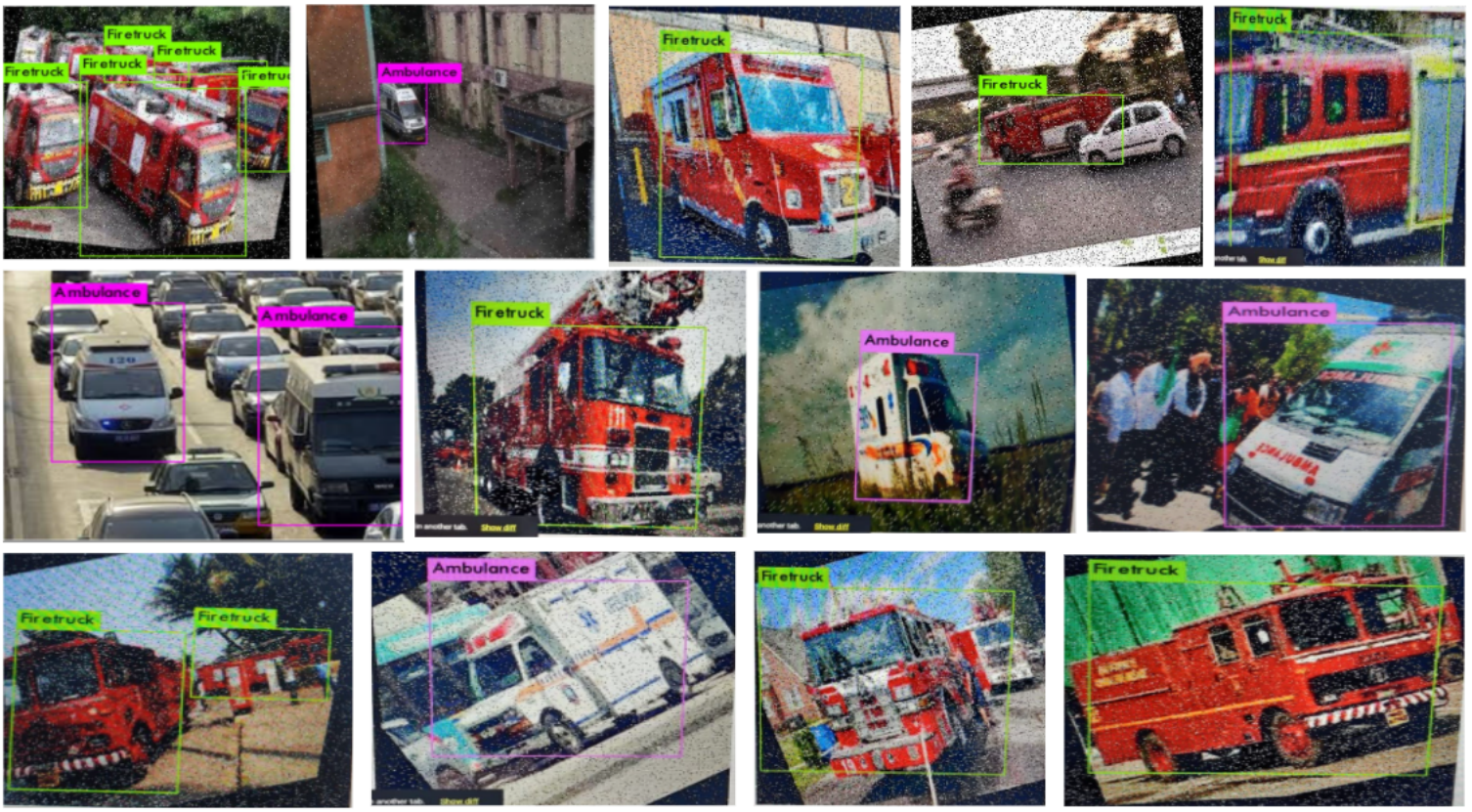


Fig. 37(b) : Output after Augmentation

Module - 3 : Embedded Controller/ Simulator

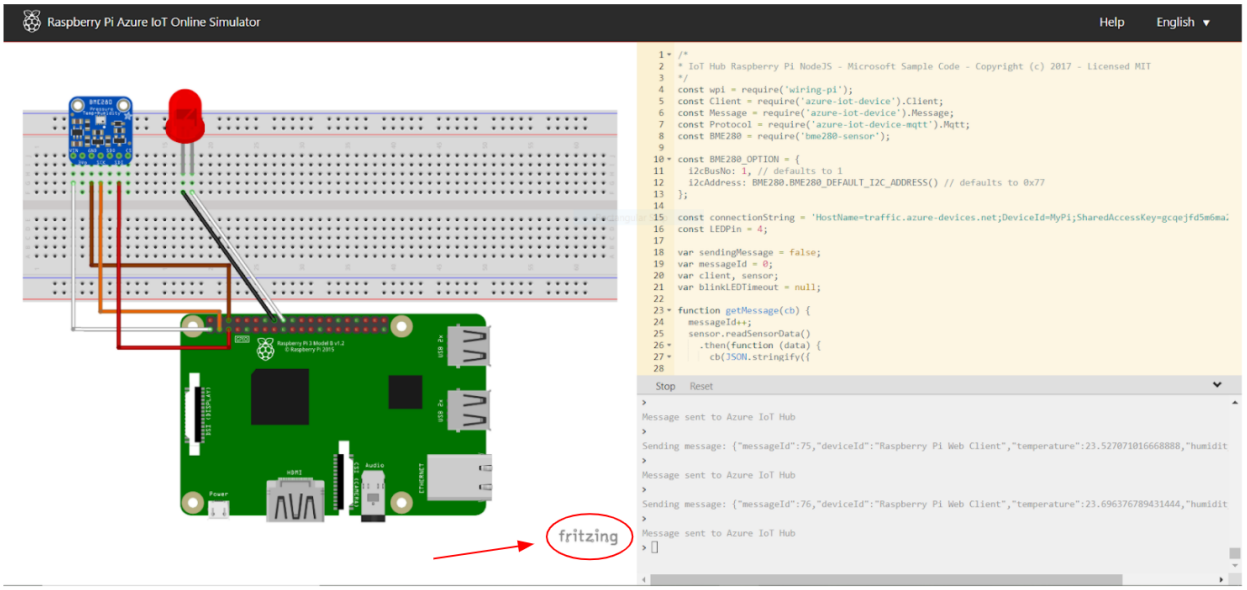


Fig. 38 : Raspberry Pi online simulator (sample)

- Raspberry pi circuit made using fritzing software. The circuit is designed to run a LED blink program on the hardware.

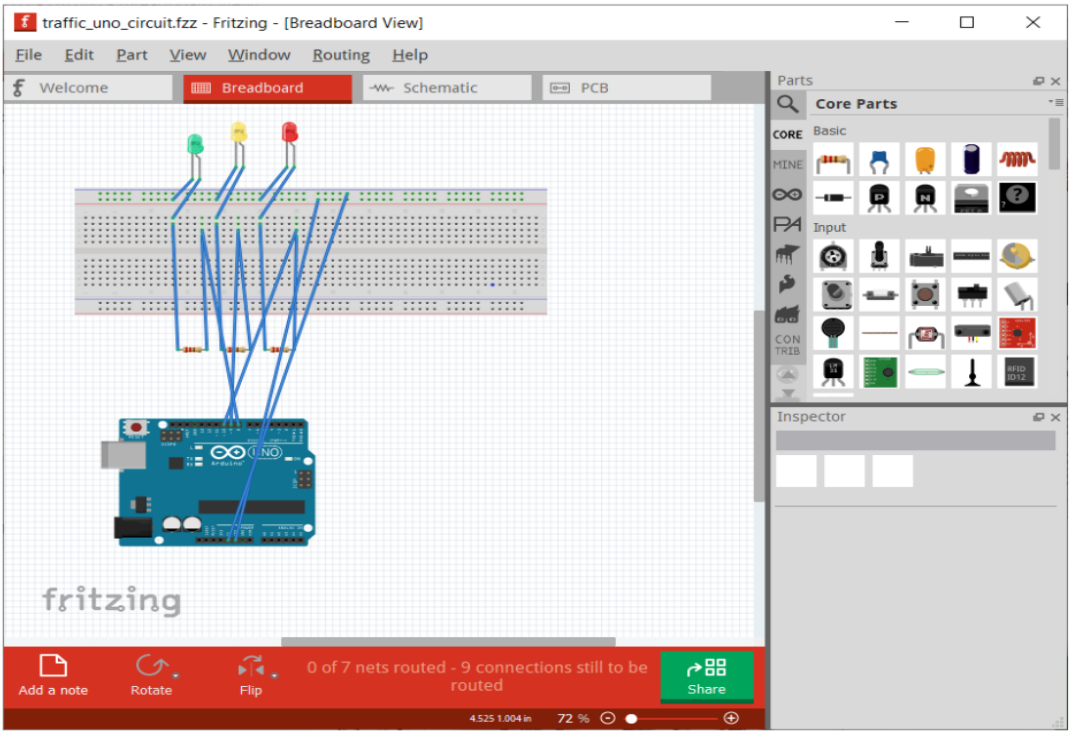


Fig. 39 : Fritzing Arduino circuit (basic)

- Arduino Uno circuit made using fritzing software. The circuit is designed to run a multi-LED blink program to imitate a traffic signal.

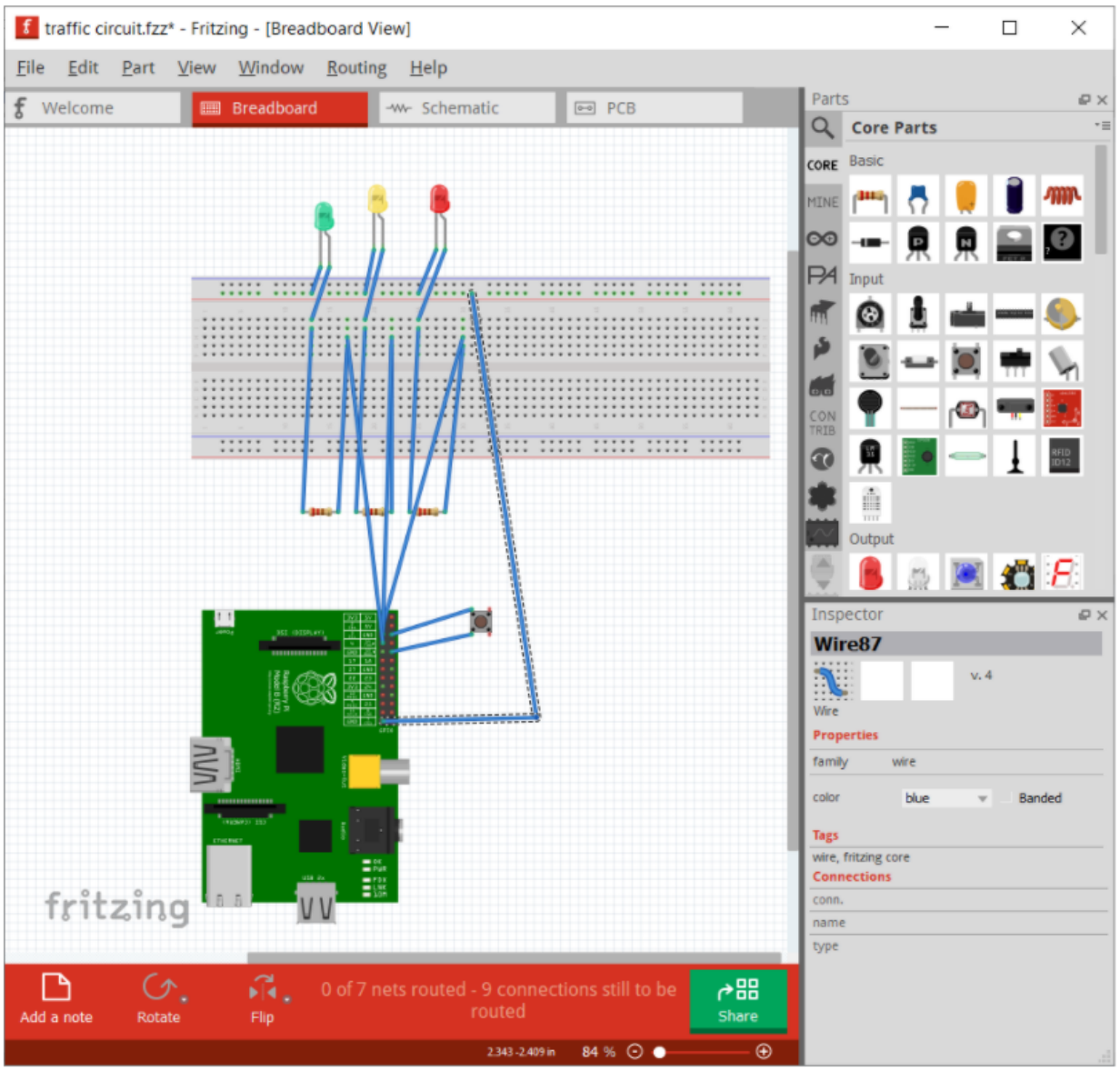


Fig. 40 : Fritzing Raspberry Pi circuit (basic)

- Raspberry pi circuit made using fritzing software. The circuit is designed to run a multi-LED blink program to imitate a traffic signal.

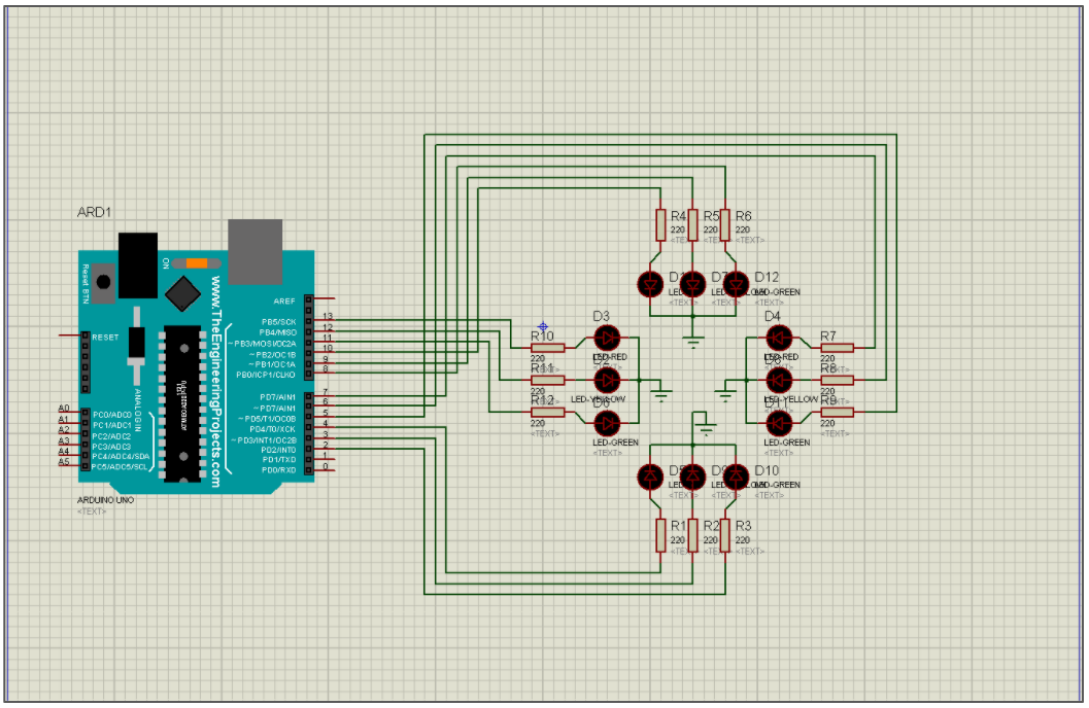


Fig. 41 : Arduino implementation

- Circuit diagram of traditional traffic junction implemented on Proteus 7 simulation software. The circuit diagram shows a 4-way signal and the controller used is Arduino UNO.

code_for_arduino

```
int signal1[] = {23, 25, 27};
int signal2[] = {46, 48, 50};
int signal3[] = {13, 12, 11};
int signal4[] = {10, 9, 8};
int redDelay = 5;
int yellowDelay = 2;
void setup() {
  // Declaring all the LED's as output
  for (int i = 0; i < 3; i++) {
    pinMode(signal1[i], OUTPUT);
    pinMode(signal2[i], OUTPUT);
    pinMode(signal3[i], OUTPUT);
    pinMode(signal4[i], OUTPUT);
  }
}
void loop() {
  // Making Green  LED at signal 1 and red LED's at other signal HIGH
  digitalWrite(signal1[2], HIGH);
  digitalWrite(signal1[0], LOW);
  digitalWrite(signal2[0], HIGH);
  digitalWrite(signal3[0], HIGH);
  digitalWrite(signal4[0], HIGH);
  delay(redDelay);
  // Making Green LED at signal 1 LOW and making yellow LED at signal 1 HIGH for 2 seconds
  digitalWrite(signal1[1], HIGH);
  digitalWrite(signal1[2], LOW);
  delay(yellowDelay);
  digitalWrite(signal1[1], LOW);
  // Making Green  LED at signal 2 and red LED's at other signal HIGH
  digitalWrite(signal1[0], HIGH);
  digitalWrite(signal2[2], HIGH);
  digitalWrite(signal2[0], LOW);
  digitalWrite(signal3[0], HIGH);
  digitalWrite(signal4[0], HIGH);
  delay(redDelay);
  // Making Green LED at signal 2 LOW and making yellow LED at signal 2 HIGH for 2 seconds
  digitalWrite(signal2[1], HIGH);
  digitalWrite(signal2[2], LOW);
  delay(yellowDelay);
  digitalWrite(signal2[1], LOW);
  // Making Green  LED at signal 3 and red LED's at other signal HIGH
  digitalWrite(signal1[0], HIGH);
  digitalWrite(signal2[0], HIGH);
  digitalWrite(signal3[2], HIGH);
  digitalWrite(signal3[0], LOW);
  digitalWrite(signal4[0], HIGH);
  delay(redDelay);
  // Making Green LED at signal 3 LOW and making yellow LED at signal 3 HIGH for 2 seconds
  digitalWrite(signal3[1], HIGH);
  digitalWrite(signal3[2], LOW);
  delay(yellowDelay);
  digitalWrite(signal3[1], LOW);
  // Making Green  LED at signal 4 and red LED's at other signal HIGH
  digitalWrite(signal1[0], HIGH);
  digitalWrite(signal2[0], HIGH);
  digitalWrite(signal3[0], HIGH);
  digitalWrite(signal4[2], HIGH);
  digitalWrite(signal4[0], LOW);
  delay(redDelay);
  // Making Green LED at signal 4 LOW and making yellow LED at signal 4 HIGH for 2 seconds
  digitalWrite(signal4[1], HIGH);
  digitalWrite(signal4[2], LOW);
  delay(yellowDelay);
  digitalWrite(signal4[1], LOW);
}
```

Fig. 42 : Arduino implementation code

- Code snippet for the implementation of the circuit diagram shown in Fig. 33. The code is run on Arduino IDE.

