INTELLIGENT
ROAD TRAFFIC CONTROL SYSTEM
DESIGN MANUAL

CO321 CO322 CO324 CO325 UNIFIED PROJECT

Group No 7

E/15/123: Wishma Herath
E/15/280: Pubudu Premathilaka
E/15/316: Suneth Samarasinghe

Deaprtment of Computer Engineering
Unviersity of Peradeniya

2019

Contents

List of Figures

1 Introduction

1.1 Overview

Vehicle travel across the world is increasing, especially in larger urban areas. It is a serious problem in traffic congestions in many major cities around the world and in these cities, it has become a nightmare for travelers. With the increase of population and urbanization, traffic congestions in cities have started to grow rapidly. This leads to lot of problems such as time wastage, bad fuel consumption, air pollution, noise pollution and much more. Since Traditional systems have implemented to follow a preset time schedule, they do not control variable flows coming near junctions.

One study done in Boston has proved the timings of 60 intersections in one district of the city could save 1.2 million dollars per year

There is a goal of customizing traffic flow vehicles in a junction. As the number of roads increases steadily, and the resources provided by the current infrastructure are limited, intelligent control of traffic in the future will become a very important issue. Therefore, to better accommodation of this increasing demand, traffic control algorithms need to be simulated and optimized.

1.2 Available solution

Most of the counties still majorly use the traditional method of controlling road traffic based on the preset time schedules. The problem we identified in this solution is, even if there are no vehicles on some lanes, still the time will be allocated for them and others have to wait for their turn. When collectively taken these small-time delays will result in huge time wastage and it comes with all the other results such as fuel wastage, environmental pollution, pileups during peak hours, induce drivers to neglect signals in lean hours and etc.

1.3 Our solution

We propose a system that can control the traffic while keeping track of the number of vehicles on each lane around the junction. Advantages of our solution

- Saves time by cutting out the unnecessary time periods allocated for lanes that do not have vehicles waiting.
- The cost and complexity of implementation is less.
- Data monitoring tools are provided.
- Data can be used for surveying purposes.
- Opportunity to experiment and analyze the performance of different traffic controlling algorithms

2 Hardware

2.1 Sensor Node

Sensor nodes are consist of two main functionalities. One is reading data from the vehicle counting sensors and another one is communicating with the Relay Node. It consists of an Arduino Nano microcontroller and a Radio Frequency transceiver module (NRF24l01). Each vehicle counter is directly connected to the Sensor node. To have errorless results (not to miss a vehicle) the Sensor Node should keep polling the sensors in a continuous loop. Since the Arduino Nano microcontroller has only one thread the only thing it will be able to do is to keep polling the sensors. But it should facilitate another feature which is communication with the Relay Node.

To overcome this problem we could use the interrupt feature of the NRF24l01 module. The interrupt pin on the module is called the "IRQ" (Interrupt ReQuest) pin.

The IRQ pin is normally HIGH and by default, it will send out a low pulse at three different events when

- It received data.
- It transmitted data.
- Transmission failed or no ack received

The maskIRQ (txok, txfail, txready) function from the TRMh20 RF24 library can be used to enable the above three functionalities.

We used this signal to trigger an external interrupt (we used digital pin 3) on the Arduino Nano module. Then, when the Relay Node makes a request to the Sensor node asking for the required information, the sensor node can sense the incoming request and it will only reply on that point of the time and it goes back to keep listening to the magnetic reed sensors. After implemented the abovementioned functionality, we measured the response time of the Sensor node, the time it takes to receive the request from the relay node and respond to it with the vehicle count. The average time was about 24ms. Also, we sent requests from the Relay node with very fewer time delays between two requests, to check whether the sensor node has time to respond to the request and go back to counting vehicles. We set the delay between requests to 10ms and verified that the sensor node can even handle requests at the amount of requesting frequency.

