Smart Bus Ticket System Using IOT Technology

Mr. Poovaraghan, Assistant Professor

Sumitted By- Sumit Kumar, Piyush Kumar, Kartikay Tyagi, Ayush Kumar

Abstract— Reliability in public transport is of superb significance today. Millions of people journeying by using public buses waste a lot of time ready at bus stops. The focuses of paper on imparting a solution to handle the said hassle by utilise IoT technology stack. If the human beings traveling get correct actual time region of the buses alongside with estimate time for arrival at bus end primarily based on the actual time site visitors conditions, it will facilitate an normal extend in reliability on the buses of public. The answer proposed in this paper involves the use of the current net enabled devices on the bus (like the e-ticketing system) or a simple android tablet to seize the actual time area and ship to the servers. Accessing this location information from servers will be facilitated by Representational State Transfer (REST) APIs which customers can get entry to through android application, SMS or web-portals. Person to Stand in long queues at bus station or stands, quarrelling with conductors for trifle matters make his journey uncomfortable for the passengers. Because of that; we have proposed an notion for implementing clever card -technology for ticketing the passengers thos who are touring in bus. The clever card is in general primarily based on cutting-edge Radio Frequency Identification (RFID) technology. For this purpose, an interface is constructed between RFID setup and driver's mobile phone. The interface helps to send passenger ID from RFID reader to the driver's cell telephone by Bluetooth. The objective of this mission is to count the passenger the use of IR sensor and calculating the distance travelled by way of passenger routinely the usage of motor and u-slot of sensor, and the corresponding quantity is debited from main RFID card. In addition to that, in thought system the prevalence of accident information is routinely transmitted to the nearest clinic using GSM and GPS.

I. INTRODUCTION

Transportation act as primary usually centered vicinity for cities. With the ever growing and enhancing load on public transport systems, it is really essential to enlarge more effectivity in these systems. Due to vast no. of visitors conditions, over-crowding and many different similar issues, public buses are missing in punctuality and reliability. The required answer ought to no longer solely have an effect on the improvement in the services, however should the using component for expand in have faith on the public bus transport systems.

Reliability on public transport will expand when the traveller precisely is aware of when the buses will arrive to the bus cease or when will the bus attain the destination. The solution proposed in this paper harnesses the actual time vicinity of the buses tocalculate the estimated time for reaching a specific position. By saving vicinity on the server along with corresponding time stamps, we can calculate the timing for the bus to arrive as a bus stop, or time to attain a destination, using services like Google maps .

This paper affords a simple and value high quality way to make public transportation services 'more smart'. The paper will current the concept, science part, elements and the effects of implementation. The fundamental goal of the proposed answer is to decrease the costs concerned in implementation and to create a returned end that can scale up without difficulty with extend in demand.

The region facts amassed from the buses be accessible by means of customers as well as developers to harness the accumulated information to create greater fee from amassed data.

Typically, bus monitoring structures show statistics associated to the arriving buses, time to arrival and estimated departure time based on region of bus that is obtained with the aid of Radio Frequency (RF) transceiver [2]. There are such transceivers on each and every bus and bus stops so that the buses talk the region to bus stops. The microprocessors at bus quit then calculate estimated time for arrival and display it on a screen.

SMS services on GSM are used in some options as ability for communication in such cases. There are GSM modules with Global Positioning System (GPS) on the bus for better help purpose. All these periodically ship the vicinity to databases

through SMS. The region facts can then be accessed with the aid of sending SMS to receivers on the database servers.

Location data can be received with the aid of the of GPS based totally tracking devices hooked up on vehicles. These structures makes use of Hyper Text Transfer Protocol (HTTP) to ship vicinity data to the database[4]. Android gadgets in the buses circulation area information to servers, which can be accessed through android applications.

However, none of these systems can handle an increase in requests. The data consumption for sending the data from buses to servers is very high. The backend will crash if there is an increase.