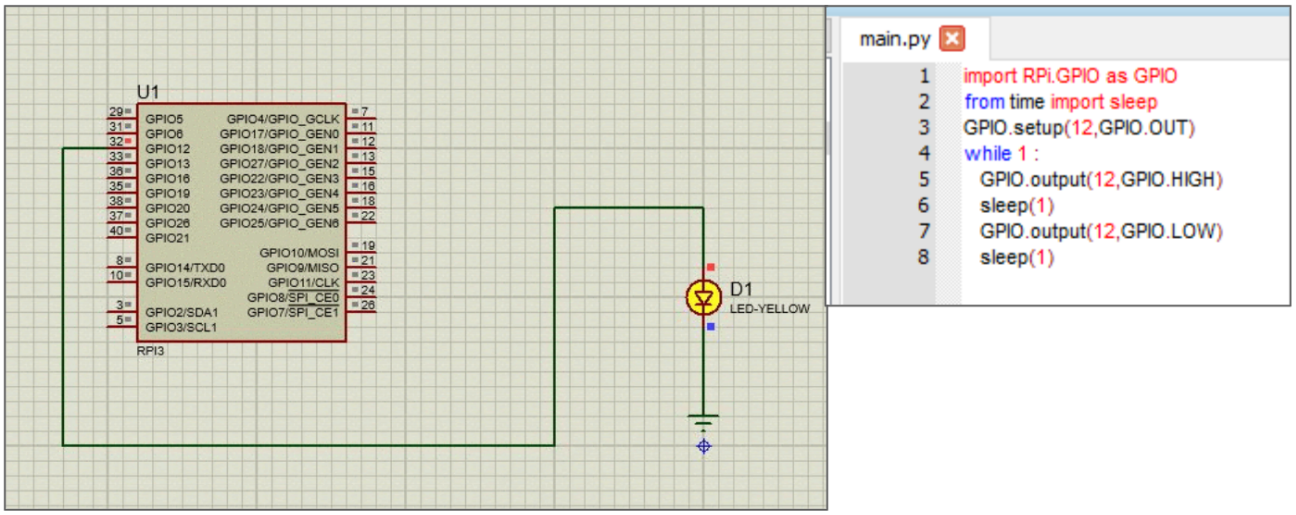


Fig. 43 : Raspberry Pi (LED Blink)

- Circuit diagram for Raspberry blink on the Proteus 7 software.

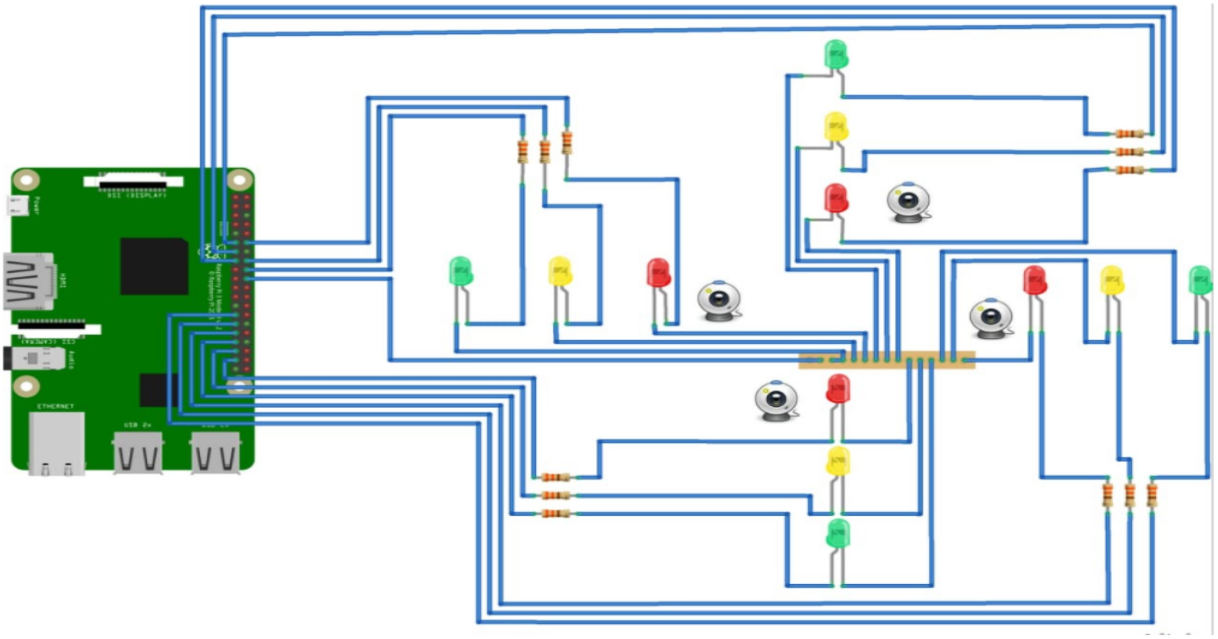


Fig.44 : Raspberry Pi circuit

- After implementation on Arduino, we switched to Raspberry pi because of some pros over Arduino Uno. Circuit diagram made on Fritzing software of a traditional traffic junction.

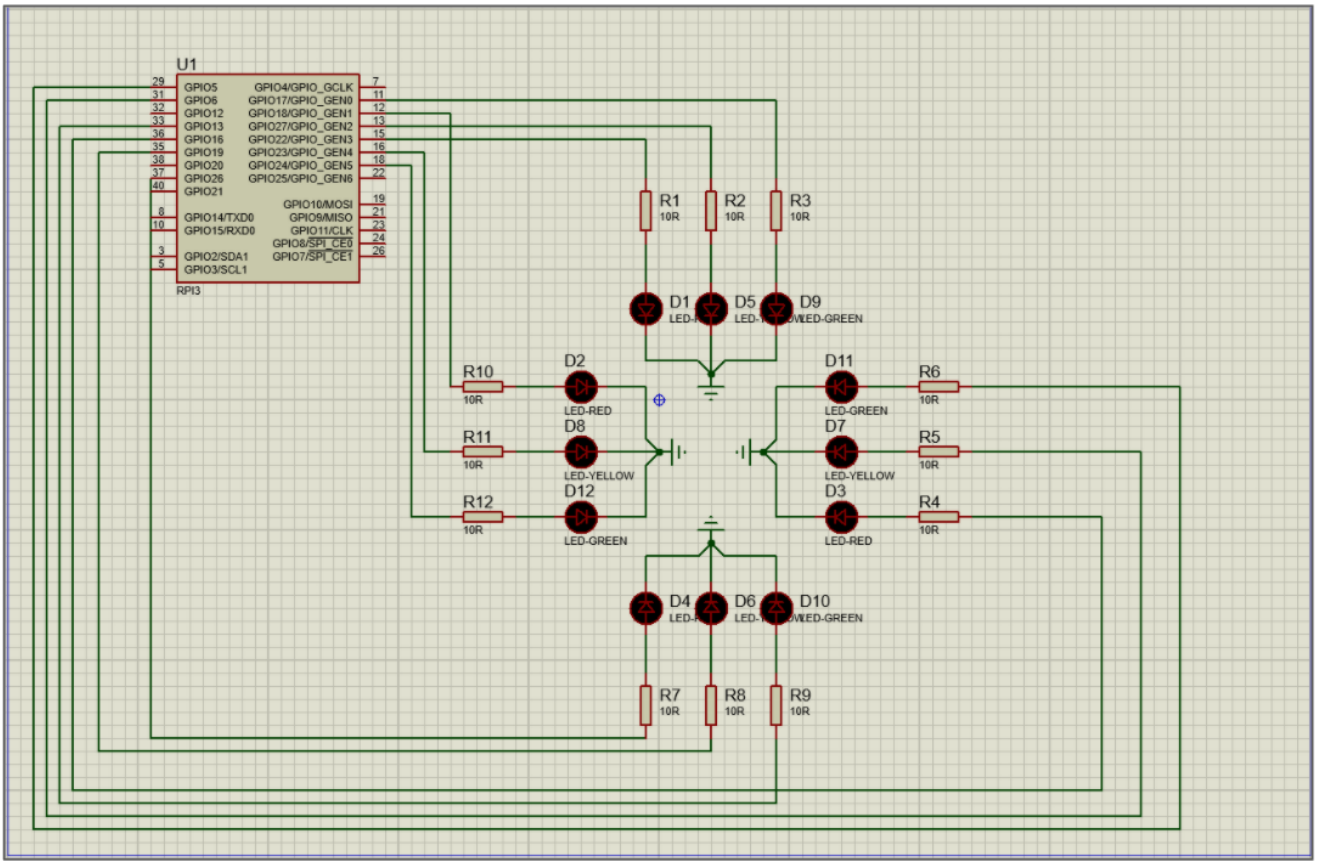


Fig. 45 : Raspberry Pi implementation

- Implementation of the circuit diagram shown in Fig. 36. The implementation is done on the Proteus 7 software.

```
main.py
import RPi.GPIO as GPIO
from time import sleep

GPIO.setwarnings(False)
GPIO.setmode(GPIO.BOARD)

signal1=[12,16,18]
signal2=[15,13,11]
signal3=[36,31,29]
signal4=[37,35,33]
redDelay=2
yellowDelay=1

GPIO.setup(signal1[0],GPIO.OUT, initial=GPIO.LOW)
GPIO.setup(signal1[1],GPIO.OUT, initial=GPIO.LOW)
GPIO.setup(signal1[2],GPIO.OUT, initial=GPIO.LOW)

GPIO.setup(signal2[0],GPIO.OUT, initial=GPIO.LOW)
GPIO.setup(signal2[1],GPIO.OUT, initial=GPIO.LOW)
GPIO.setup(signal2[2],GPIO.OUT, initial=GPIO.LOW)

GPIO.setup(signal3[0],GPIO.OUT, initial=GPIO.LOW)
GPIO.setup(signal3[1],GPIO.OUT, initial=GPIO.LOW)
GPIO.setup(signal3[2],GPIO.OUT, initial=GPIO.LOW)

GPIO.setup(signal4[0],GPIO.OUT, initial=GPIO.LOW)
GPIO.setup(signal4[1],GPIO.OUT, initial=GPIO.LOW)
GPIO.setup(signal4[2],GPIO.OUT, initial=GPIO.LOW)

while 1:
    #Making Green  LED at signal 1 and red LED's at other signal HIGH
    GPIO.output(signal1[2],GPIO.HIGH)
    GPIO.output(signal1[0],GPIO.LOW)
    GPIO.output(signal2[0],GPIO.HIGH)
    GPIO.output(signal3[0],GPIO.HIGH)
    GPIO.output(signal4[0],GPIO.HIGH)
    sleep(redDelay)
    #Making Green LED at signal 1 LOW and making yellow LED at signal 1 HIGH for 2 seconds
    GPIO.output(signal1[1],GPIO.HIGH)
    GPIO.output(signal1[2],GPIO.LOW)
    sleep(yellowDelay)
    GPIO.output(signal1[1],GPIO.LOW)
    #Making Green  LED at signal 2 and red LED's at other signal HIGH
    GPIO.output(signal1[0],GPIO.HIGH)
    GPIO.output(signal2[2],GPIO.HIGH)
    GPIO.output(signal2[0],GPIO.LOW)
    GPIO.output(signal3[0],GPIO.HIGH)
    GPIO.output(signal4[0],GPIO.HIGH)
    sleep(redDelay)
    #Making Green LED at signal 2 LOW and making yellow LED at signal 2 HIGH for 2 seconds
    GPIO.output(signal2[1],GPIO.HIGH)
    GPIO.output(signal2[2],GPIO.LOW)
    sleep(yellowDelay)
    GPIO.output(signal2[1],GPIO.LOW)
    #Making Green  LED at signal 3 and red LED's at other signal HIGH
    GPIO.output(signal1[0],GPIO.HIGH)
    GPIO.output(signal2[0],GPIO.HIGH)
    GPIO.output(signal3[2],GPIO.HIGH)
    GPIO.output(signal3[0],GPIO.LOW)
    GPIO.output(signal4[0],GPIO.HIGH)
    sleep(redDelay)
    #Making Green LED at signal 3 LOW and making yellow LED at signal 3 HIGH for 2 seconds
    GPIO.output(signal3[1],GPIO.HIGH)
    GPIO.output(signal3[2],GPIO.LOW)
    sleep(yellowDelay)
    GPIO.output(signal3[1], GPIO.LOW)
    #Making Green  LED at signal 4 and red LED's at other signal HIGH
    GPIO.output(signal1[0],GPIO.HIGH)
    GPIO.output(signal2[0],GPIO.HIGH)
    GPIO.output(signal3[0],GPIO.HIGH)
    GPIO.output(signal4[2],GPIO.HIGH)
    GPIO.output(signal4[0],GPIO.LOW)
    sleep(redDelay)
    #Making Green LED at signal 4 LOW and making yellow LED at signal 4 HIGH for 2 seconds
    GPIO.output(signal4[1],GPIO.HIGH)
    GPIO.output(signal4[2],GPIO.LOW)
    sleep(yellowDelay)
    GPIO.output(signal4[1],GPIO.LOW)
```

Fig. 46 : Raspberry Pi implementation code

- Code snippet for the implementation of the circuit diagram shown in Fig. 36. The code is in python and runs on pycharm IDE