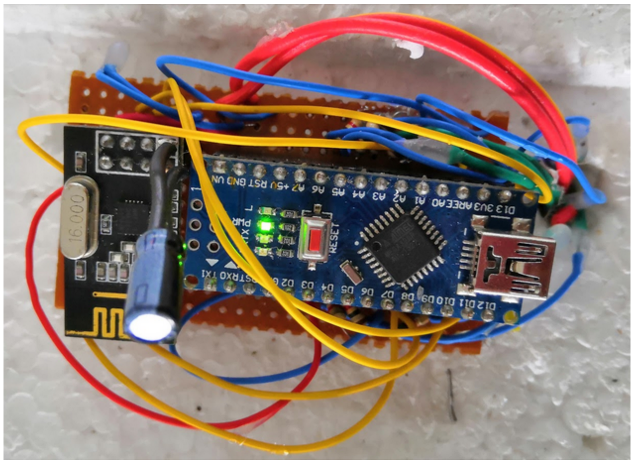


Figure 1: Sensor Node

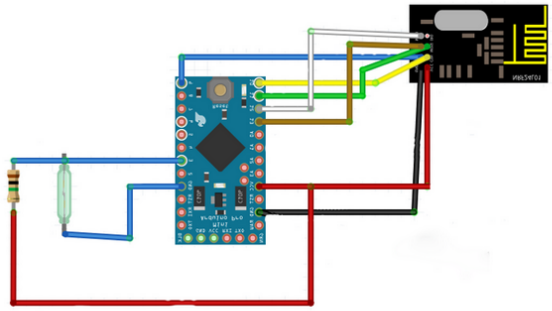


Figure 2: Sensor Node Circuit Diagram

2.2 Relay Node

The motherboard of the Relay Node is an ESP32 microcontroller and for radio communication, it has a radio transceiver module(NRF24l01). The relay node has three main functionalities.

First, it continuously gets data from the Sensor Nodes. The ESP32 microcontroller has the ability to handle two tasks at the same time using two threads; thread 0 and thread 1. One thread is programmed to keep requesting and receiving the traffic data from all the Sensor Nodes.

Second, the relay node communicates with the central server. The relay node connects with the central server using the WiFi module built-in with the ESP32 microcontroller. Depending on the control signals coming from the Central Server.

Third, the immediate decision to control color lights is taken by the relay node while all the data is communicated and further improved decision details are retrieved from the central server.