II. OVERVIEWOFSMARTTRACKINGSYSTEMWITHSCALABLE BACKEND

The population is continuously rising nowadays at some stage in the world. The number of human beings that count number on public transport will expand multi-fold in the years to come. With upward thrust in population, there will be increase in the load on the public transport systems which will require increase in range of buses. The increase in the number of buses will end result to a remarkable amplify in the amount of statistics generated real time. The location information of these hundreds of buses will want to be treated via the back quit reliably. There certainly is a need for a scalable lower back ship for systems. The traditionally used HTTP protocol wants to be replaced by means of a better and light weight protocol. Using Message Queue Telemetry Transport (MQTT) will now not only end result in less internet bandwidth consumption compared to other protocols, however will also amplify the flexibility of the solution. MQTT has been tested to require minimum bandwidth even in a confined network. Use of MQTT protocol will enable the use of a distributed backend. With broker clustering in MQTT, the device will be fairly available, fault tolerant and scalable.

The area statistics accumulated from buses can be used to build records analytics applications. There can be a lot of useful insights related to site visitors conditions based on this data. Hence, it is essential to make this statistics handy to builders by using APIs to motivate its use for creating applications for the gain of society.

Fig. 1 suggests the mannequin of the proposed solution. The solution is divided into three parts. The first phase is the series of area and associated records from buses. The 2d phase is managing this facts on a disbursed IoT backend.

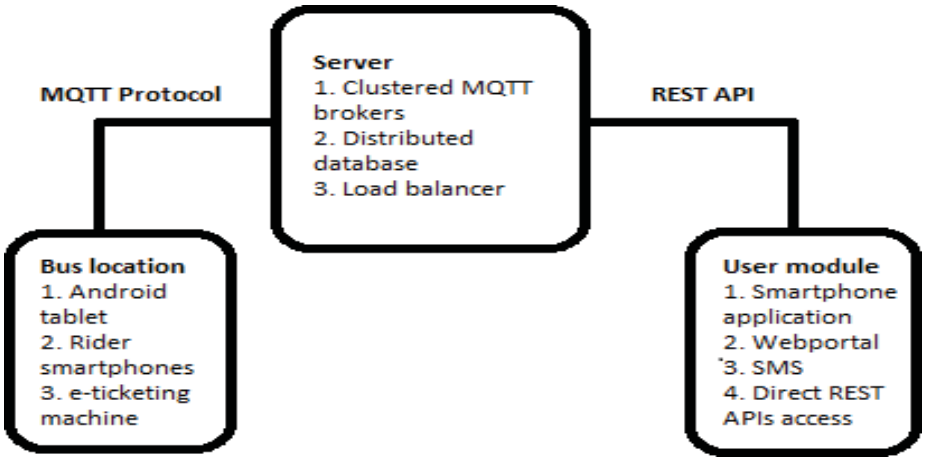

Fig. 1. Overview of the proposed solution

III. COLLECTION OF DATA FROMBUS

Collecting real time location data is the major issue to tackle due to involvement of hardware which will need heavy investment. The traditional solution is installing GPS and General Packet Radio Service (GPRS) enabled devices in every bus.

These devices though costly, serve as one time investment to provide services of importance.

The following are the cheaper alternatives available:

A. E-ticketing systems

Today, most buses in cities like Pune, India already have the integral hardware proving the capability for actual time region tracking. Having an e-ticketing system is the solution to forestall the installation of new high-priced hardware.

These structures are GPRS enabled. By community triangulation, it is feasible to get real time place of the devices. Thus, with area available and internet access, they can be used to share place records with customers with minor changes.

B. Android tablet phones

There are many cheap android units handy in market with web and GPS. These capsules can double as infotainment systems in the bus. They can display simple records like route number, destination, subsequent bus stop, time and also information associated to weather, nearby visitor spots. The public transport businesses can exhibit commercials on the tablets.

C. Crowd sourcing location data

This method involves no GPS/GPRS units established in the bus. Most of the people have smart phones today. Smart telephones of human beings touring by using a particular bus can act as supply of region records for that bus. Thus, the data can be 'crowd sourced'.

IV. IOT BACKEND ARCHITECTURE

The facts accrued from a source needs to be sent to servers efficiently and reliably. It must be saved in a way such that customers can question it and have to be on hand even when there is a heavy load on the servers due to large number of people querying it. Fig. 2 indicates the proposed architecture.