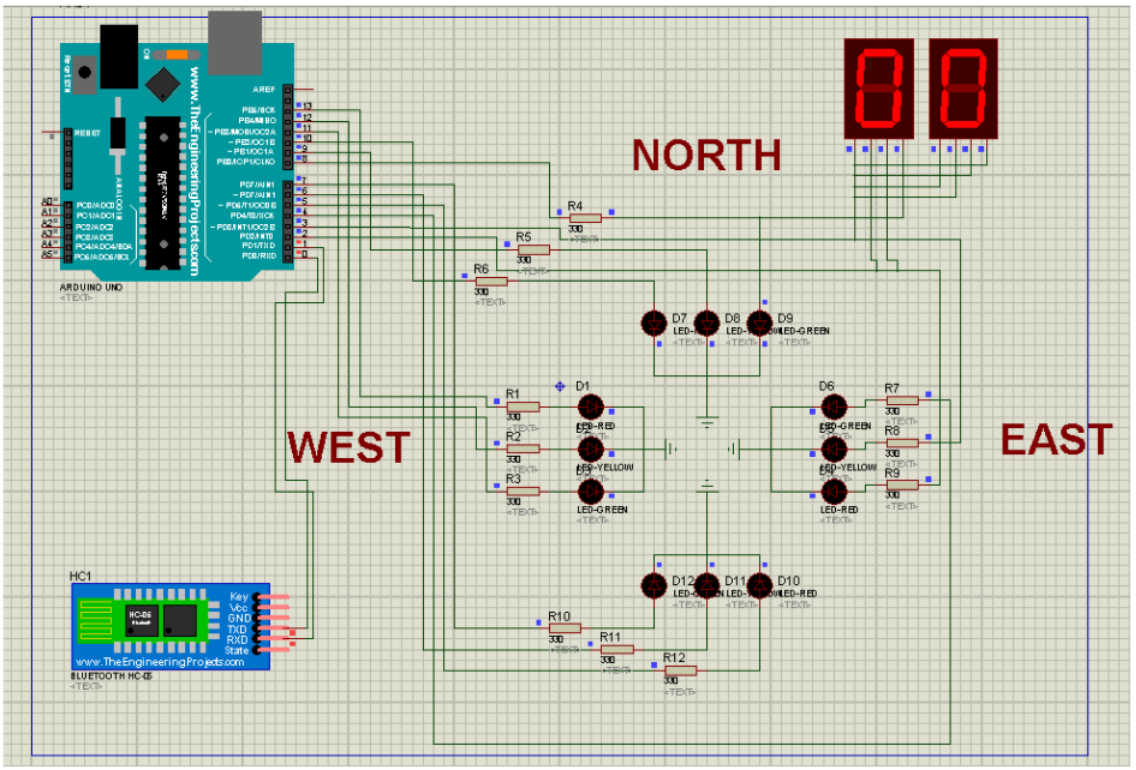
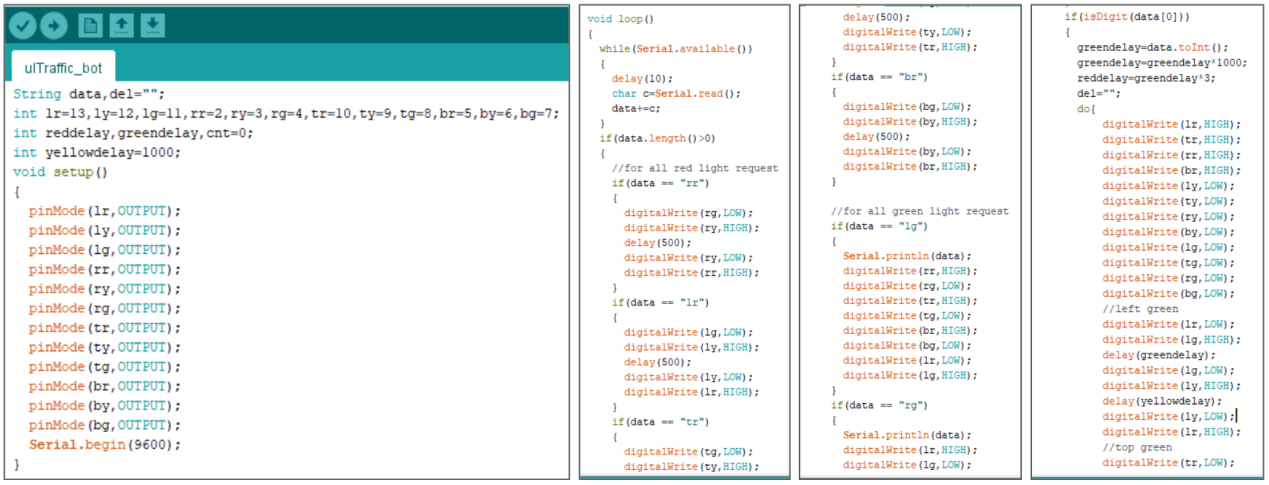


Fig. 47 : Simulator implementation

- Change in the circuit diagram, 4-way lanes are named as North, West, East, South. The timer is added to display the remaining time.



```
ulTraffic_bot

String data,del="";
int lr=13,ly=12,lg=11,rr=2,ry=3,rg=4,tr=10,ty=9,tg=8,br=5,by=6,bg=7;
int reddelay,greendelay,cnt=0;
int yellowdelay=1000;
void setup()
{
  pinMode(lr,OUTPUT);
  pinMode(ly,OUTPUT);
  pinMode(lg,OUTPUT);
  pinMode(rr,OUTPUT);
  pinMode(ry,OUTPUT);
  pinMode(rg,OUTPUT);
  pinMode(tr,OUTPUT);
  pinMode(ty,OUTPUT);
  pinMode(tg,OUTPUT);
  pinMode(br,OUTPUT);
  pinMode(by,OUTPUT);
  pinMode(bg,OUTPUT);
  Serial.begin(9600);
}
```

```
void loop()
{
  while(Serial.available())
  {
    delay(10);
    char c=Serial.read();
    data+=c;
  }
  if(data.length()>0)
  {
    //for all red light request
    if(data == "rr")
    {
      digitalWrite(rg,LOW);
      digitalWrite(ry,HIGH);
      delay(500);
      digitalWrite(ry,LOW);
      digitalWrite(rr,HIGH);
    }
    if(data == "lr")
    {
      digitalWrite(lg,LOW);
      digitalWrite(ly,HIGH);
      delay(500);
      digitalWrite(ly,LOW);
      digitalWrite(lr,HIGH);
    }
    if(data == "tr")
    {
      digitalWrite(tg,LOW);
      digitalWrite(ty,HIGH);
```

```
      delay(500);
      digitalWrite(ty,LOW);
      digitalWrite(tr,HIGH);
    }
    if(data == "br")
    {
      digitalWrite(bg,LOW);
      digitalWrite(by,HIGH);
      delay(500);
      digitalWrite(by,LOW);
      digitalWrite(br,HIGH);
    }

    //for all green light request
    if(data == "lg")
    {
      Serial.println(data);
      digitalWrite(rr,HIGH);
      digitalWrite(rg,LOW);
      digitalWrite(tr,HIGH);
      digitalWrite(tg,LOW);
      digitalWrite(br,HIGH);
      digitalWrite(bg,LOW);
      digitalWrite(lr,LOW);
      digitalWrite(lg,HIGH);
    }
    if(data == "rg")
    {
      Serial.println(data);
      digitalWrite(lr,HIGH);
      digitalWrite(lg,LOW);
```

```
    if(isDigit(data[0]))
    {
      greendelay=data.toInt();
      greendelay=greendelay*1000;
      reddelay=greendelay*3;
      del="";
      do{
          digitalWrite(lr,HIGH);
          digitalWrite(tr,HIGH);
          digitalWrite(rr,HIGH);
          digitalWrite(br,HIGH);
          digitalWrite(ly,LOW);
          digitalWrite(ty,LOW);
          digitalWrite(ry,LOW);
          digitalWrite(by,LOW);
          digitalWrite(lg,LOW);
          digitalWrite(tg,LOW);
          digitalWrite(rg,LOW);
          digitalWrite(bg,LOW);
          //left green
          digitalWrite(lr,LOW);
          digitalWrite(lg,HIGH);
          delay(greendelay);
          digitalWrite(lg,LOW);
          digitalWrite(ly,HIGH);
          delay(yellowdelay);
          digitalWrite(ly,LOW);
          digitalWrite(lr,HIGH);
          //top green
          digitalWrite(tr,LOW);
```

Fig. 48 : Simulator implementation code snippet

- Code snippet for the implementation of the circuit diagram shown in Fig. 39. The code is run on pycharm IDE.

- Implementation of Module-1 on Raspberry pi

Fig. 49(a) : Input Image

Fig. 49(a) : Output Image

```
pi@raspberrypi:~/Desktop/yolo-object-detection $ python3 vehicle_density.py -i images/Anuvrat-dwar2.jpg -y yolo-coco
[INFO] loading YOLO from disk...
[INFO] YOLO took 6.961342 seconds
Cars: 7
Motorbike: 1
Person: 2
```

Fig. 50 : Output (Count of vehicles etc.)

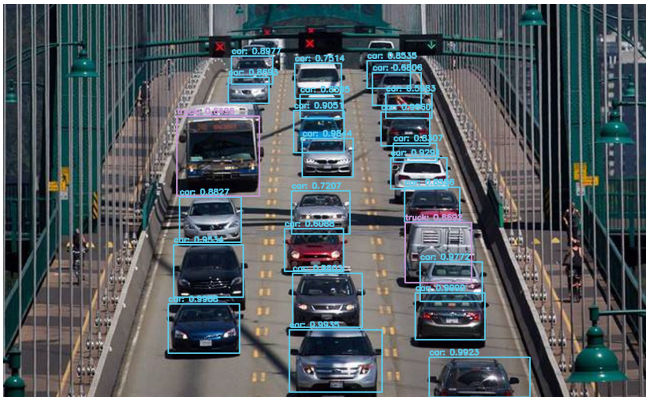


Fig. 51(a) : Input Image

Fig. 51(a) : Output Image

```
pi@raspberrypi:~/Desktop/yolo-object-detection $ python3 vehicle_density.py -i images/traffic-4.jpg -y yolo-coco
[INFO] loading YOLO from disk...
[INFO] YOLO took 6.995734 seconds
Cars: 23
Truck: 2
```

Fig. 52 : Output (Count of vehicles etc.)

Fig. 53(a) : Input Image

Fig. 53(b) : Output Image

```
pi@raspberrypi:~/Desktop/yolo-object-detection $ python3 vehicle_density.py -i images/traffic-5.jpg -y yolo-coco
[INFO] loading YOLO from disk...
[INFO] YOLO took 6.999207 seconds
Cars: 27
Motorbike: 1
Truck: 3
Bus: 6
Person: 5
```

Fig. 54 : Output (Count of vehicles etc.)

- Implementation of Modules on Server.

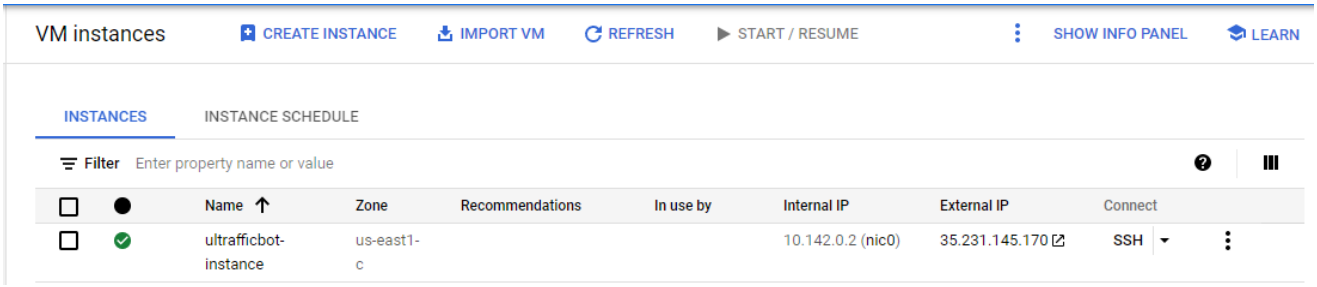


Fig. 55 : GCP console for VM instance.

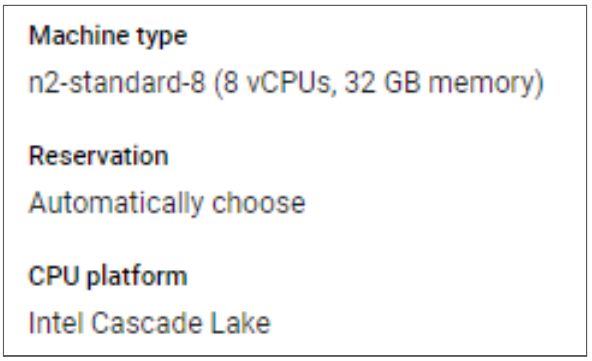


Fig 56 : VM instance configuration.

- Calculation of Green time.

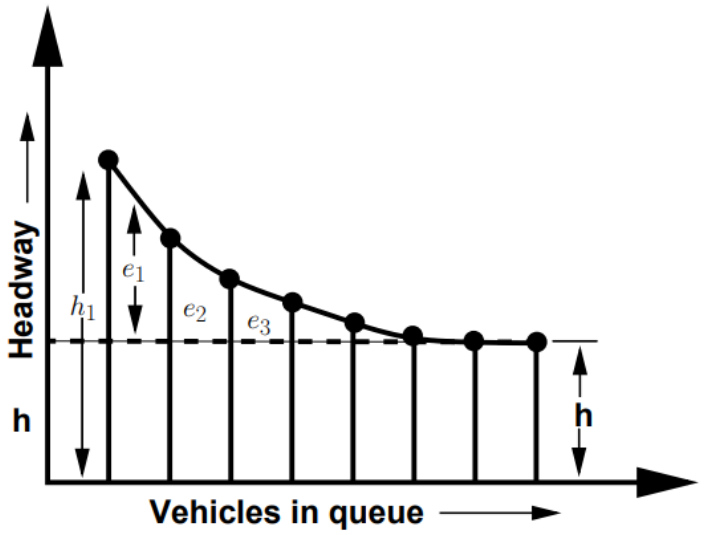


Fig. 57 : Headways departing signal. [15]

- As noted earlier, the headway will be more than **h,** particularly for the first few vehicles. The difference between the actual headway and **h** for the **i**th vehicle and is denoted as shown in Fig. xx.
- These differences for the first few vehicles can be added to get start-up lost time, $\boldsymbol{l_1}$ which is given by:

$$l = \sum_{i=1}^{n} e_i \quad \rightarrow (1)$$

- The green time required to clear **N** vehicles can be found out as:

$$T = l + \frac{h.N}{P} \rightarrow (2)$$

- Where **T** is the time required to clear **N** vehicles through signal, $\boldsymbol{l_1}$ is the start-up lost time, and h is the saturation headway in seconds.
- And **P** is the number of lanes on the road.

Module - 4 : Android Application

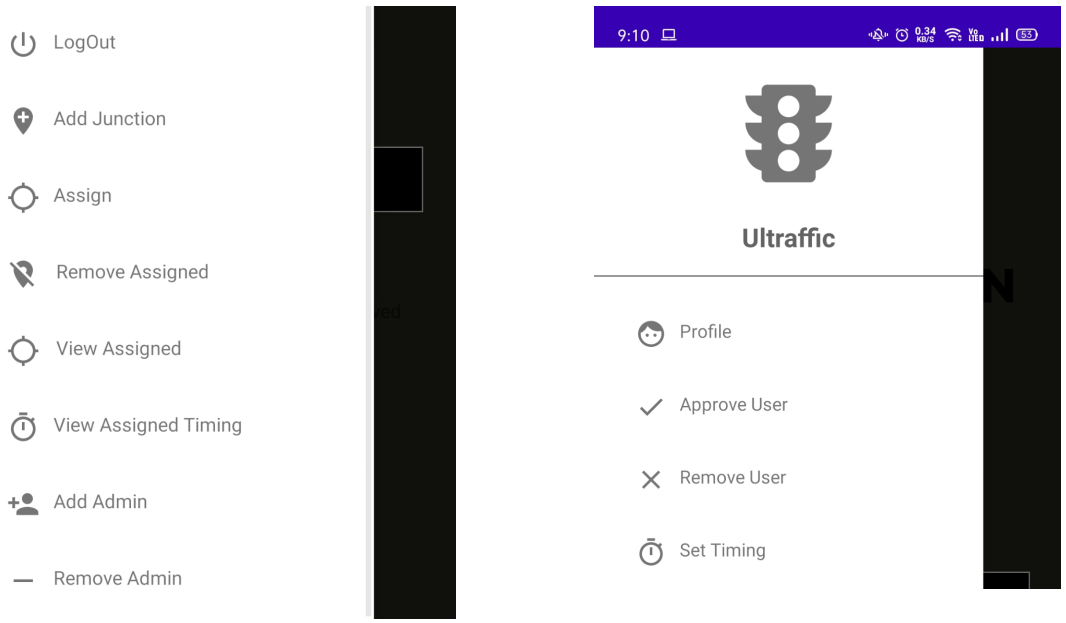


Fig. 58(a) : Admin - collapsable menu bar

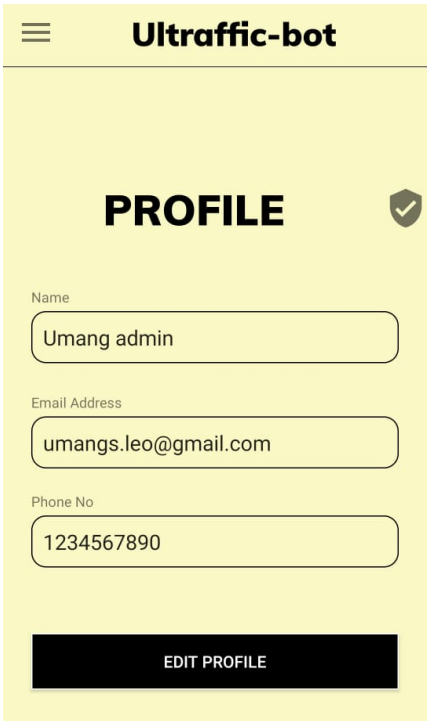


Fig. 58(b) : Admin-Profile

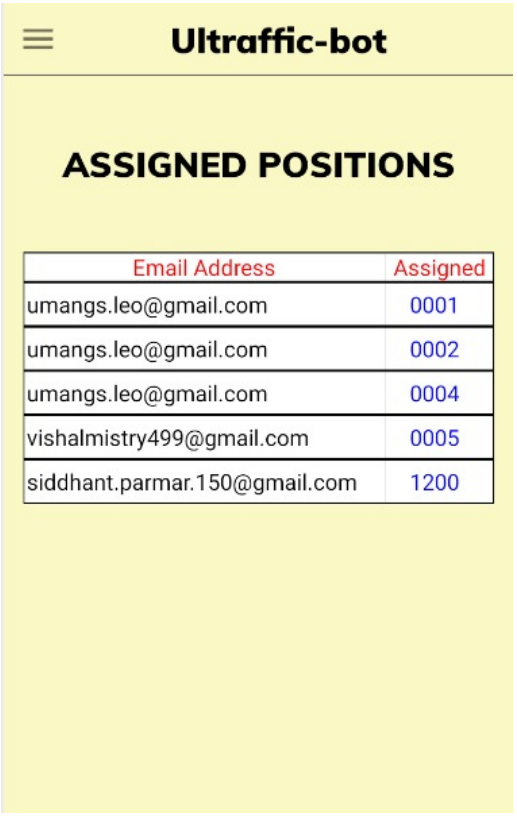


Fig. 59 : Admin-View assigned positions (Home Page)