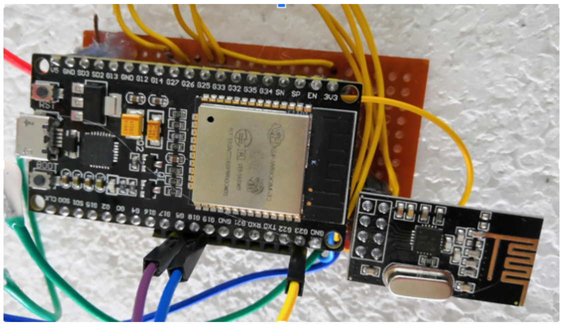


Figure 3: Relay Node

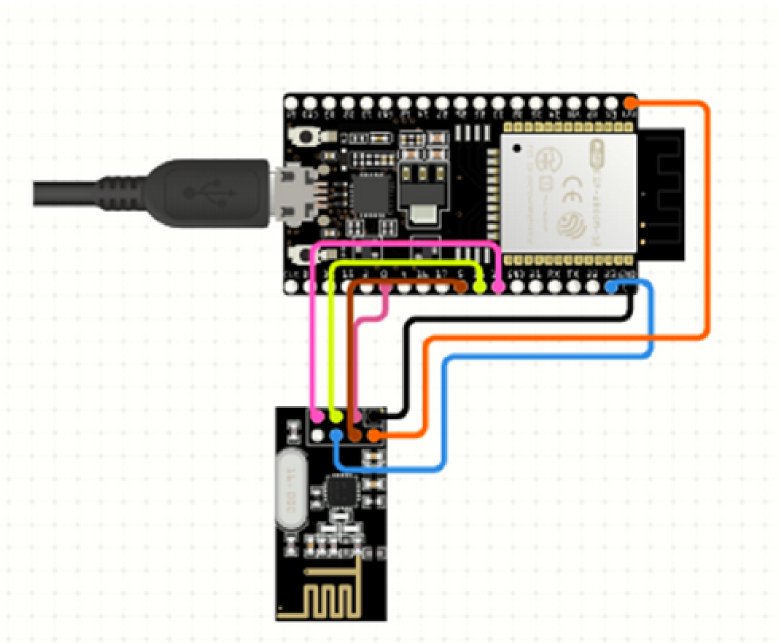


Figure 4: Relay Node Circuit Diagram

2.3 Color light arrangement

As per the junction model (4-way junction), we are considering for this project, the following diagram shows the arrangement of the color light modules we want per one junction. Since the vehicles which move on the green-colored arrow paths are free to pass the junction, we do not want to have a color light module for them.

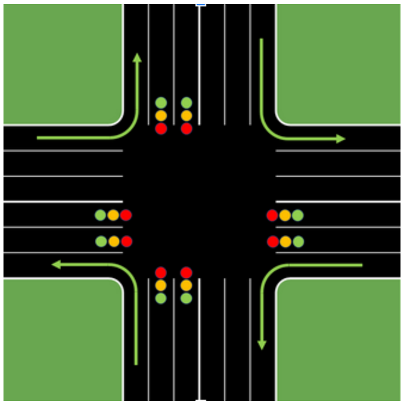


Figure 5: Color light arrangement Simulator View

Therefore, 8 color light modules for the junction prototype should be made. This week we were able to complete the color light modules.
Then, We found that controlling color lights (24 LEDs) using the same Relay Node (ESP32) is not practical since we have already used some pins for the NRF module on it. Therefore, we decided to use a shift register configuration to control those 24 LEDs. Because each LED should be addressable or lit separately. In that way, we would be able to use only 3 pins.

2.4 Shift Register Module

In order to avoid the pin limitation of ESP32 microcontroller(since color lights are controlled by ESP32), a shift register module has used in the prototype.

By using shift register module 8 color lights (8x3=24 Led bulbs) could be handled using just 3 GPIO pins in ESP32.

