Crowd Indicating Bus Tracking System

GROUP 15

E/14/181 – Hashan Kavinda

E/14/178 – Yasitha Kavishka

E/14/298 – Jayamal Ruwantha

Contents

Introduction

Project background

A lot of people do not like to travel in crowded buses. We would love to travel sitting, sleeping. It will be very useful if there is a system to know whether the next bus on our route is full or not, to know the location of that bus, to know whether there are any alternative buses on route and etc.

So we came up with a solution which is an embedded system and huge network of various devices. Real-time bus data, which comes from the bus tracking system, shows the expected time in minutes until the arrival of the relevant bus. Customers can view this anticipated arrival time at a location via an app, website or, where available on the public information display system at the bus stop.

Purpose of the project

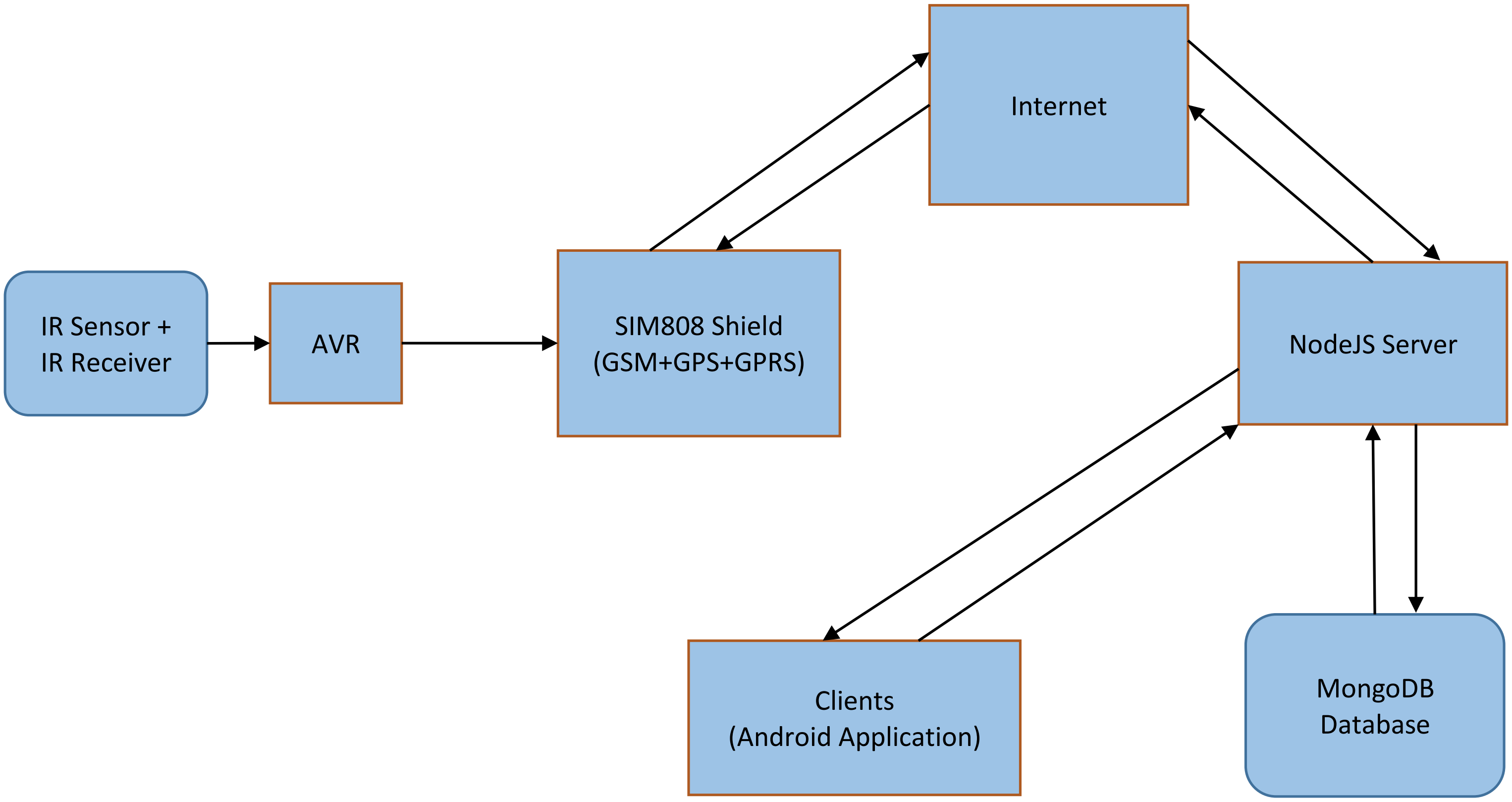
The main purpose in this idea is to check whether the buses are heavily loaded and late. So that if a person is willing to use another way of transportation, he can use that considering the information given by the system.