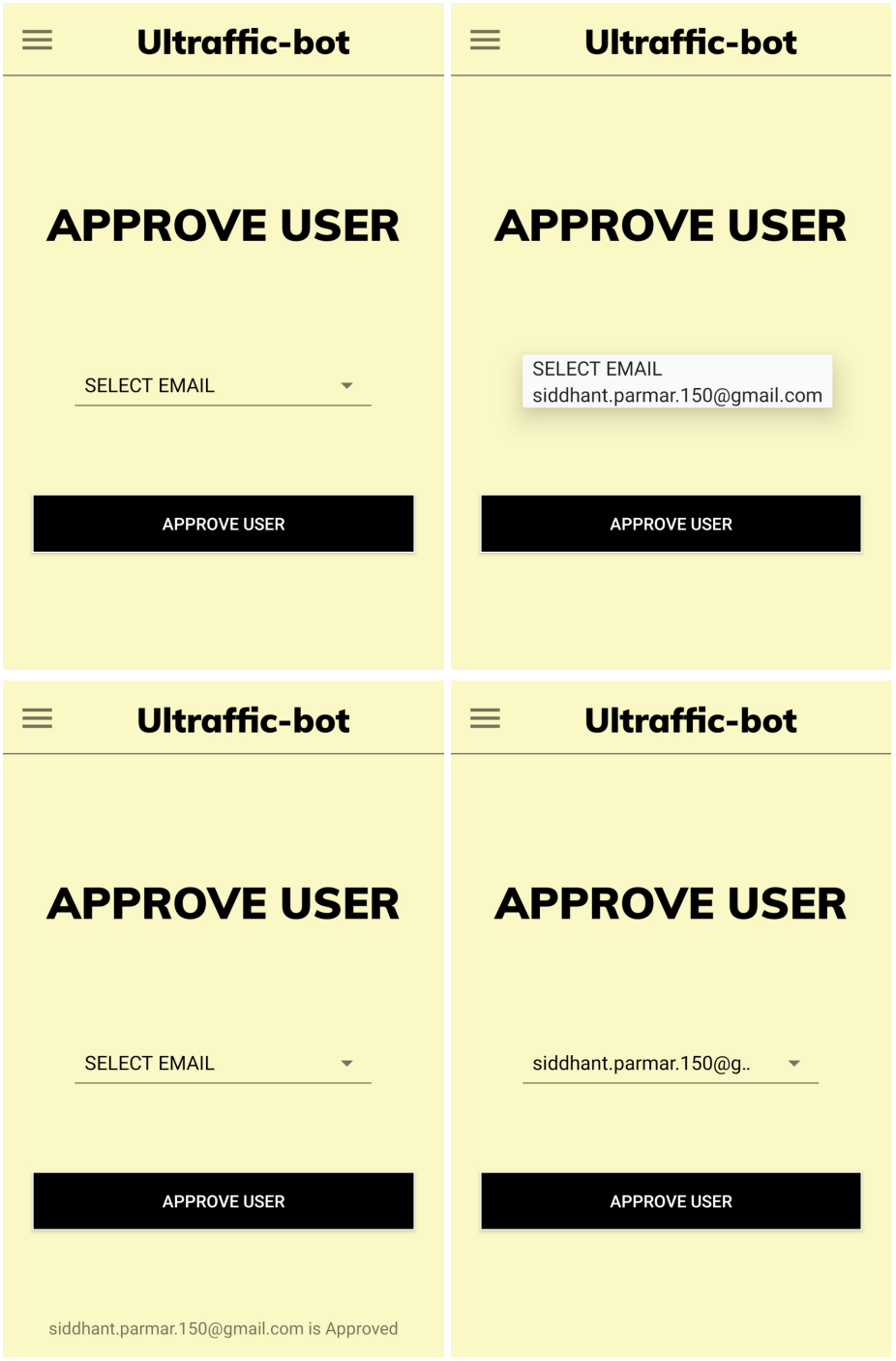


Fig. 60 : Admin-Approve user screen

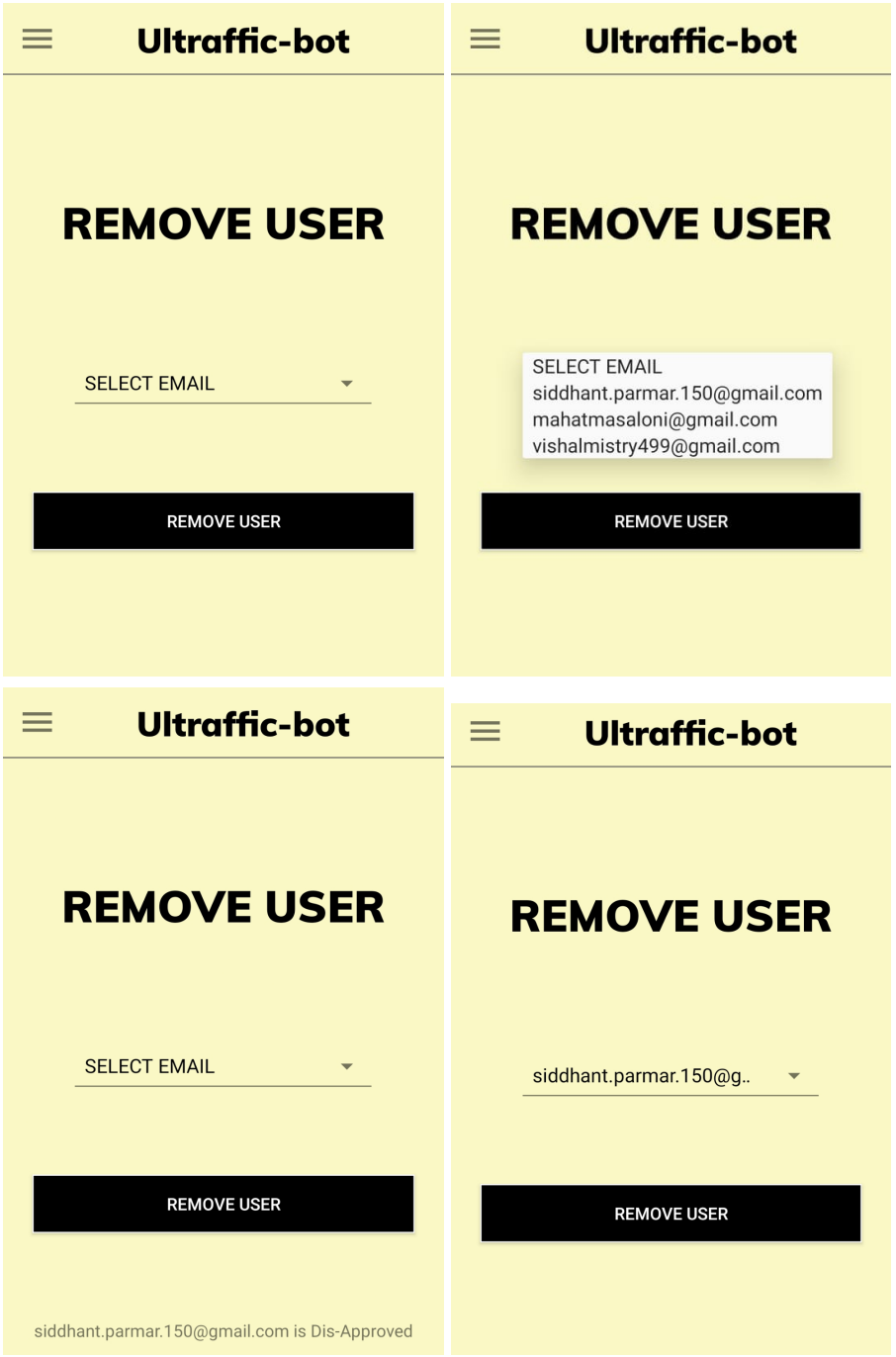


Fig. 61 : Admin-Remove user screen

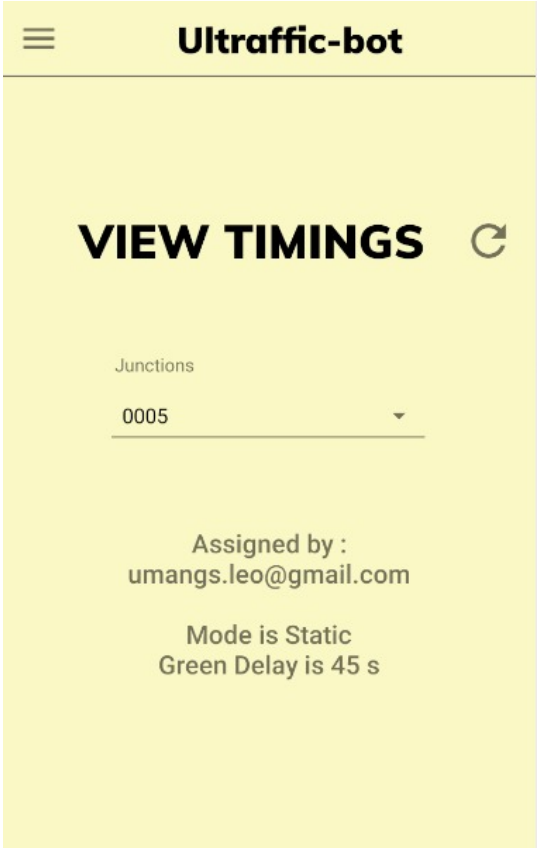


Fig. 62 : Admin-View assigned timings

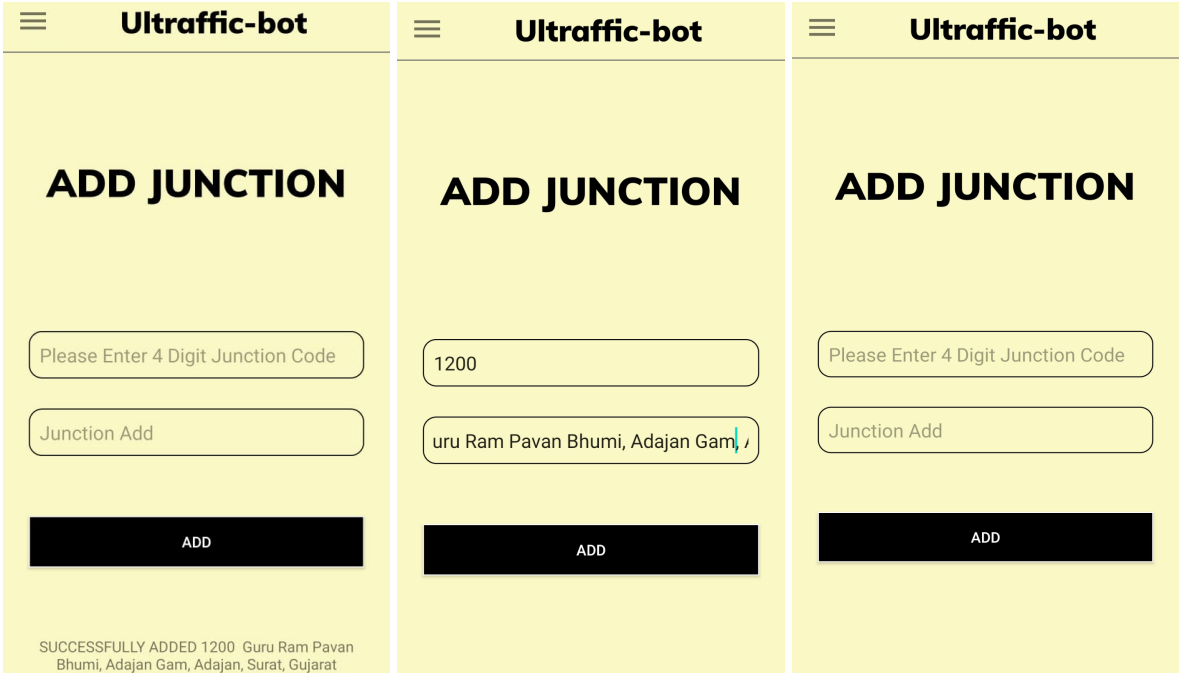


Fig. 63 : Admin-
Add junction screen

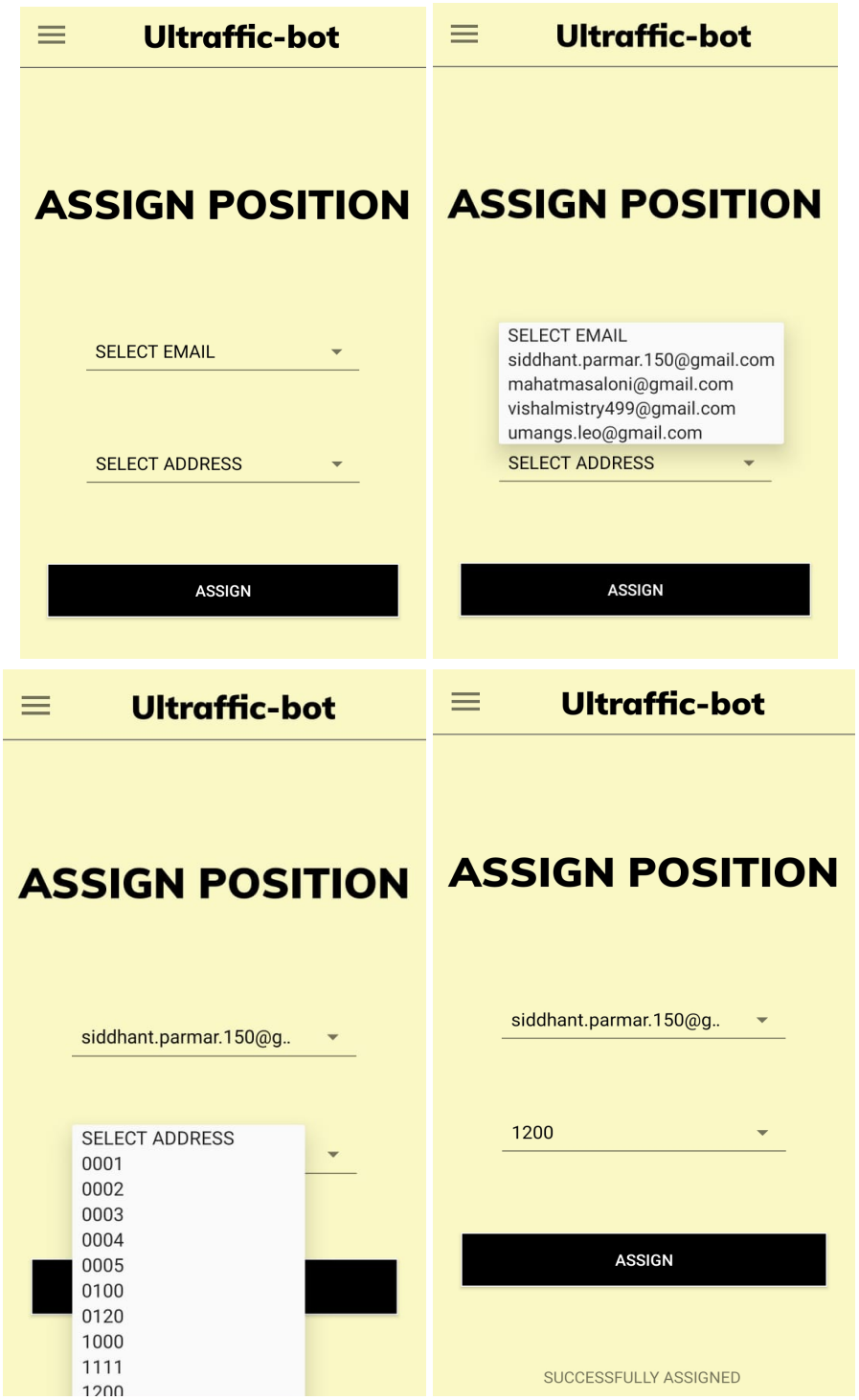


Fig. 64 : Admin- Assign position screen

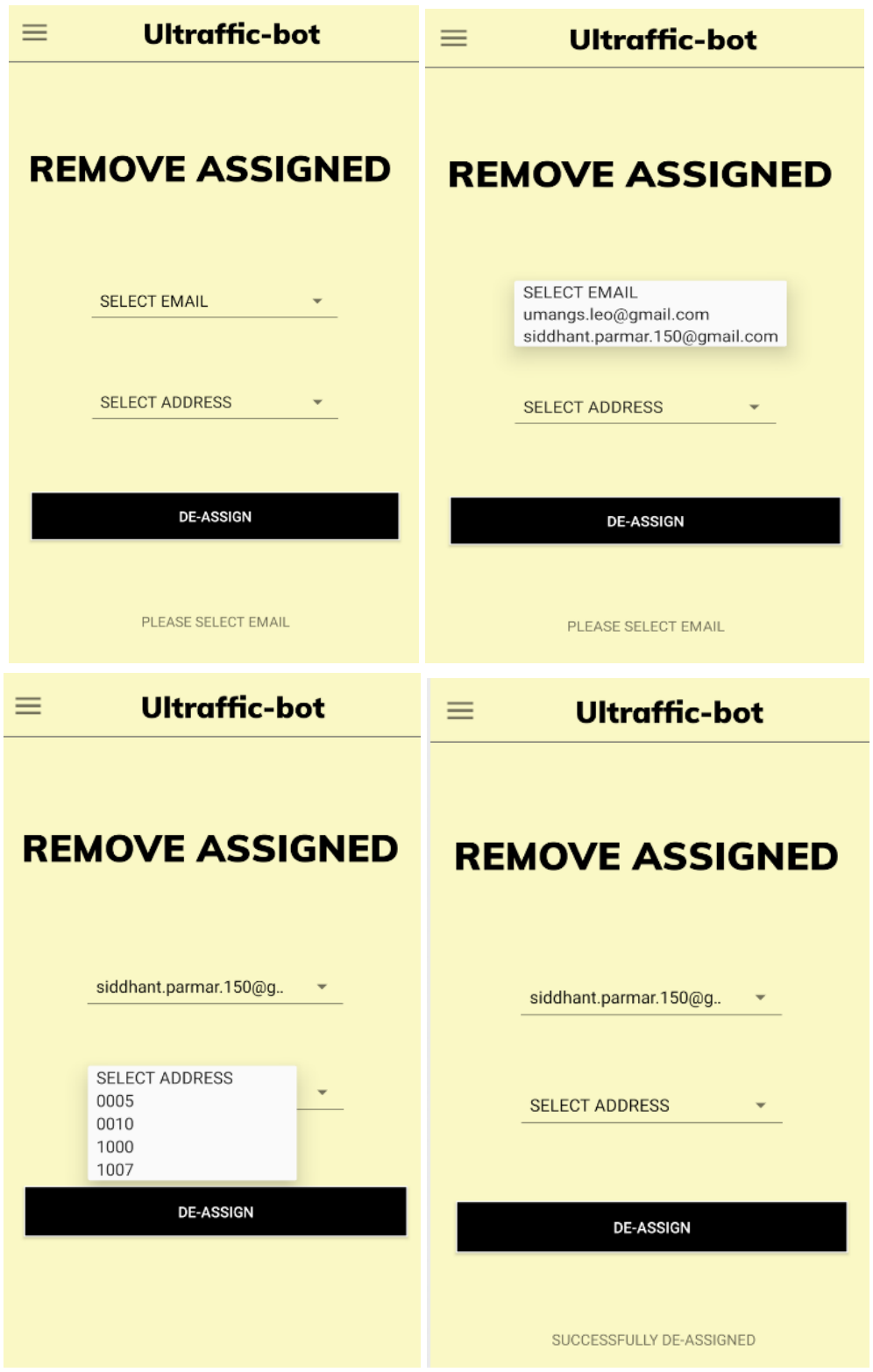


Fig. 65 : Admin-De-Assign position screen

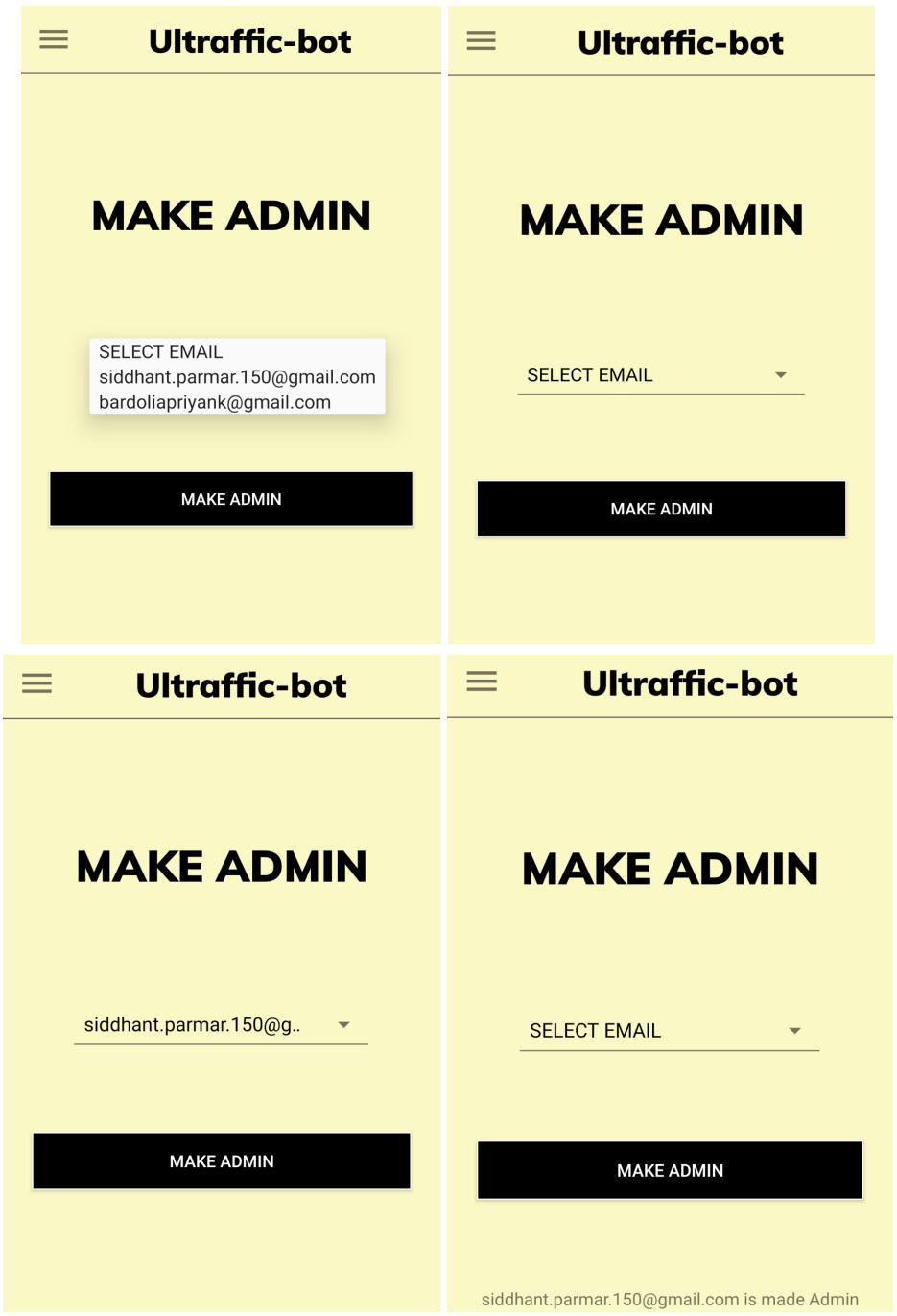


Fig. 66 : Admin-Make Admin