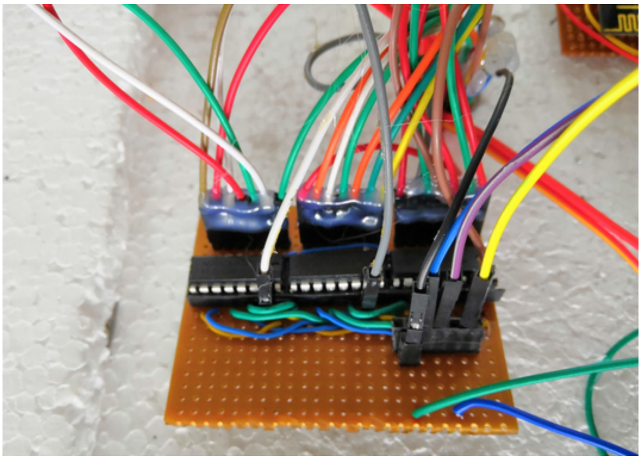


Figure 6: Register Module

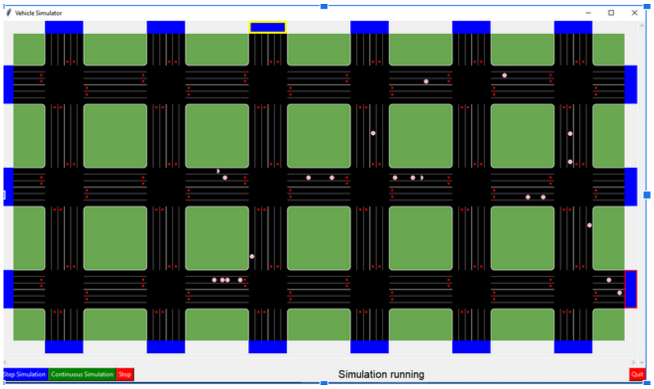
3 Software

3.1 Vehicle Simulator

As the demonstration has planned to do with a prototype of a junction that intersects two roads, other junctions will be demonstrated using the simulator software. Using a camera fixed from the top of the prototype, the video stream is taken to the simulator for the convenience of observing the behavior of the algorithm.

The simulator software will have the following features.

- Customizable dimensions for the junction grid. can be added to the grid and a specific path can be chosen using the GUI.
- Vehicles which has random paths can be added.
- The camera view of the top view of the prototype can be shown on the grid replacing a virtual junction.
- Add vehicle defining a custom path using GUI.
- Stepwise simulation.
- Continuous simulation.
- Add vehicles while the simulation is running.
- Simulating multiple vehicles at the same time.
- Pause and continue the simulation.
- Camera View added and it can work while the simulation is running.



Simulation running

Figure 7: Software Simulator

4 Server

4.1 MQTT

MQTT is a lightweight publish/subscribe messaging protocol designed for M2M(machine to machine) telemetry in low bandwidth environments. MQTT stands for Message Queuing Telemetry Transport.
MQTT messages light weight (small headers) thus offer high throughput. In our case this is not a problem since messages sent are small is size.
The first concept is the publish and subscribe system. In a publish and subscribe system, a device can publish a message on a topic, or it can be subscribed to a particular topic to receive messages.

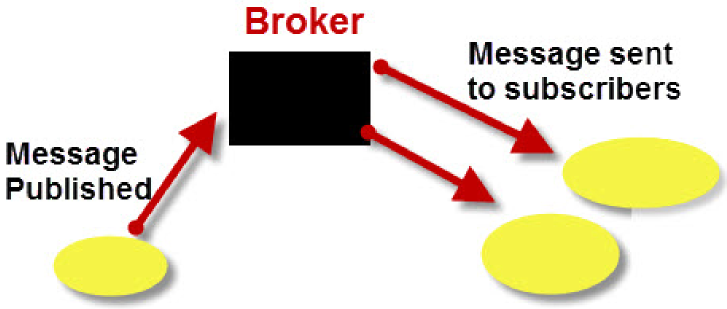


Figure 8: MQTT Publish Subscribe Model

4.2 MQTT Broker

The broker is primarily responsible for receiving all messages, filtering the messages, decide who is interested in them and then publishing the message to all subscribed clients. There are several brokers we can use.
In the project we have used Mosquitto broker which is installed on the raspberry pi device and it works as the server to the junction. Relay node works as the MQTT client to the Mosquitto server. Broker is programmed using python and broker program uploads data to the SQL database which is located on the raspberry pi device.
This local server contains all the traffic data in it's scope. Our propose is to connect local servers to a global server, process data and control traffic among multiple junctions optimally.

4.3 MQTT Client

In our system, client server communication works a hierarchy.Relay node works as a master to sensor nodes and Mosquitto broker works as a server to the realy node. The overall network diagram of the system can be shown as below.