After all, the collected data will be very valuable also. They can be used to analyze and check which route has heavy load of people, whether the time differences among buses should be adjusted and should there be more buses in a route in a particular time and any more.

Architecture

Overview of the project

The very basic view of the project is shown below.



Hardware

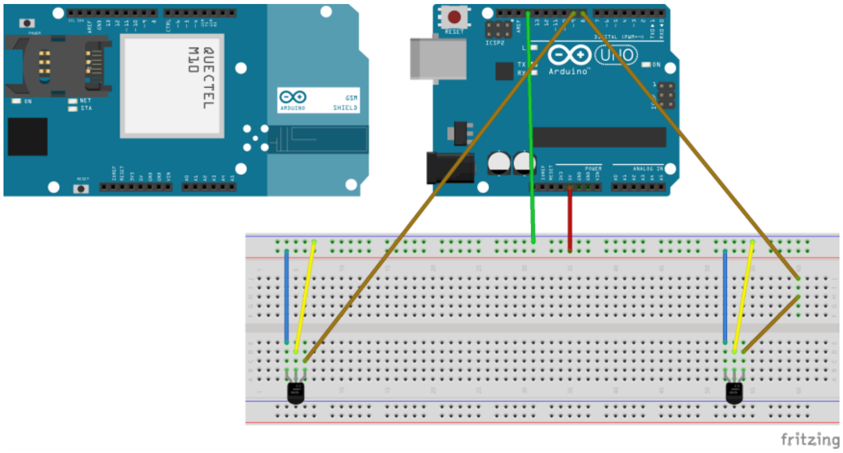
The hardware implementation for the system is as follows.

1. Arduino MEGA board – Process each and every signal and data flow in the inner system.

2. IR sensor – Gives a signal to count when a person crosses the path.

3. SIM808 shield (GSM+GPRS+GPS) – Retrieve the coordinates of the bus and help connect the system to the server via internet.

4. Rechargeable battery – Gives power to the system.



Above figure is a basic circuit design.

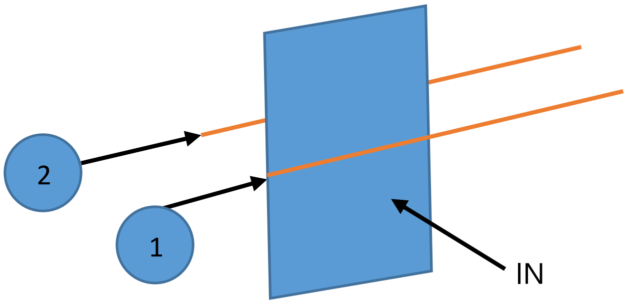
- SIM808 module is a complete Quad-Band GSM/GPRS module which combines GPS technology for satellite navigation. The compact design which integrated GPRS and GPS in a SMT package will significantly save both time and costs for customers to develop GPS enabled applications. Featuring an industry-standard interface and GPS function, it allows variable assets to be tracked seamlessly at any location and anytime with signal coverage.

- An infrared sensor is an electronic device, that emits in order to sense some aspects of the surroundings. An IR sensor can measure the heat of an object as well as detects the motion. IR Sensors use Infra-Red(IR) rays to emit and detect the amount of IR light that returns.