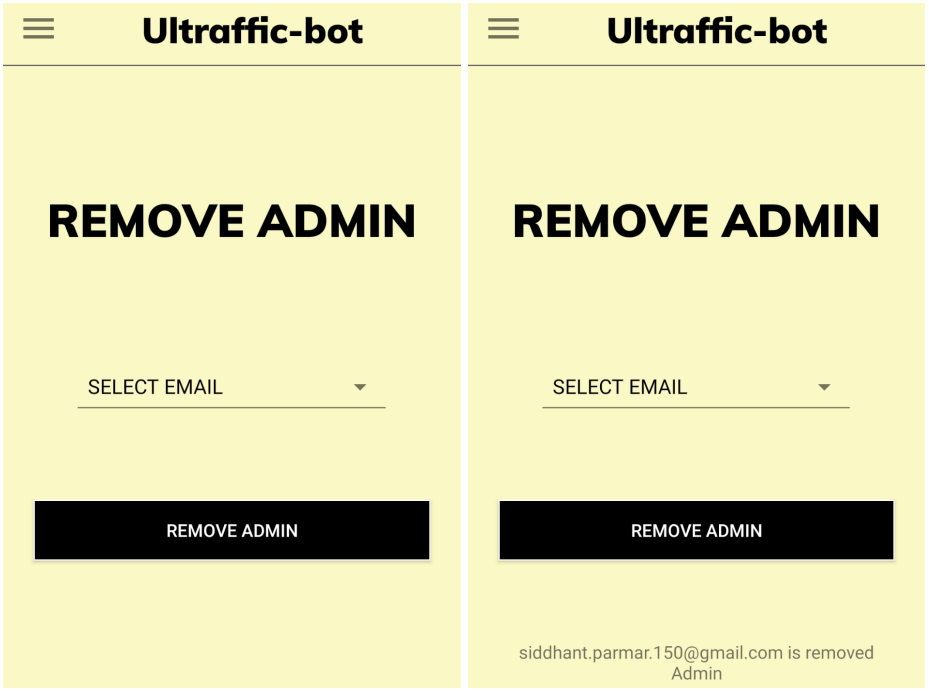


Fig. 67 : Admin-Remove Admin

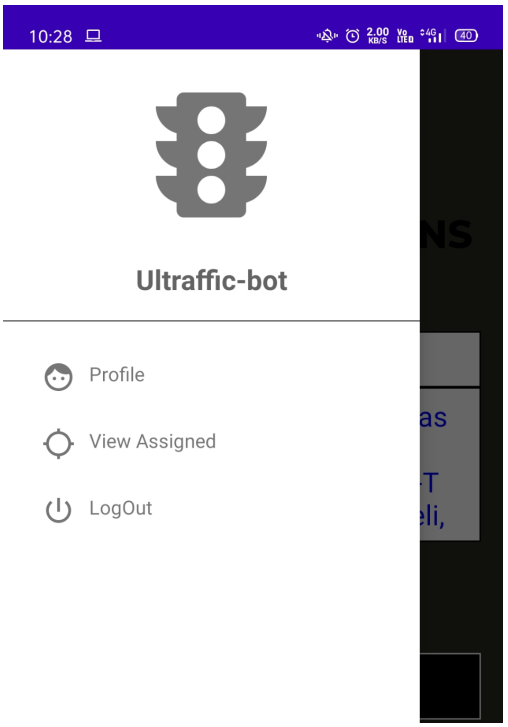


Fig. 68 : User-collapsable menu bar

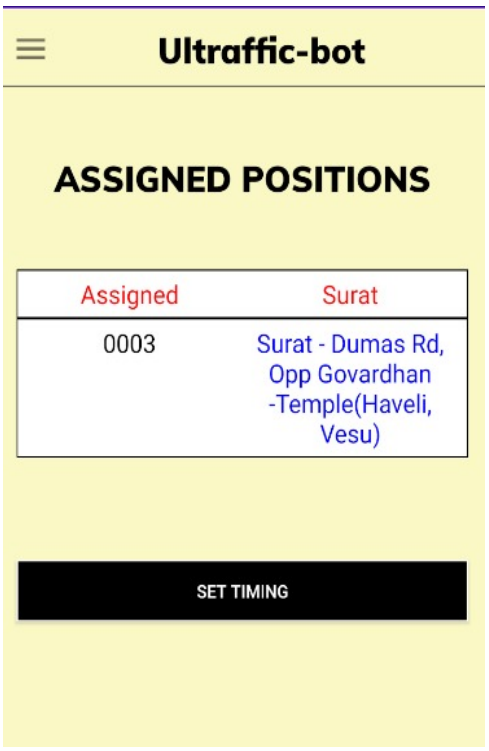


Fig. 69 : User-View Assigned (homepage)

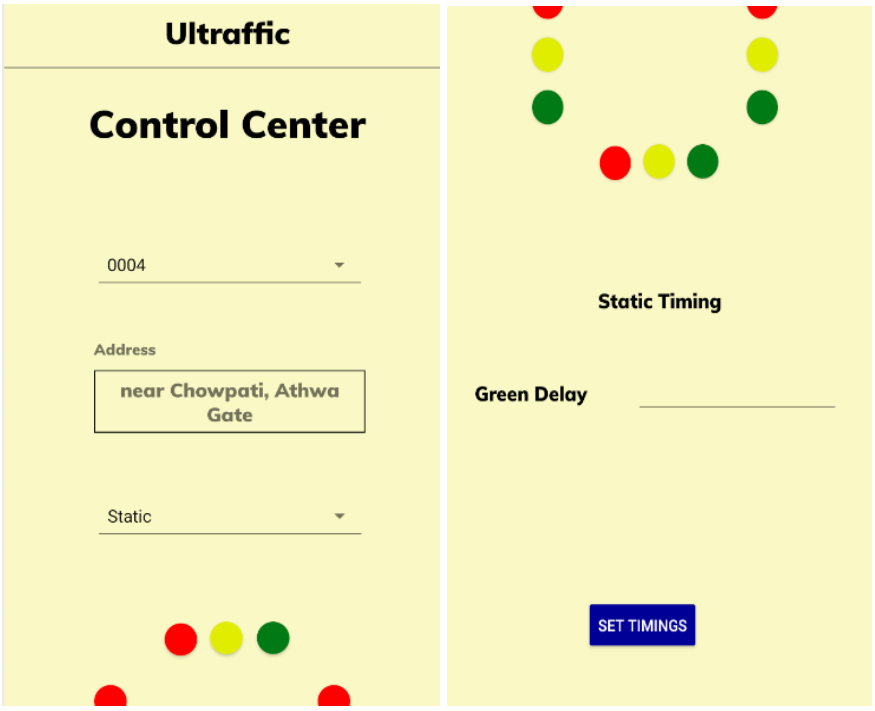


Fig. 70 : User-Set time screen

Module - 4 :

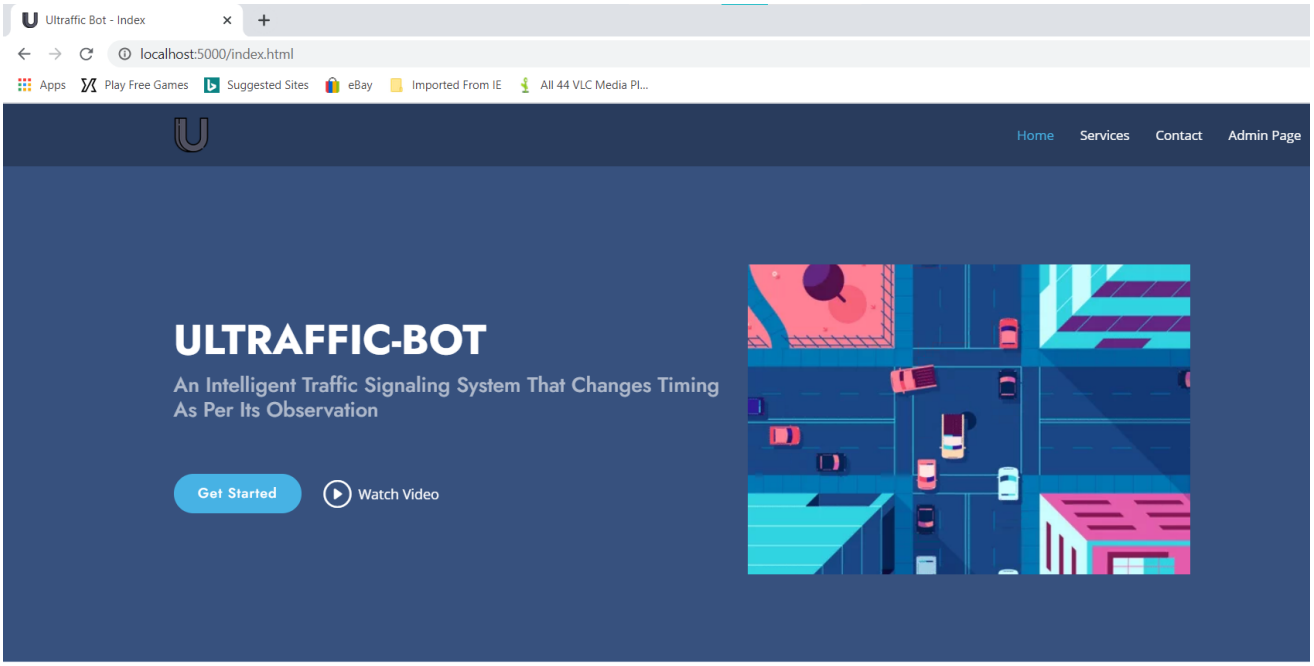


Fig. 71 : Home Page

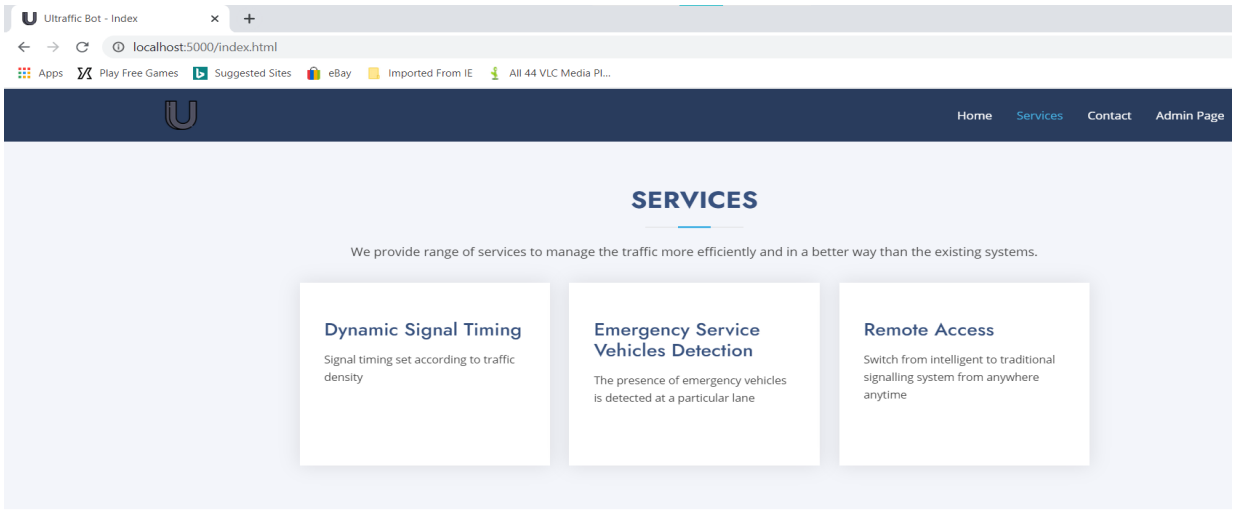


Fig. 72 : Home Page (Services section)

- The above two figures show the snapshot of the Home page of the website.