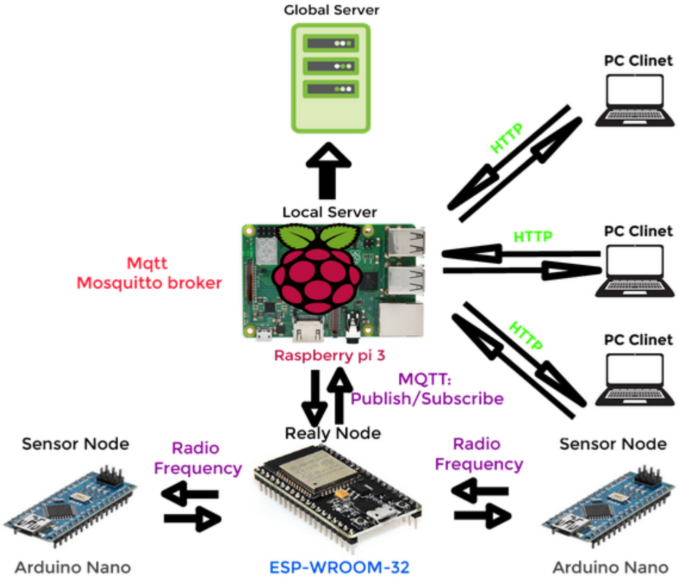


Figure 9: Complete Network Diagram with Protocols

5 Web

5.1 Importance

Optimizing the current traffic controlling system is a project that can be undertaken as a mandatory requirement for society as the current human lifestyle as mentioned before. Build up the website that fulfills the user needs will be a great help for the users to maintain their day to day lifestyle in the scope of traffic congestion.

Before starting up working on the web interface initially we did a feasibility study regarding our project in order to get the basic idea about what are the basic requirements that user aspects get from a website.

5.2 Feasibility study

we interviewed some stakeholders(day today drivers,.. etc) as our feasibility study. From the viewpoint of stakeholders, we decided to interview some students who use vehicles for their day to day traveling. So following are the comments that the stakeholders provided. Our website should.

- Shows the current traffic conditions in each junction of the city.
- Shows a History of the count of vehicles that passes through each road in a junction. (For survey kind of purposes).
- Provide registry for administrators in the aspect of updating the algorithm runs in the back-end.
- Provide a section to send feedback for the users.
- Data can be used for surveying purposes
- Provide a subscription facility to notify the traffic condition as the user's interest.

After taking the basic functionalities that the website should have, recognizing each entity and assign the appropriate attributes was done.

5.3 Front-End

Usability, or User Experience, is the art of making your website simple, user-friendly and easy to use. Understanding the customer's online behavior gives you insight into what works and what doesn't.
The Web interface was developed using HTML, CSS, java script. Apart from these the following libraries and plugins were used:

- Bootstrap
- W3-CSS

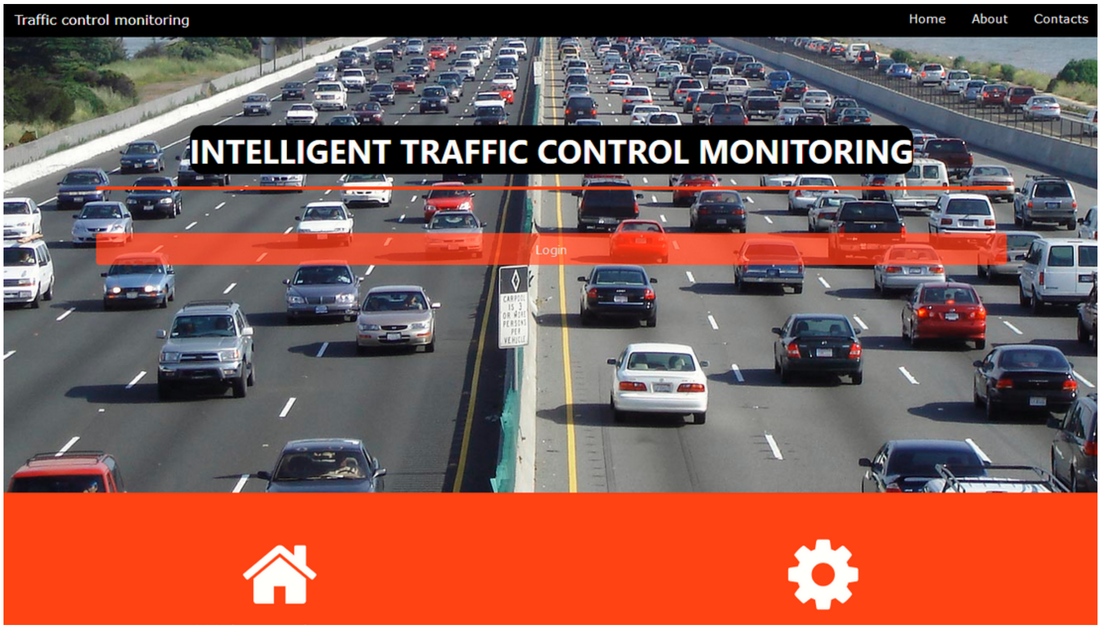
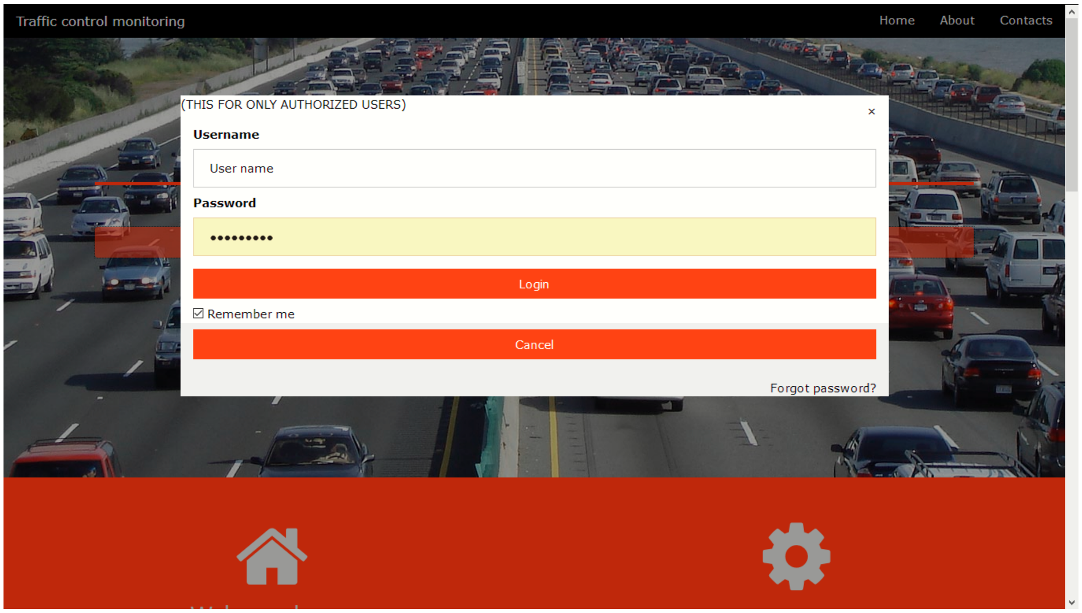

Traffic control monitoring
Home
About
Contacts
INTELLIGENT TRAFFIC CONTROL MONITORING
Login

Figure 10: Front-End Development

5.4 Wire-framing /Prototyping

When the website gets simpler it gets much easier to interact with the web interface. Anyhow to bring the best user interaction we should focus on to detect and eliminate potential usability issues. This step is called wireframing or prototyping. A wireframe is essentially a schematic, a blueprint or a pencil sketch of the website's content. It typically has no design elements or color; rather it focuses on substance. The exclusive focus is on content, structure, flow, and functionality without other visuals to distract the eye.The web interface consist of multiple features. Each feature of the Web Interface is described below.

Login window.This window can only be used by the authorized users who are working on the algorithm development which runs inside the server. Any new member can be added to the authorized users' category only with the support of an authorized user.