The solution involves light weight protocol to send the accumulated data to databases. There are many protocols that can be used to attain this. However, MQTT will satisfactory serve this use- case due to its low bandwidth utilization and aid scalability.MQTT backend can be without problems scaled up by way of clustering of its brokers. Once the server gets the data, it will shop it in NoSQL database like Mongo DB [8], which will be disbursed and have more than one situations to enable load balancing. Users can then question bus areas from these databases.

In MQTT protocol, there is a broker(server) that carries 'topics'. Topics are the way to decide who receives the records generated through a sender. A receiver has to 'subscribe' to a subject matter to get hold of statistics 'published' with that subject matter by means of any sender [9]. The vicinity statistics posted over MQTT from the buses will have the route quantity as topic. Location statistics comprising of the modern latitude and longitude of a bus will be posted at a frequency of 5 seconds and will nonetheless devour less bandwidth due to the fact of the use of MQTT. To make the gadget scalable, the MQTT brokers will be clustered as shown in Fig. three The clustered MQTT brokers act collectively as one broker. Every dealer in the cluster will maintain the matters list. When one dealer is unavailable, every other broker handles the request. This way, the brokers stability inside themselves. Even if one server goes down, other servers will be there in its area to handle the requests and data. The server will acquire the location facts thru MQTT and will save it in a database collection for that unique route number. This can be done with the aid of without delay editing the MQTT broker to write to database when information is obtained or by using creating a MQTT customer to do so.

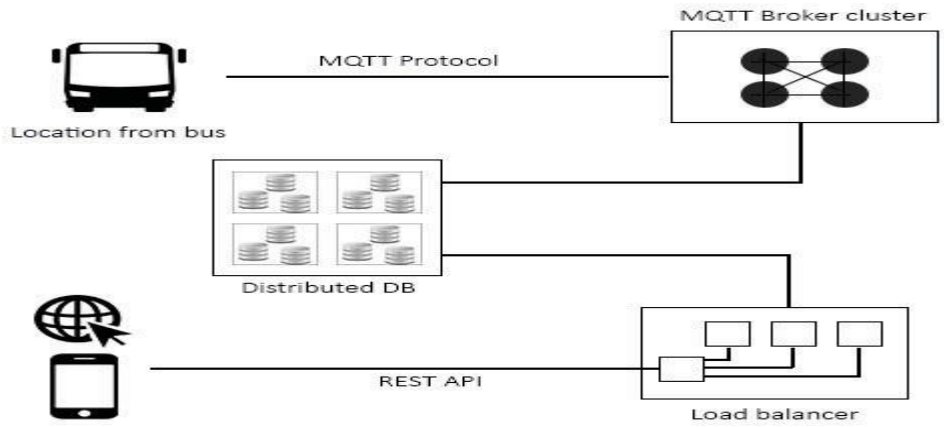

Fig. 2. Architecture Components

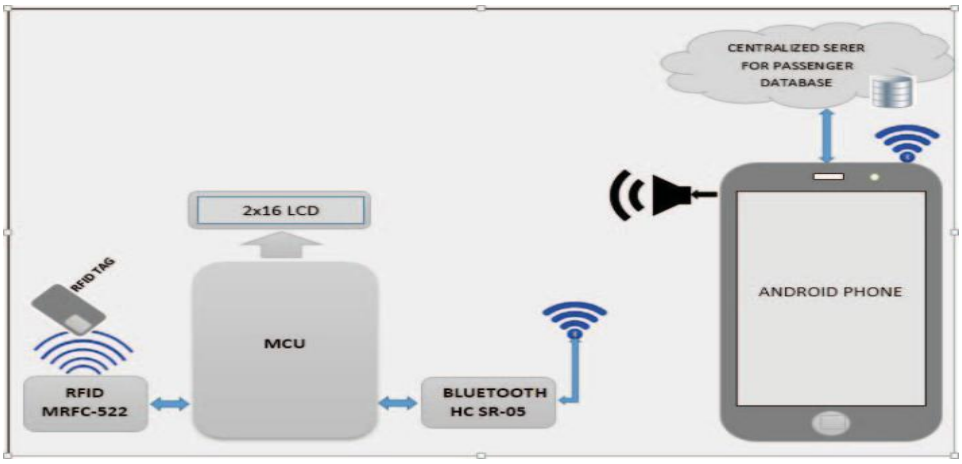

Fig. 3. Block diagram of system architecture

Example document in the database:

"Latitude": 17,

"Longitude": 23, "bus_registration_number": "MH04A1234", "bus_route_number":100;
"direction": 0,

"timestamp": 211005102016

}

Here, direction indicates which way the bus is going for the particular route (A to B or B to A).