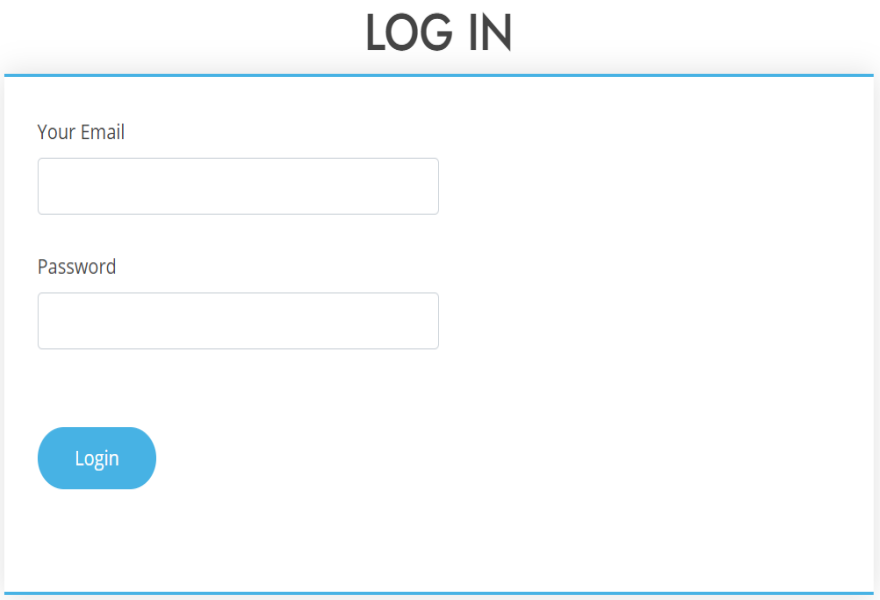


Fig. 73 : Login Page

- Snapshot of the Login page of the website.

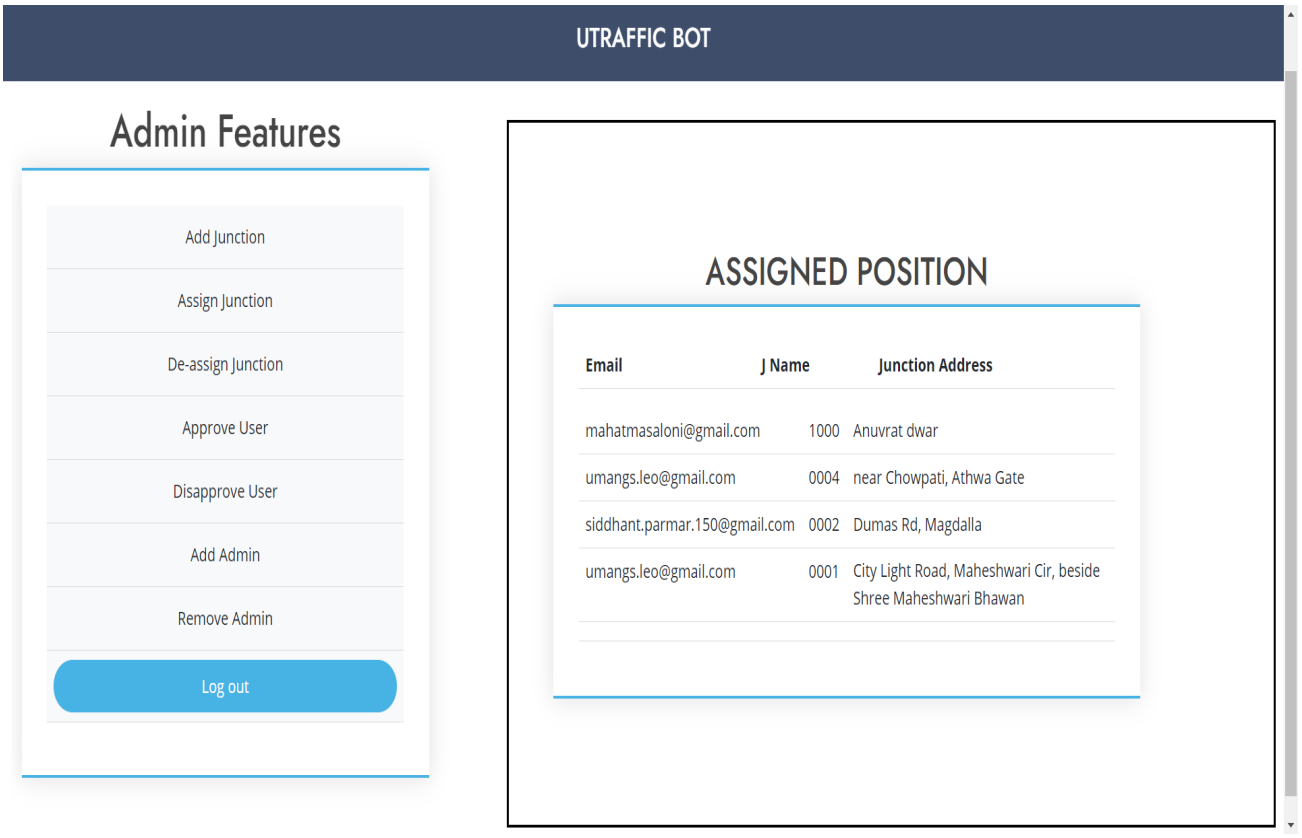


Fig. 74 : Admin Page

- The figure above is a snapshot of the Admin page.
- This page provides all the administrative functionalities in order to manage the users and junctions only to the admin users. It also shows a table of currently assigned junctions to a particular user.

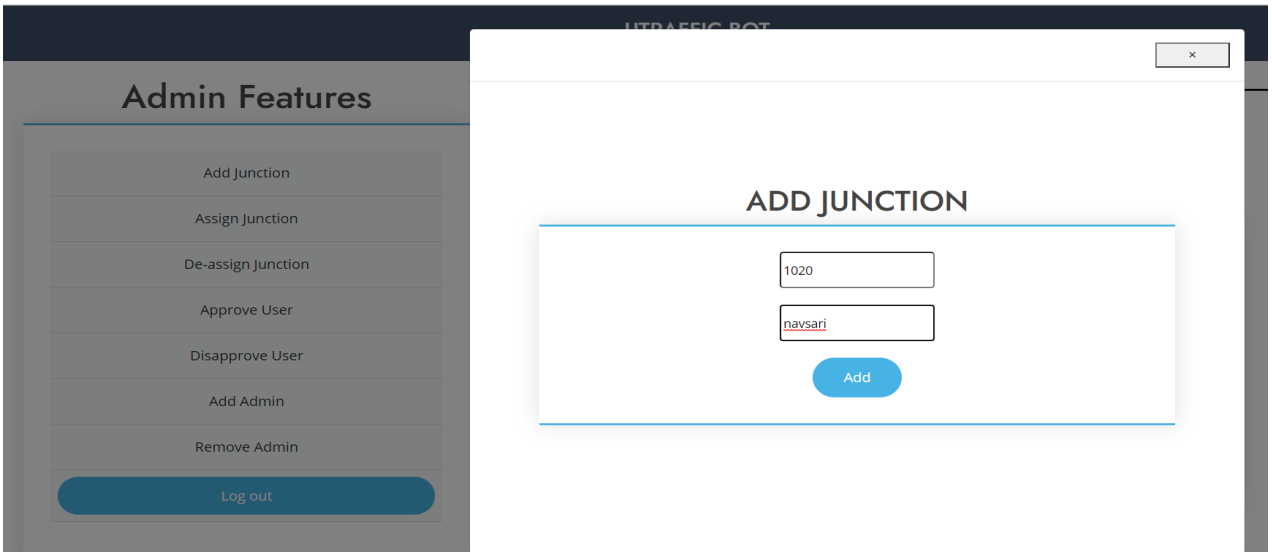


Fig. 75 : Admin Page (Add Junction)

- The Add Junction window lets the admin add a junction to the firestore database.

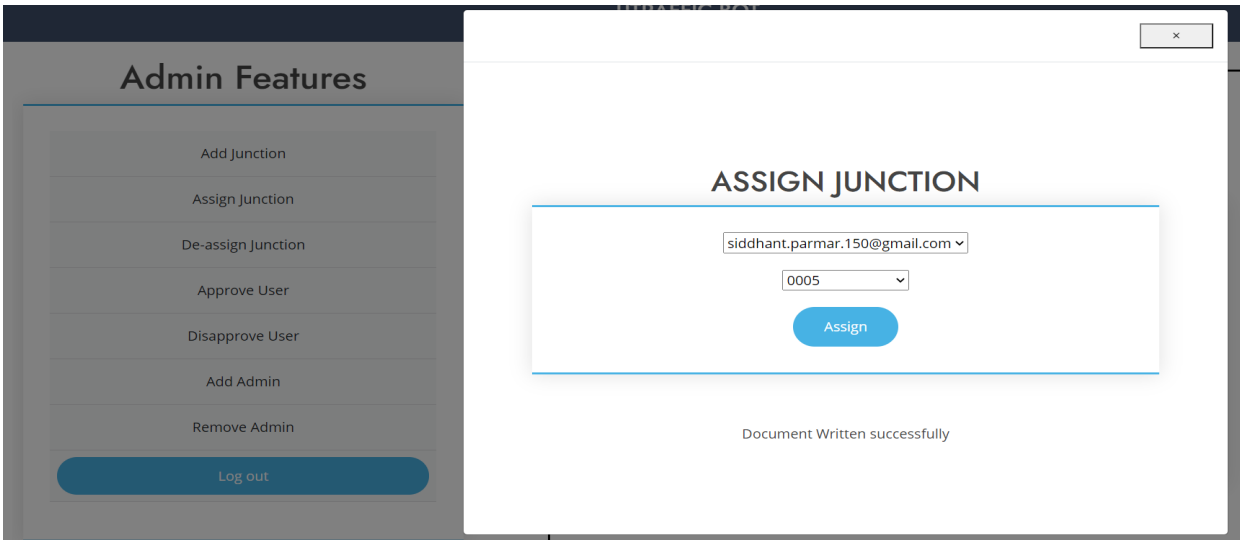


Fig. 76 : Admin Page (Assign Junction)

- The Assign Junction window lets the admin assign a junction to a specific user.

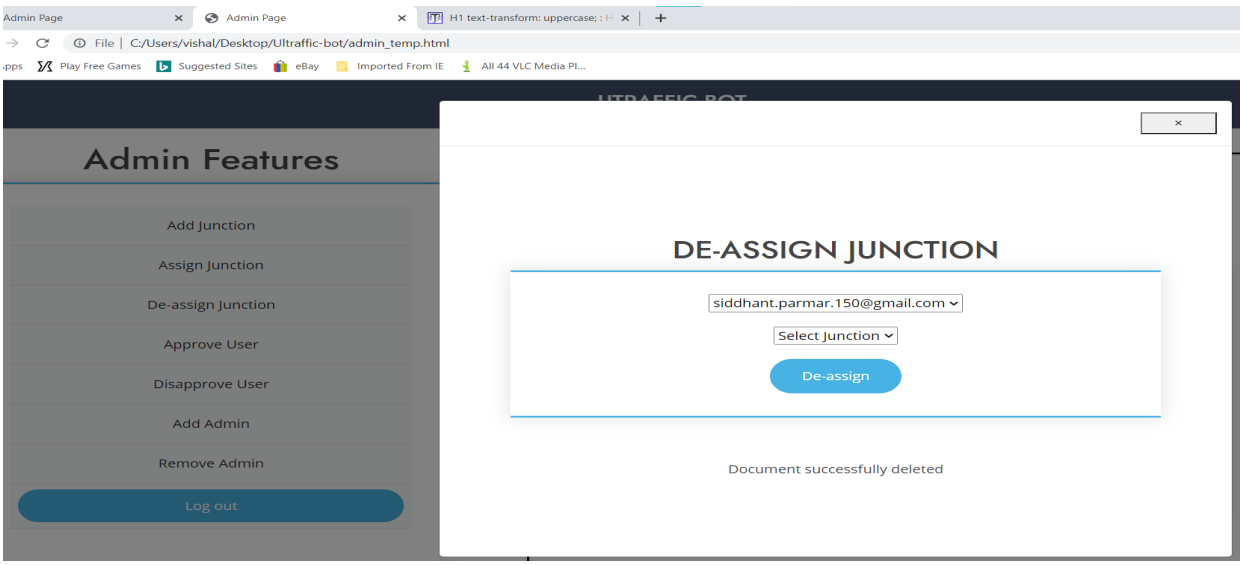


Fig. 77 : Admin Page (De-assign Junction)

- The De-assign Junction window lets the admin de-assign a junction from a specific user.

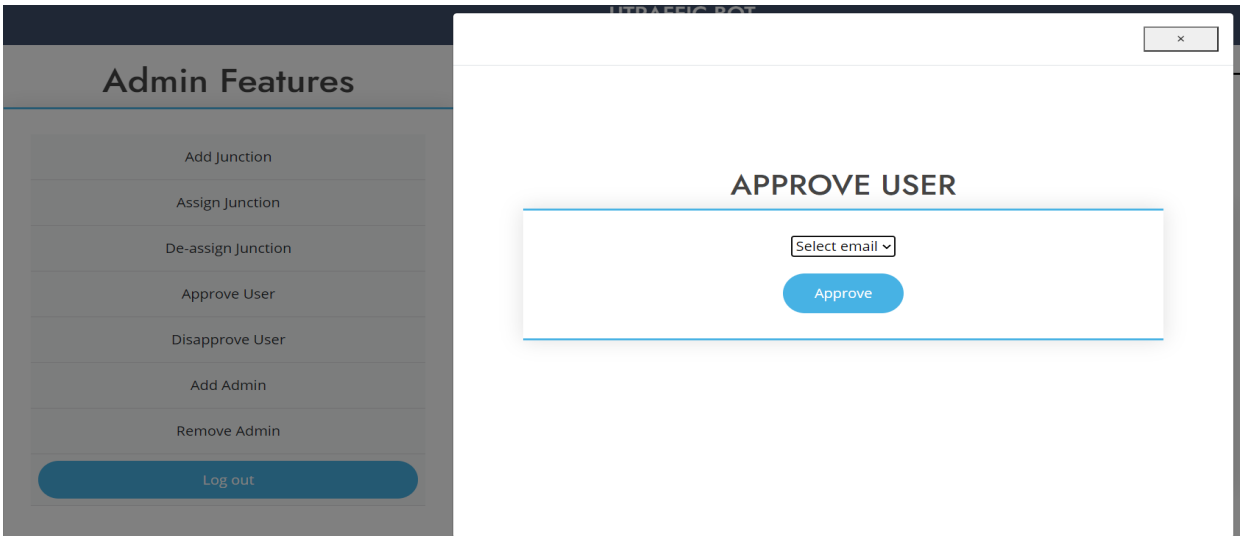


Fig. 78 : Admin Page (Approve User)

- The Approve User window lets the admin approve a user.