(THIS FOR ONLY AUTHORIZED USERS)

Username

User name

Password

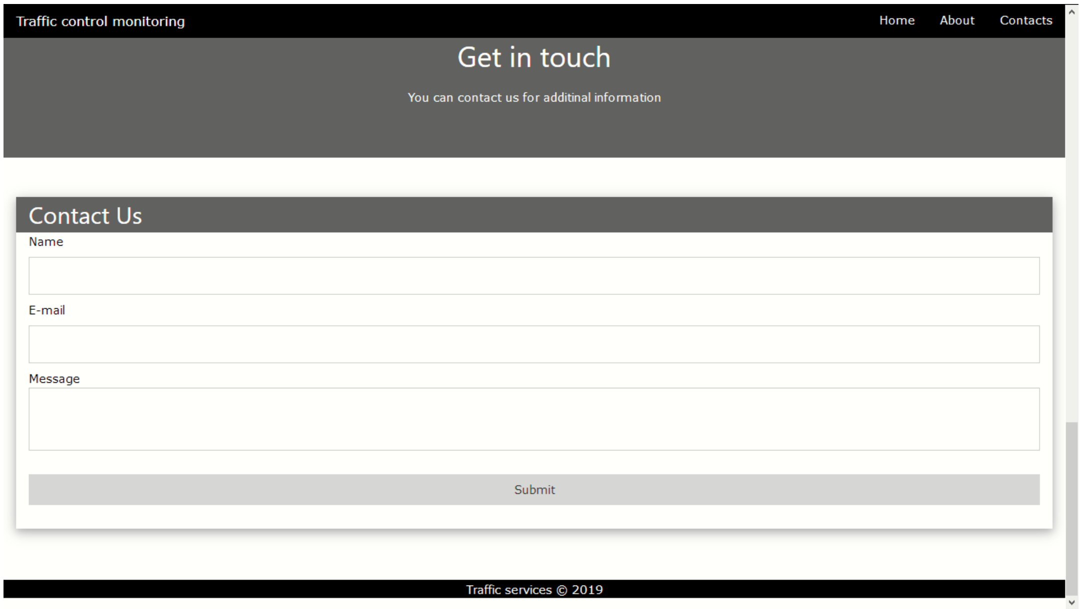
•••••••••

☑ Remember me

Figure 11: Logging Window

Viewing statics of each junction in the city. This feature can be used to view the real-time statistics and also the past statics on the vehicle count of the particular junction as per each lane. In order to provide the data on each junction, charts are used in such a way that the user can identify clearly the data and also compare the current data with the previous data. So the data are represented mainly using spline Area type and bar type.

Feedback Area. The web interface provides the facility that any user can feedback on survey data, what should add to the website, advantages, disadvantages etc.



You can contact us for additinal information

Traffic services © 2019

Figure 12: Feedback Window

6 Security

6.1 Sensor node - Relay node

In order to establish the communication between the sensor node and the relay node radio frequency used as the medium.
In this communication, we have implemented a protocol such that each sensor node works as a slave to the relay node. Here, relay node allows each sensor node to access it using a separate pipe. Therefore, data integrity is preserved without signal collisions.

6.2 Relay Node - Sever

MQTT mainly provides confidentiality, integrity, authentication to preserve security between publisher and the subscriber.
In the system in-order to preserve confidentiality shared key encryption is used between the publisher and the subscriber to encrypt the message.
In the broker Access control list is used to filter unrecognized topics and also it can be used to filter clients according to their assigned ids'.

6.3 Default ESP32 Security Features

- IEEE 802.11 standard security features all supported, including WFA, WPA/WPA2 and WAPI.
- Secure boot.
- Flash encryption.
- 1024-bit OTP, up to 768-bit for customers.
- Cryptographic hardware acceleration: AES, SHA-2, RSA, elliptic curve cryptography (ECC), random number generator (RNG)

6.4 Database

Mysql database is given access to only authorized users.Password lagging used to ensure the authentication. Users who have privileges can access to the database and read the content.

END