Consider a bus with registration range 'MH 12A3023' servicing route '100', the region information obtained from the bus will be saved in NoSQL series named '100' as a central collection 'Buses'. Thus, when a user desires to comprehend what the stay places of buses on route a hundred are, the solution will search in collection named '100'. When the person queries the region of a particular bus, the series named 'Buses' will be searched. Fig. four suggests the collections in which the place data of two buses on route variety a hundred and 200 will be stored.

The e-ticketing machines or android tablets in the bus will post data over MQTT with the route quantity as the topic. In most cases, these machines or capsules stay identical for a specific bus. So they will have the bus registration range pre-fed in them. In case of streaming location information from smartphone of bus riders, there will be an utility where the person can set the route variety and bus registration range to begin streaming place of that bus to server. To cast off the procedure of manually entering the numbers, there can be a NFC tag or Bluetooth beacon in the bus [10]. Scanning the tag will mechanically begin streaming the area facts with right credentials.

them. In case of streaming location data from smartphone of bus riders, there will be an application where the user can set the route number and bus registration number to start streaming location of that bus to server. To remove the process of manually entering the numbers, there can be a NFC tag or Bluetooth beacon in the bus [10]. Scanning the tag will automatically start streaming the location data with correct credentials.

The high number of user queries can be handled by distributing the NoSQL database. For example, NoSQL collection for bus route '100' and bus route '200' will be on different servers 'A' and 'B'

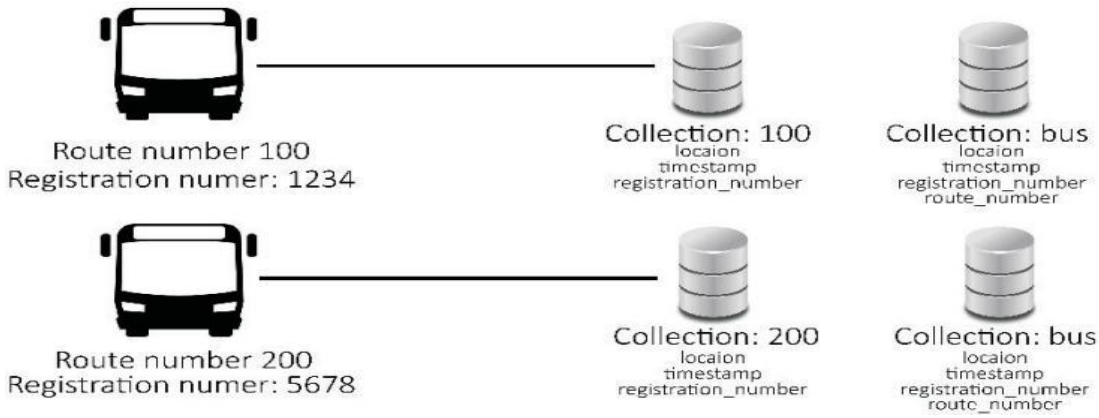

Fig. 4. Database collections

Thus, queries for bus route one hundred will be directed to server A and that for route 200 to server B thereby distributing loads. Also, maintaining cases of equal database on a couple of servers will assist in managing the load. For example, there will be multiple instances of database series A on multiple servers. If there are a high number of requests for route 100, the load balancer will automatically route the requests to the occasion of collection A with minimal load. Consider three servers keeping collection

A. Let S1, S2, S3 be the 3 servers. Load on S1 is 'x', on S2 is 1.5x, on S3 is 1.2x. When a request comes for collection A, the load balancer will routinely route it to the server with least of load, in this cases, S1.

When MQTT writes to the databases, the facts will be modified on all the situations to make certain that all cases have trendy data.

The load distribution will be finished the use of equipment like HAProxy. The requests from a user will come to an intermediate server which will then route it to the splendid server primarily based on the request query.