- At the testing phase it would be better to use a rechargeable battery, but as a final product it would be applicable to use the power source of the bus with regulated currents and voltages.

Deploying the system

Two IR sensors are placed parallel near the door.

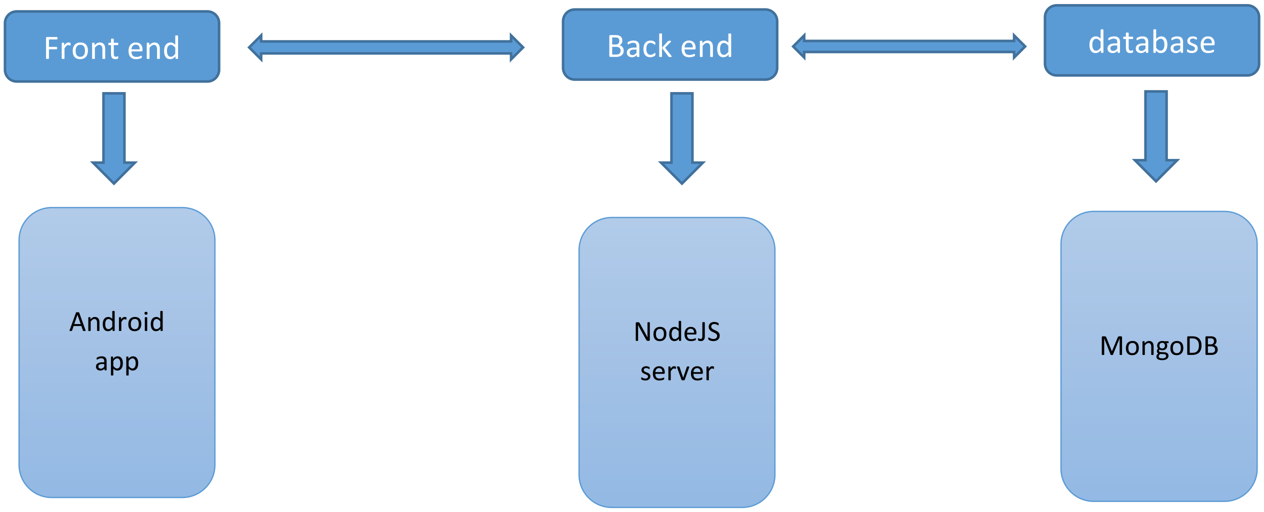


When people go through these two IR sensors, they are sequentially triggered one after the other. This method can be used to count the people inside the bus in a specific time. When a person enters the bus, he crosses 1st IR sensor before the 2nd one. When a person leaves the bus, he crosses 2nd IR sensor before the 1st one.

With the order of signals generated it can be verified whether people are entering the bus or leaving it. With that the exact number of people inside the bus can be confirmed in any specific time.

Software

The software implementation for the system is as follows.



Front end

Following technologies are used for the front end of the system.

- Android Studio
- JavaScript

Back end

Following technologies are used for the back end of the system.

- NodeJS
- JSON
- Google map APIs

➢ mongoDB which is a non-sql database system, is used as the database of the system.

Why an Android app?

The beneficiaries of this system are majority of common people. This system can be easily accessed via a mobile phone. So it is obvious that an Android app is the most suitable way of accessing this system for all of us.

Why NodeJS?

Node.js is a JavaScript runtime environment built on Chrome's V8 JavaScript engine. Node.js shines in real-time web applications employing push technology over websockets. There are web applications with real-time, two-way connections, where both the client and server can initiate communication, allowing them to exchange data freely, thanks to this.

Why JSON?

JSON stands for JavaScript Object Notation which is light-weighted designed such that humans can easily read data interchange and its execution. It does not require JavaScript to read or write because it is made in text format which is language independent and can be run everywhere. JSON syntax is very easy to use. Since its syntax is very small and light weighted that's the reason that it executes the response in the faster way. JSON server-side parsing is the strong point that indicates us to use the JSON on the server side.

Why Google map APIs?