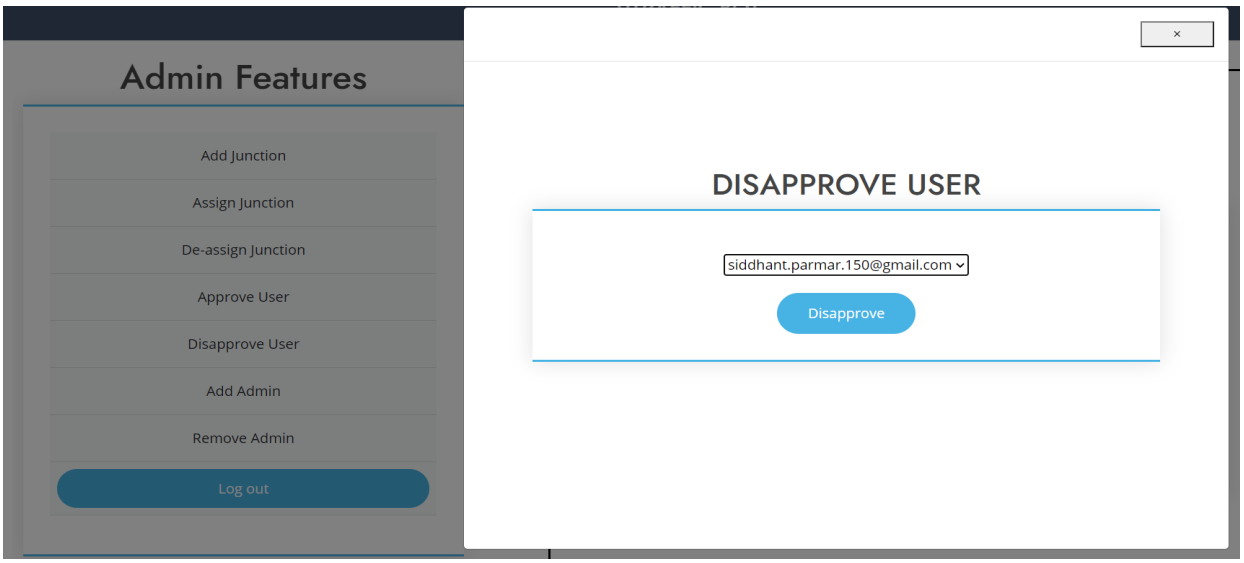


Fig. 79 : Admin Page (Disapprove User)

- The Disapprove User window lets the admin disapprove of a user.

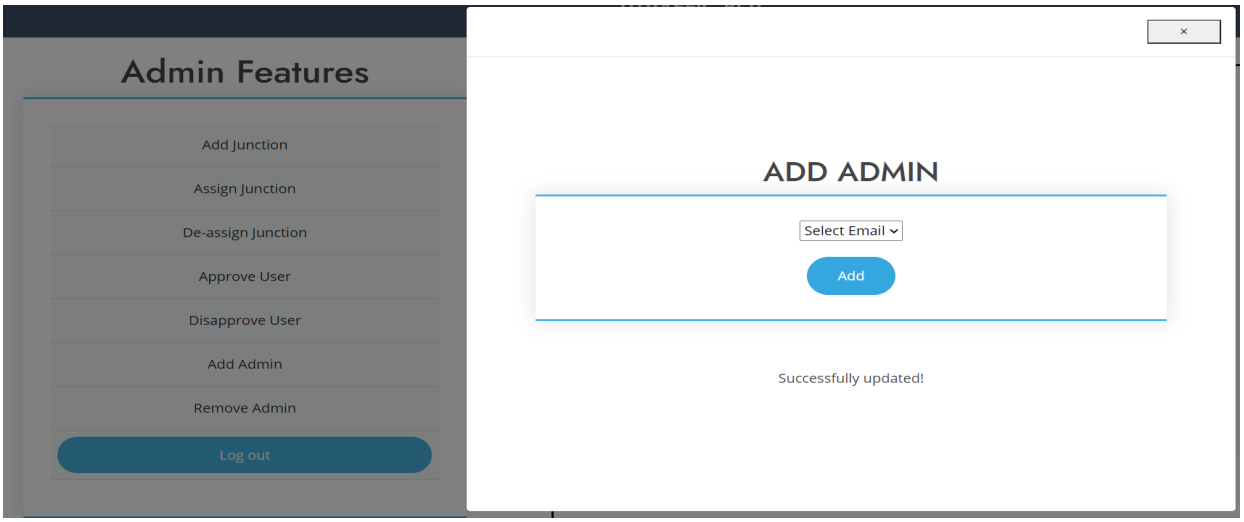


Fig. 80 : Admin Page (Add Admin)

- The Add Admin window lets the admin make another user an admin and thus granting them admin permissions.

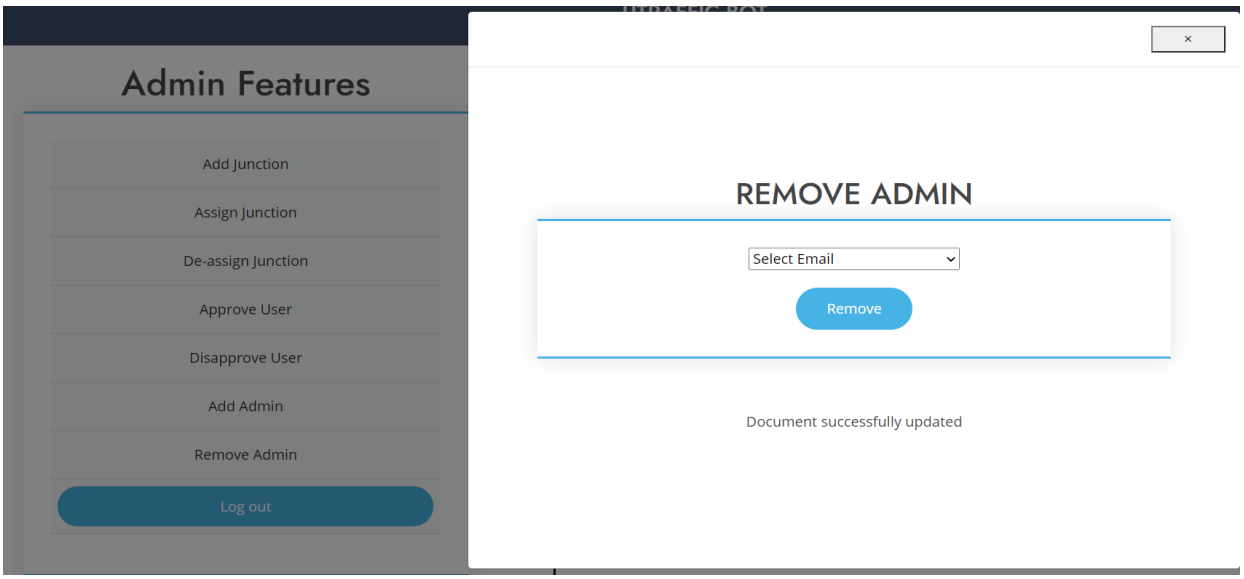


Fig. 81 : Admin Page (Remove Admin)

- The Remove Admin window lets the admin revoke the admin permissions from a user..

CHAPTER – 6 CONCLUSION

6.1 CONCLUSION

With the help of computer vision and image processing, we detect the density of the traffic and make the **traffic signals dynamic** as per the analysed data.

With this system, we not only make traffic signals intelligent but also help the **emergency service vehicles** with easy passage and thus helping people by saving their precious time.

Hence we would like to conclude that ***"the new age of traffic management will begin, and being stuck in a traffic jam will no longer be an excuse to be late"***.

6.2 ADVANTAGES

Efficient passage of vehicles is performed at the junction and hence time is saved.

Prevention of Emergency Vehicles from getting stuck in the traffic by giving them Priority access.

Efficient management of traffic with an aim to reduce the pollution, wastage of fuel, cost of transportation and stress to drivers.

6.3 LIMITATIONS

In a scenario when there is a power cut the signal won't be able to work and communication will be interrupted.

With more images in our dataset, we can improve our accuracy.

In case of harsh weather conditions, the model won't be able to work properly as the images fetched will not be clearer.

6.4 EXTENSION OF OUR WORK

Linking of multiple signals together to determine the flow of traffic and suggesting efficient traffic routes in order to reduce the time of travelling.

References

[1] Nellore K, Hancke GP. Traffic Management for Emergency Vehicle Priority Based on Visual Sensing. Sensors (Basel). 2016;16(11):1892. Published 2016 Nov 10. doi:10.3390/s16111892

[2] Dr. Ruhi Subhash Bakhare, Dr. Rakhi Vyankatesh Vyaghra. An Automated System And Method To Control Traffic Signals For Emergency Vehicles. Application No.202021026234 A

[3] Kevin Clare Eichhorst. Adaptive Traffic Signal Preemption. Patent No.: US 9,299,253 B2. Date of Patent: Mar. 29, 2016

[4] Himani Jerath, Dushyant Kumar Singh, Saurabh Singh. EMERGENCY TRAFFIC CONTROL AND MANAGEMENT SYSTEM. Publication No. 51/2019.

[5] DR. A. CECIL DONALD, DR. STEPHEN R, DR. A. DALVIN VINOTH KUMAR, DR. SANTIAGO SEBASTIN. TRAFFIC MONITORING: INTELLIGENT TRAFFIC MONITORING SYSTEM USING AI TECHNIQUES. Publication No. 23/2020

[6] Shabnam Sayyed, Prajakta Date, Richa Gautam, Gayatri Bhandari. Design of Dynamic Traffic Signal Control System. International Journal of Engineering Research & Technology (IJERT) ISSN: 2278-0181 Vol. 3 Issue 1, January - 2014.

[7] Choudekar, Pallavi & Banerjee, Sayanti & SAYANTI, & Muju, Mohan. (2011). REAL TIME TRAFFIC LIGHT CONTROL USING IMAGE PROCESSING. Indian Journal of Computer Science and Engineering.

[8] Sakhare, Nitin & Tatale, Subhash & Sakhare, Sachin & Dusaane, Hemant & Puri, Mamta & Girme, Pratika & Sankpal, Rutuja & Ghule, Padmavati. (2020). Image Processing and IoT Based Dynamic Traffic Management System. International Journal of Scientific Research in Science, Engineering and Technology. 180-188. 10.32628/IJSRSET207230.

[9] H. Razalli, R. Ramli and M. H. Alkawaz, "Emergency Vehicle Recognition and Classification Method Using HSV Color Segmentation," 2020 16th IEEE International Colloquium on Signal Processing & Its Applications (CSPA), Langkawi, Malaysia, 2020, pp. 284-289, doi: 10.1109/CSPA48992.2020.9068695.

[10] Times of India - https://timesofindia.indiatimes.com/city/mumbai/mumbaikars-lose-85-minutes-a-day-in-traffic-jams/articleshow/73285609.cms

[11] Times of India - https://timesofindia.indiatimes.com/india/bengaluru-has-worlds-worst-traffic-congestion-mumbai-at-number-4/articleshow/73747725.cms

[12] Research Gate - https://www.researchgate.net/publication/331346712_Intelligent_Traffic_Light_System_Using_Computer_Vision_with_Android_Monitoring_and_Control

[13] Labcenter - https://www.labcenter.com/projectnotes/

[14] Roboflow - https://blog.roboflow.com/train-yolov4-tiny-on-custom-data-lighting-fast-detection/

[15] Traffic signal design https://nptel.ac.in/content/storage2/courses/105101087/downloads/Lec-41.pdf

APPENDIX - A

Print Back

College : SARVAJANIK COLLEGE OF ENGINEERING & TECHNOLOGY, SURAT
StudentName : Bardolia Priyank Harshad
EnrollmentNo : 170420107501 Department : Computer Engineering
MobileNo : 9825400177 Discipline : BE
Email : 170420107501.co17s2@scet.ac.in Semester : Semester 8

PPR Details

Periodic Progess Report : First PPR

Project : ULTRA-FFIC BOT

Status : Reviewed

1. What Progress you have made in the Project ?
SRS. Implemented Vehicle detection on images using Yolov3. Simulator

2. What challenge you have faced ?
No challenges faced yet.

3. What support you need ?
No support needed yet.

4. Which literature you have referred ?
We read several research papers that were relevant to our project.

Comments

Comment by Internal Guide :
Work done as per guidance given

Comment by External Guide :
None

Comment by HOD :
None

Comment by Principal :
None

Comment by University Admin :
None

Print Back

College	:	SARVAJANIK COLLEGE OF ENGINEERING & TECHNOLOGY, SURAT			
StudentName	:	Bardolia Priyank Harshad			
EnrollmentNo	:	170420107501	Department	:	Computer Engineering
MobileNo	:	9825400177	Discipline	:	BE
Email	:	170420107501.co17s2@scet.ac.in	Semester	:	Semester 8

PPR Details

Periodic Progess Report : Second PPR

Project : ULTRA-FFIC BOT

Status : Reviewed

1. What Progress you have made in the Project ?

We started the implementation of module-2. We continued the implementation of the android application and website. We started the image-based implementation of module-1 on raspberry pi.

2. What challenge you have faced ?

Due to covid-19, we faced difficulty working together on hardware while being at home. The execution of Object detection on Raspberry pi was taking a lot of time for execution.

3. What support you need ?

We faced difficulty getting hardware available among team members.

4. Which literature you have referred ?

We read several technical blogs.

Comments

Comment by Internal Guide :
Work done as per guidance given

Comment by External Guide :
None

Comment by HOD :
None

Comment by Principal :
None

Comment by University Admin :
None

Print Back

College	:	SARVAJANIK COLLEGE OF ENGINEERING & TECHNOLOGY, SURAT			
StudentName	:	Bardolia Priyank Harshad			
EnrollmentNo	:	170420107501	Department	:	Computer Engineering
MobileNo	:	9825400177	Discipline	:	BE
Email	:	170420107501.co17s2@scet.ac.in	Semester	:	Semester 8

PPR Details

Periodic Progess Report : Third PPR

Project : ULTRA-FFIC BOT

Status : Reviewed

1. What Progress you have made in the Project ?

We started the video-based implementation of module-1 and module-2 on raspberry pi. We worked on the integration of the database for the android application and website.

2. What challenge you have faced ?

We noticed that the direct implementation of module-1 and module-2 on the raspberry pi is too slow to be practical. Data from the website and android app were inconsistent.

3. What support you need ?

Requirement of re-implementing our modules on a high processing server. Making our database consistent.

4. Which literature you have referred ?

We read some forums and technical articles relevant to our project.

Comments

Comment by Internal Guide :
Work done as per guidance given

Comment by External Guide :
None

Comment by HOD :
None

Comment by Principal :
None

Comment by University Admin :
None

Print Back

College	:	SARVAJANIK COLLEGE OF ENGINEERING & TECHNOLOGY, SURAT			
StudentName	:	Bardolia Priyank Harshad			
EnrollmentNo	:	170420107501	Department	:	Computer Engineering
MobileNo	:	9825400177	Discipline	:	BE
Email	:	170420107501.co17s2@scet.ac.in	Semester	:	Semester 8

PPR Details

Periodic Progess Report : Forth PPR

Project : ULTRA-FFIC BOT

Status : Reviewed

1. What Progress you have made in the Project ?

We implemented module-1 and module-2 on a high processing server. We solved the problem of database inconsistency. We integrated all the modules of our project. We achieved a Testing Accuracy of 95.4% We started working on BMC and the project report.

2. What challenge you have faced ?

Execution time for running the object detection on video was very high. Lookalike vehicles wrt emergency vehicles were detected.

3. What support you need ?

Our guide guided us through the steps.

4. Which literature you have referred ?

We referred to GTU documents such as “http://files.gtu.ac.in/circulars/16JUL/14072016.pdf”

Comments

Comment by Internal Guide :
Work done as per guidance given

Comment by External Guide :
None

Comment by HOD :
None

Comment by Principal :
None

Comment by University Admin :
None

Print Back

College : SARVAJANIK COLLEGE OF ENGINEERING & TECHNOLOGY, SURAT
StudentName : Mahatma Saloni Nileshkumar
EnrollmentNo : 170420107530
MobileNo : 9099237357
Email : 170420107530.co17s2@scet.ac.in
Department : Computer Engineering
Discipline : BE
Semester : Semester 8

PPR Details

Periodic Progess Report : First PPR

Project : ULTRA-FFIC BOT

Status : Reviewed

1. What Progress you have made in the Project ?
SRS. Implemented Vehicle detection on images using Yolov3. Simulator

2. What challenge you have faced ?
No challenges faced yet

3. What support you need ?
No support needed yet.

4. Which literature you have referred ?
We read several research papers that were relevant to our project.

Comments

Comment by Internal Guide :
Work done as per guidance given

Comment by External Guide :
None

Comment by HOD :
None

Comment by Principal :
None

Comment by University Admin :
None

Print Back

College	:	SARVAJANIK COLLEGE OF ENGINEERING & TECHNOLOGY, SURAT			
StudentName	:	Mahatma Saloni Nileshkumar			
EnrollmentNo	:	170420107530	Department	:	Computer Engineering
MobileNo	:	9099237357	Discipline	:	BE
Email	:	170420107530.co17s2@scet.ac.in	Semester	:	Semester 8

PPR Details

Periodic Progess Report : Second PPR

Project : ULTRA-FFIC BOT

Status : Reviewed

1. What Progress you have made in the Project ?

We started the implementation of module-2. We continued the implementation of the android application and website. We started the image-based implementation of module-1 on raspberry pi.