V. DATA ACCESS POINTS FORUSERS

The users be in a position to get entry to area facts of the buses real time through a variety of structures like web, smartphones or SMS. The proposed answer will make this viable by using ability of REST APIs. The person will ship HTTP requests to the servers which will then fetch required information primarily based on context of the user.

The customers can access the statistics via the user module (web- portal, SMS or smartphone application). This region facts will be proven to the users using Google maps or equivalent map services. The user module will enable the person to search a bus by means of bus registration number directly. However if registration number isn't known, the person module will first ask to select the nearest boarding factor (bus stop) to the user. This method can be automated by using maintaining a database of bus stops along with their corresponding place data. The user's place will be compared with areas of the bus stops and distance will be calculated by means of using the formula of Haversine

La1: User latitude Lo1: User longitude La2: Bus stop latitude
Lo2: Bus stop longitude Dlon: Lo2 – Lo1
Dlat: La2 – La1 R: Radius ofearth

$$A = (\sin(Dlat/2))^2+\cos(La1)*\cos(La2)*(\sin(DLon/2))^2 \quad C = 2 * \tan^{-1}(\sqrt{A},\sqrt{1-A})$$

$$Distance = R*C \qquad (1)$$

Using , the bus end with shortest distancewill be the nearest bus stop.The person module will then ask the bus route number to be selected or the vacation spot bus stop.

Users can then question live vicinity of a bus based totally on the route number, goal destination or bus registration

number. For registration range entered, the user API will return the ultimate regarded location of the bus from the central buses collection.

When the person selects the destination bus stop, the mapping of boarding factor – destination factor to route range can be achieved by using retaining a easy database of all bus stops on all routes, as shown in Table I. So, after deciding on boarding factor and destination, the consumer will get stay place of all buses on routes taking to the destination. The 1st row of Table I indicates the route numbers. The corresponding columns incorporate the listing of bus stops on that route. When the consumer queries data by using boarding and destination bus stops, the information is checked in the table. Consider 'A' and 'B' are the boarding and destination bus stops respectively. The query exams the desk to see if 'A' occurs in the desk and is accompanied by way of 'B'. Bus Stop 'B' need not right away follow 'A'. Routes 100, 200, 300 have such condition true. The question will then fetch the live areas of buses on these three routes and return them to the user.

The queries from the person module will be identical throughout all systems (Smartphone or web). The SMS provider deployment will be by using putting a server to receive SMS queries and sending the effects back as SMS.

The algorithm of the entire system is provided using a flow chart as given in Fig. 4