The Google Maps APIs give developers several ways of embedding Google Maps into web pages or retrieving data from Google Maps, and allow for either simple use or extensive customization. The API automatically handles access to Google Maps servers, data downloading, map display, and response to map gestures.

Why mongoDB?

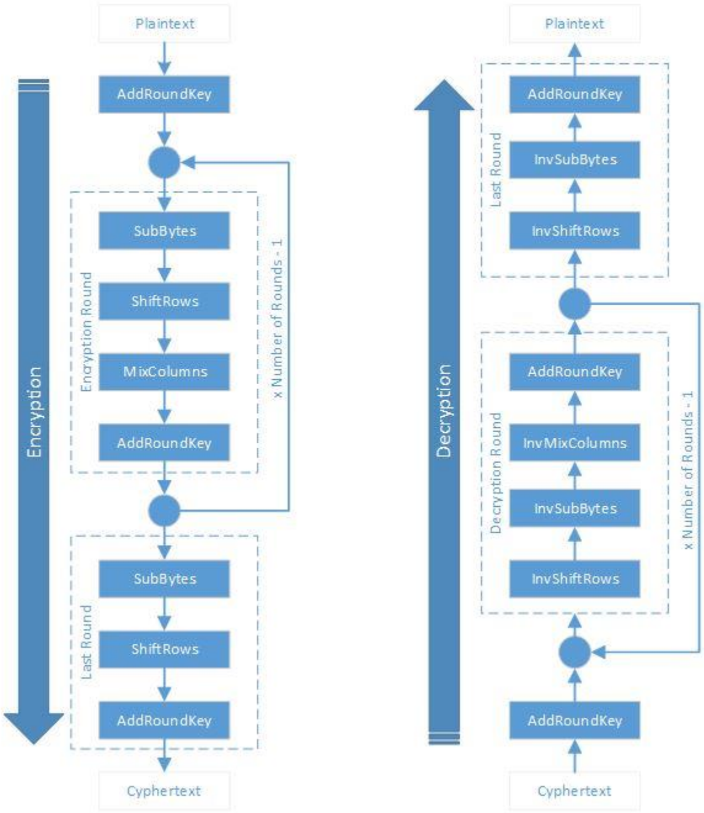
MongoDB is a document database in which one collection holds different documents. Number of fields, content and size of the document can differ from one document to another. MongoDB is easy to scale and the data is stored in the form of JSON style documents.

Network security

AES (Advanced Encryption Standard) is used for the system.

AES is a symmetric block cipher with a block length of 128 bits. It allows using three different key lengths: 128, 192 or 256 bits. The encryption/decryption algorithm consists of several rounds of processing; the number of rounds depends on the key length: 10 rounds for 128-bit keys, 12 rounds for 192-bit keys and 14 rounds for 256-bit keys. Except for the last round, all other rounds are identical. Each round of processing (excepting the last one) includes the following steps:

– one single-byte based substitution (SubBytes/InvSubBytes);
– row-wise permutation (ShiftRows/InvSubRows);
– column-wise mixing (MixColumns/InvMixColumns);
– addition of the round key (AddRoundKey).



Timeline

TASKS	Week 2	Week 3-5	Week 7	Week 11	Week 13	Week 14
Task 1	🟦					
Task 2		🟩				
Task 3			🟫	🟫	🟫	
Task 4				🟥	🟥	
Task 5				🟦	🟦	
Task 6						🟨

Task 1 – Choosing a suitable project

Task 2 – Learning through videos and reading, Obtaining the hardware

Task 3 – Designing and building the model

Task 4 – Building the client side approach (related to app)

Task 5 – Developing security side

Task 6 – Testing and applying modifications

Final Budget

Device	Quantity	Price
IR sensors	2	1550
SIM808 module	1	3750
Vero board normal	1	35
28pin IC base	1	10
Total	5345	

Prices are in LKR.

References

http://ctabustracker.com/bustime/home.jsp

https://www.tutorialspoint.com/mongodb/index.htm

https://www.seeedstudio.com/SIM808-GSM&GPRS-+-GPS-Module-p-2523.html