2. What challenge you have faced ?

Due to covid-19, we faced difficulty working together on hardware while being at home. The execution of Object detection on Raspberry pi was taking a lot of time for execution.

3. What support you need ?

We faced difficulty getting hardware available among team members.

4. Which literature you have referred ?

We read several technical blogs.

Comments

Comment by Internal Guide :

Work done as per guidance given

Comment by External Guide :

None

Comment by HOD :

None

Comment by Principal :

None

Comment by University Admin :

None

Print Back

College	:	SARVAJANIK COLLEGE OF ENGINEERING & TECHNOLOGY, SURAT			
StudentName	:	Mahatma Saloni Nileshkumar			
EnrollmentNo	:	170420107530	Department	:	Computer Engineering
MobileNo	:	9099237357	Discipline	:	BE
Email	:	170420107530.co17s2@scet.ac.in	Semester	:	Semester 8

PPR Details

Periodic Progess Report : Third PPR

Project : ULTRA-FFIC BOT

Status : Reviewed

1. What Progress you have made in the Project ?

We started the video-based implementation of module-1 and module-2 on raspberry pi. We worked on the integration of the database for the android application and website.

2. What challenge you have faced ?

We noticed that the direct implementation of module-1 and module-2 on the raspberry pi is too slow to be practical. Data from the website and android app were inconsistent.

3. What support you need ?

Requirement of re-implementing our modules on a high processing server. Making our database consistent.

4. Which literature you have referred ?

We read some forums and technical articles relevant to our project.

Comments

Comment by Internal Guide :
Work done as per guidance given

Comment by External Guide :
None

Comment by HOD :
None

Comment by Principal :
None

Comment by University Admin :
None

Print Back

College	:	SARVAJANIK COLLEGE OF ENGINEERING & TECHNOLOGY, SURAT			
StudentName	:	Mahatma Saloni Nileshkumar			
EnrollmentNo	:	170420107530	Department	:	Computer Engineering
MobileNo	:	9099237357	Discipline	:	BE
Email	:	170420107530.co17s2@scet.ac.in	Semester	:	Semester 8

PPR Details

Periodic Progess Report : Forth PPR

Project : ULTRA-FFIC BOT

Status : Reviewed

1. What Progress you have made in the Project ?

We implemented module-1 and module-2 on a high processing server. We solved the problem of database inconsistency. We integrated all the modules of our project. We achieved a Testing Accuracy of 95.4% We started working on BMC and the project report.

2. What challenge you have faced ?

Execution time for running the object detection on video was very high. Lookalike vehicles wrt emergency vehicles were detected.

3. What support you need ?

Our guide guided us through the steps.

4. Which literature you have referred ?

We referred to GTU documents such as “http://files.gtu.ac.in/circulars/16JUL/14072016.pdf”

Comments

Comment by Internal Guide :
Work done as per guidance given

Comment by External Guide :
None

Comment by HOD :
None

Comment by Principal :
None

Comment by University Admin :
None

Print Back

College	:	SARVAJANIK COLLEGE OF ENGINEERING & TECHNOLOGY, SURAT			
StudentName	:	Mistry Vishal Dharmeshbhai			
EnrollmentNo	:	170420107536	Department	:	Computer Engineering
MobileNo	:	9909999629	Discipline	:	BE
Email	:	170420107536.co17s2@scet.ac.in	Semester	:	Semester 8

PPR Details

Periodic Progess Report : First PPR

Project : ULTRA-FFIC BOT

Status : Reviewed

1. What Progress you have made in the Project ?

1. SRS. 2. Implemented Vehicle detection on images using Yolov3. 3. Simulator

2. What challenge you have faced ?

No challenges faced yet

3. What support you need ?

No support needed yet.

4. Which literature you have referred ?

We read several research papers that were relevant to our project.

Comments

Comment by Internal Guide :

Work done as per guidance given

Comment by External Guide :

None

Comment by HOD :

None

Comment by Principal :

None

Comment by University Admin :

None

Print Back

College	:	SARVAJANIK COLLEGE OF ENGINEERING & TECHNOLOGY, SURAT			
StudentName	:	Mistry Vishal Dharmeshbhai			
EnrollmentNo	:	170420107536	Department	:	Computer Engineering
MobileNo	:	9909999629	Discipline	:	BE
Email	:	170420107536.co17s2@scet.ac.in	Semester	:	Semester 8

PPR Details

Periodic Progess Report : Second PPR

Project : ULTRA-FFIC BOT

Status : Reviewed

1. What Progress you have made in the Project ?

1. We started the implementation of module-2. 2. We continued the implementation of the android application and website. 3. We started the image-based implementation of module-1 on raspberry pi.

2. What challenge you have faced ?

1. Due to covid-19, we faced difficulty working together on hardware while being at home. 2. The execution of Object detection on Raspberry pi was taking a lot of time for execution.

3. What support you need ?

We faced difficulty getting hardware available among team members.

4. Which literature you have referred ?

We read several technical blogs.

Comments

Comment by Internal Guide :
Work done as per guidance given

Comment by External Guide :
None

Comment by HOD :
None

Comment by Principal :
None

Comment by University Admin :
None

Print Back

College	:	SARVAJANIK COLLEGE OF ENGINEERING & TECHNOLOGY, SURAT			
StudentName	:	Mistry Vishal Dharmeshbhai			
EnrollmentNo	:	170420107536	Department	:	Computer Engineering
MobileNo	:	9909999629	Discipline	:	BE
Email	:	170420107536.co17s2@scet.ac.in	Semester	:	Semester 8

PPR Details

Periodic Progess Report : Third PPR

Project : ULTRA-FFIC BOT

Status : Reviewed

1. What Progress you have made in the Project ?

1. We started the video-based implementation of module-1 and module-2 on raspberry pi. 2. We worked on the integration of the database for the android application and website.

2. What challenge you have faced ?

1. We noticed that the direct implementation of module-1 and module-2 on the raspberry pi is too slow to be practical. 2. Data from the website and android app were inconsistent.

3. What support you need ?

1. Requirement of re-implementing our modules on a high processing server. 2. Making our database consistent.

4. Which literature you have referred ?

We read some forums and technical articles relevant to our project.

Comments

Comment by Internal Guide :
Work done as per guidance given

Comment by External Guide :
None

Comment by HOD :
None

Comment by Principal :
None

Comment by University Admin :
None

Print Back

College	:	SARVAJANIK COLLEGE OF ENGINEERING & TECHNOLOGY, SURAT			
StudentName	:	Mistry Vishal Dharmeshbhai			
EnrollmentNo	:	170420107536	Department	:	Computer Engineering
MobileNo	:	9909999629	Discipline	:	BE
Email	:	170420107536.co17s2@scet.ac.in	Semester	:	Semester 8

PPR Details

Periodic Progess Report : Forth PPR

Project : ULTRA-FFIC BOT

Status : Reviewed

1. What Progress you have made in the Project ?

1. We implemented module-1 and module-2 on a high processing server. 2. We solved the problem of database inconsistency. 3. We integrated all the modules of our project. 4. We achieved a Testing Accuracy of 95.4% 5. We started working on BMC and the project report.

2. What challenge you have faced ?

1. Execution time for running the object detection on video was very high. 2. Lookalike vehicles wrt emergency vehicles were detected.

3. What support you need ?

Our guide guided us through the steps.

4. Which literature you have referred ?

We referred to GTU documents such as 14072016.pdf (gtu.ac.in)

Comments

Comment by Internal Guide :

Work done as per guidance given

Comment by External Guide :

None

Comment by HOD :

None

Comment by Principal :

None

Comment by University Admin :

None

Print Back

College : SARVAJANIK COLLEGE OF ENGINEERING & TECHNOLOGY, SURAT
StudentName : Siddhant Parmar
EnrollmentNo : 170420107552
MobileNo : 9737017333
Email : 170420107552.co17s2@gmail.com
Department : Computer Engineering
Discipline : BE
Semester : Semester 8

PPR Details

Periodic Progess Report : First PPR

Project : ULTRA-FFIC BOT

Status : Reviewed

1. What Progress you have made in the Project ?

1. SRS. 2. Implemented Vehicle detection on images using Yolov3. 3. Simulator

2. What challenge you have faced ?

No challenges faced yet

3. What support you need ?

No support needed yet.

4. Which literature you have referred ?

We read several research papers that were relevant to our project.

Comments

Comment by Internal Guide :

Work done as per guidance given

Comment by External Guide :

None

Comment by HOD :

None

Comment by Principal :

None

Comment by University Admin :

None

Print Back

College	:	SARVAJANIK COLLEGE OF ENGINEERING & TECHNOLOGY, SURAT			
StudentName	:	Siddhant Parmar			
EnrollmentNo	:	170420107552	Department	:	Computer Engineering
MobileNo	:	9737017333	Discipline	:	BE
Email	:	170420107552.co17s2@gmail.com	Semester	:	Semester 8

PPR Details

Periodic Progess Report : Second PPR

Project : ULTRA-FFIC BOT

Status : Reviewed

1. What Progress you have made in the Project ?

1. We started the implementation of module-2. 2. We continued the implementation of the android application and website. 3. We started the image-based implementation of module-1 on raspberry pi.

2. What challenge you have faced ?

1. Due to covid-19, we faced difficulty working together on hardware while being at home. 2. The execution of Object detection on Raspberry pi was taking a lot of time for execution.

3. What support you need ?

We faced difficulty getting hardware available among team members.

4. Which literature you have referred ?

We read several technical blogs.

Comments

Comment by Internal Guide :

Work done as per guidance given

Comment by External Guide :

None

Comment by HOD :

None

Comment by Principal :

None

Comment by University Admin :

None

Print Back

College	:	SARVAJANIK COLLEGE OF ENGINEERING & TECHNOLOGY, SURAT			
StudentName	:	Siddhant Parmar			
EnrollmentNo	:	170420107552	Department	:	Computer Engineering
MobileNo	:	9737017333	Discipline	:	BE
Email	:	170420107552.co17s2@gmail.com	Semester	:	Semester 8

PPR Details

Periodic Progess Report : Third PPR

Project : ULTRA-FFIC BOT

Status : Reviewed

1. What Progress you have made in the Project ?

1. We started the video-based implementation of module-1 and module-2 on raspberry pi. 2. We worked on the integration of the database for the android application and website.

2. What challenge you have faced ?

1. We noticed that the direct implementation of module-1 and module-2 on the raspberry pi is too slow to be practical. 2. Data from the website and android app were inconsistent.

3. What support you need ?

1. Requirement of re-implementing our modules on a high processing server. 2. Making our database consistent.

4. Which literature you have referred ?

We read some forums and technical articles relevant to our project.

Comments

Comment by Internal Guide :
Work done as per guidance given

Comment by External Guide :
None

Comment by HOD :
None

Comment by Principal :
None

Comment by University Admin :
None

Print Back

College	:	SARVAJANIK COLLEGE OF ENGINEERING & TECHNOLOGY, SURAT			
StudentName	:	Siddhant Parmar			
EnrollmentNo	:	170420107552	Department	:	Computer Engineering
MobileNo	:	9737017333	Discipline	:	BE
Email	:	170420107552.co17s2@gmail.com	Semester	:	Semester 8

PPR Details

Periodic Progess Report : Forth PPR

Project : ULTRA-FFIC BOT

Status : Reviewed

1. What Progress you have made in the Project ?

1. We implemented module-1 and module-2 on a high processing server. 2. We solved the problem of database inconsistency. 3. We integrated all the modules of our project. 4. We achieved a Testing Accuracy of 95.4% 5. We started working on BMC and the project report.

2. What challenge you have faced ?

1. Execution time for running the object detection on video was very high. 2. Lookalike vehicles wrt emergency vehicles were detected.

3. What support you need ?

Our guide guided us through the steps.

4. Which literature you have referred ?

We referred to GTU documents such as “http://files.gtu.ac.in/circulars/16JUL/14072016.pdf”

Comments

Comment by Internal Guide :

work done as per guidance given

Comment by External Guide :

None

Comment by HOD :

None

Comment by Principal :

None

Comment by University Admin :

None

Print Back

College	:	SARVAJANIK COLLEGE OF ENGINEERING & TECHNOLOGY, SURAT			
StudentName	:	Srivastava Umang Vipul			
EnrollmentNo	:	170420107554	Department	:	Computer Engineering
MobileNo	:	9426891924	Discipline	:	BE
Email	:	umangs.leo@gmail.com	Semester	:	Semester 8

PPR Details

Periodic Progess Report : First PPR

Project : ULTRA-FFIC BOT

Status : Reviewed

1. What Progress you have made in the Project ?

SRS. Implemented Vehicle detection on images using Yolov3. Simulator

2. What challenge you have faced ?

No challenges faced yet

3. What support you need ?

No support needed yet.

4. Which literature you have referred ?

We read several research papers that were relevant to our project.

Comments

Comment by Internal Guide :

Work done as per guidance given

Comment by External Guide :

None

Comment by HOD :

None

Comment by Principal :

None

Comment by University Admin :

None

Print Back

College	:	SARVAJANIK COLLEGE OF ENGINEERING & TECHNOLOGY, SURAT			
StudentName	:	Srivastava Umang Vipul			
EnrollmentNo	:	170420107554	Department	:	Computer Engineering
MobileNo	:	9426891924	Discipline	:	BE
Email	:	umangs.leo@gmail.com	Semester	:	Semester 8

PPR Details

Periodic Progess Report : Second PPR

Project : ULTRA-FFIC BOT

Status : Reviewed

1. What Progress you have made in the Project ?

We started the implementation of module-2. We continued the implementation of the android application and website. We started the image-based implementation of module-1 on raspberry pi.

2. What challenge you have faced ?

Due to covid-19, we faced difficulty working together on hardware while being at home. The execution of Object detection on Raspberry pi was taking a lot of time for execution.

3. What support you need ?

We faced difficulty getting hardware available among team members.

4. Which literature you have referred ?

We read several technical blogs.

Comments

Comment by Internal Guide :
Work done as per guidance given

Comment by External Guide :
None

Comment by HOD :
None

Comment by Principal :
None

Comment by University Admin :
None

Print Back

College	:	SARVAJANIK COLLEGE OF ENGINEERING & TECHNOLOGY, SURAT			
StudentName	:	Srivastava Umang Vipul			
EnrollmentNo	:	170420107554	Department	:	Computer Engineering
MobileNo	:	9426891924	Discipline	:	BE
Email	:	umangs.leo@gmail.com	Semester	:	Semester 8

PPR Details

Periodic Progess Report : Third PPR

Project : ULTRA-FFIC BOT

Status : Reviewed

1. What Progress you have made in the Project ?

We started the video-based implementation of module-1 and module-2 on raspberry pi. We worked on the integration of the database for the android application and website.

2. What challenge you have faced ?

We noticed that the direct implementation of module-1 and module-2 on the raspberry pi is too slow to be practical. Data from the website and android app were inconsistent.

3. What support you need ?

Requirement of re-implementing our modules on a high processing server. Making our database consistent.

4. Which literature you have referred ?

We read some forums and technical articles relevant to our project.

Comments

Comment by Internal Guide :
Work done as per guidance given

Comment by External Guide :
None

Comment by HOD :
None

Comment by Principal :
None

Comment by University Admin :
None

Print Back

College	:	SARVAJANIK COLLEGE OF ENGINEERING & TECHNOLOGY, SURAT			
StudentName	:	Srivastava Umang Vipul			
EnrollmentNo	:	170420107554	Department	:	Computer Engineering
MobileNo	:	9426891924	Discipline	:	BE
Email	:	umangs.leo@gmail.com	Semester	:	Semester 8

PPR Details

Periodic Progess Report : Forth PPR

Project : ULTRA-FFIC BOT

Status : Reviewed

1. What Progress you have made in the Project ?

We implemented module-1 and module-2 on a high processing server. We solved the problem of database inconsistency. We integrated all the modules of our project. We achieved a Testing Accuracy of 95.4% We started working on BMC and the project report.

2. What challenge you have faced ?

Execution time for running the object detection on video was very high. Lookalike vehicles wrt emergency vehicles were detected.

3. What support you need ?

Our guide guided us through the steps.

4. Which literature you have referred ?

We referred to GTU documents such as “http://files.gtu.ac.in/circulars/16JUL/14072016.pdf”

Comments

Comment by Internal Guide :

Work done as per guidance given

Comment by External Guide :

None

Comment by HOD :

None

Comment by Principal :

None

Comment by University Admin :

None