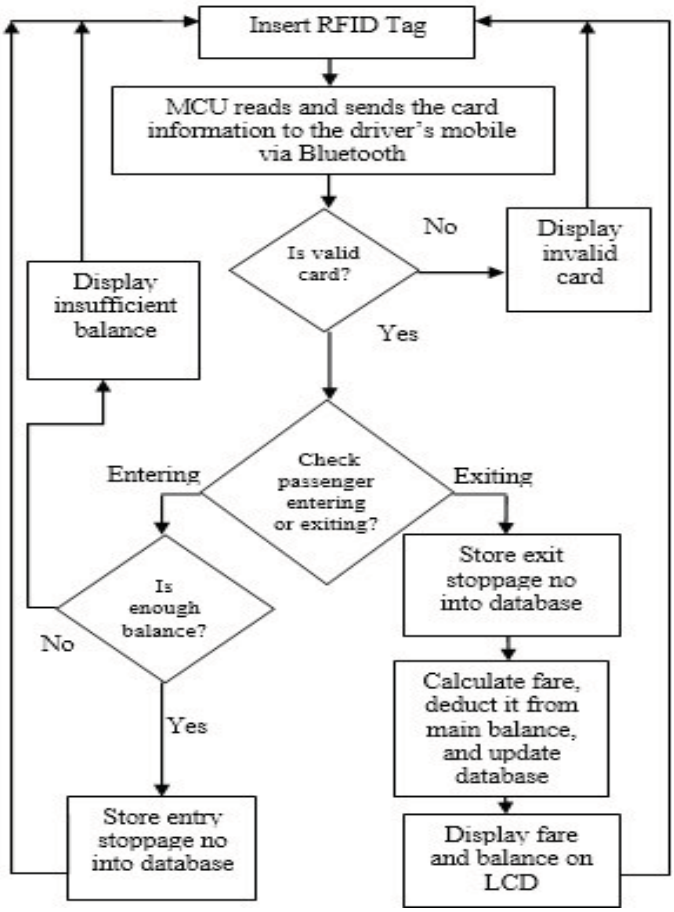

Fig. 5. RFID data format

For tag 'A', passenger Id no. is extracted from obtained data. The next step is to take a look at the card validity primarily based on passenger ID and date of expiry. If it is no longer valid, LCD displays that "Card is now not valid". If it is legitimate then in the next step is to test the balance. If balance is much less than a threshold value, then LCD displays a message showing "Don't have ample balance, please recharge your account". If balance two is more than threshold cost then it stores the bus stoppage no into the database according to the ID. The threshold price is commonly set as the most fare in that route of the bus. In this proposal, we have used SQLite database shown in Table I and II for storing passenger's facts into the android mobile. SQLite and 935–960 MHz for the different path (downlink), supplying a hundred twenty five RF channels (channel numbers 1 to 124) spaced at 200 kHz.

TABLE I. SQLITE DATABASE FOR PASSENGER ENTERING

PASSENGER ID	BALANCE	ENTRY STOPPAGE	EXIT STOPPAGE
12345	*145*	*6*	*0*
12346	*145*	*1*	*0*
12347	*125*	*3*	*0*
12348	*225*	*1*	*0*
12349	*165*	*6*	*0*
12350	*10*	*0*	*0*

8-bit Microcontroller using 4K Bytes Flash

The AT89C51 which act as power of low, high-performance CMOS 8-bit microcomputer with performance of 4K bytes of programmable and easily erasable read solely memory (PEROM). The gadget is manufactured the usage of Atmel's high- density non-volatile reminiscence science and is like minded with the industry-standard MCS-51 training set. The relevant Idle Mode makes to stops the CPU whilst allowing the RAM, timer/counters, serial port and interrupt machine to proceed functioning. Power-down Mode higly saves the RAM contents but freezes the oscillator disabling all different of chip features until the subsequent hardware wil be not done reset.

GPS:

The Global Positioning System is a area age navigational gadget that can pinpoint your role somewhere on the globe. Automobile producers are also offering moving-map shows guided by way of GPS receivers as an option on new vehicles, for use in planning a trip. GPS receiver acquired car role two latitude and longitude from satellite through GPS antenna. GPS receiver is interfaced with the microcontroller through RS232 converter.

GSM:

GSM networks operate in a number of exclusive frequency tiers (separated into 3G1T two SM frequency ranges31T for 2G and 3U1T two MTS frequency bands31Tfor 3G). Most 321T G31TGSM networks function in the 900 MHz or 1800 MHz bands. GSM-900 makes use of 890–915 MHz to send facts from the 3m1T two obile station31Tto the 3b1T ase station31T(uplink)and 935–960 MHz for the other course (downlink), supplying one hundred twenty five RF channels (channel numbers 1 to 124) spaced at 200 kHz.

The HTTP GET requests can additionally be made by developers to get access to the data. The output for the requests will be following format:

A.Query by way of registration number

Will return JSON response fetching data from 'buses'
collection with following parameters:
1.Latitude
2.Longitude
3.Bus route number
4.Bus registration number
5.Time stamp
6.Direction

B. Query by using route number

This question will return a JSON array containing objects. Each object will supply details of active buses going for walks on that route. The data fetched will be from collection with the route number as its name.

TABLE II - BUSSTOPSONEACHROUTE

100	200	300	400	500
A	A	Z	A	Q
B	B	A	S	A
C	H	B	D	C
D	L	G	F	S
E	R	H	G	D

C. Query by boarding point and destination

This query will return a JSON array containing objects for active buses on every route taking from the boarding point to the destination. This data will be fetched from various collections based on which routes are applicable.

VI. PREDICTION OF ARRIVAL TIME AND TIME TO REACH DESTINATION

In order to predict precisely the arrival time of bus at the bus-stops and time to attain destination, it is quintessential to think about actual time traffic. Historical records can be beneficial to predict arrival time . Using the core of artificial neural networks (ANN) on historic information is one way of estimating the required time . However, the ANN would now not consider real time traffic. Hence, it will be fantastic to use Map services like that of Google to furnish estimated arrival time. Google Maps considers real time traffic whilst predicting time for travel. The time for arrival accordingly can be calculated precisely with the aid of giving the live region of bus and region of bus cease to google maps.

To manage the mistaken records received from buses, radial bias characteristic neural networks can be used. These will be in a position to apprehend misguided region records and neglect delete it from the databases in order to keep away from improper information retrieval [14].

VII. RESULTS

The answer proposed will enable users to without difficulty fetch the stay locations of the buses and therefore recognize the estimated time for arrival and also the time to reach the destination. By understanding this time, customers will advantage as they can now graph their travel very accurately. The public bus carrier will turn out to be greater environment friendly and reliable due to availability of real time data. The customers can additionally get admission to this provider from a website or thru SMS.

Fig. 5 and Fig. 6are the screenshots of the android application. Fig. 5 suggests users being capable to get location of bus primarily based on route number. Fig. 6 suggests users getting bus area statistics based on destination bus stop. The utility will mechanically pick out nearest bus give up as boarding point (Shivagingar is selected as boarding point in the above screenshots).

The answer built hasErlang MQTT Broker(eMQTTD) as MQTT broker. eMQTT brokers guide clustering and are scalable, with every single node in the cluster helping up-to 1.3 million concurrent MQTT connections[15]. When there is enlarge in number of users, there is simply a want to add more brokers to this cluster. The clustered nodes in eMQTTD cluster together act as one MQTT dealer and distribute the load within them. eMQTT has modules for connecting to MongoDB. Thus, the answer created is enormously scalable and can without problems preserve increase in load on servers.

VIII. SCREENSHOTS

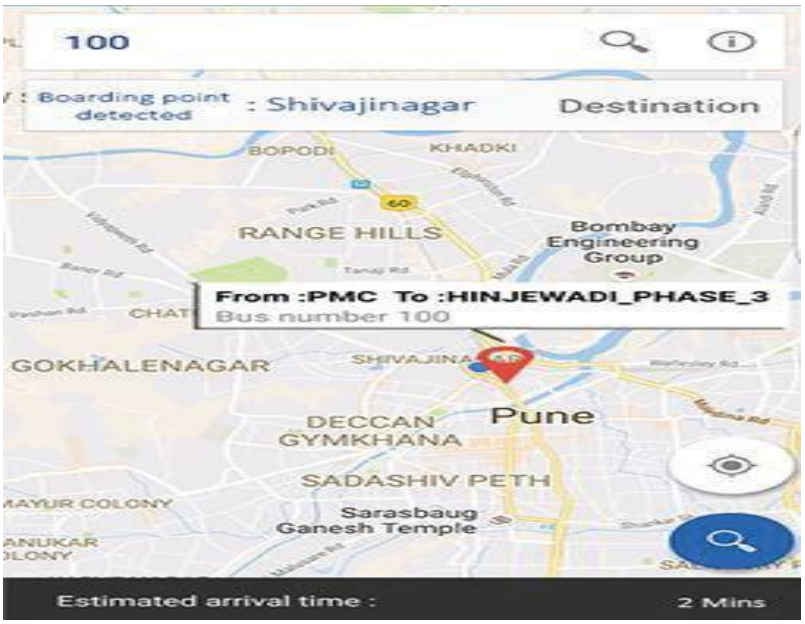

Fig. 5. Search bus by route number

Fig. 6. Search bus by destination

IX. CONCLUSION

We have proposed and applied a new way of tracking public buses via the usage of MQTT as an alternative of standard HTTP. The proposed solution is scalable and future ready. The use of proposed structure approves horizontal scalability for the solution. Use of MQTT will result to much less consumption of bandwidth and on demand get admission to to location records of the buses. Use of MQTT makes the entire solution statistics environment friendly and permits mild weight information transfer. The dispensed backend of increase in range of users makes it scalable. This system will assist in making the public transport structures dependable and efficient.

X. ACKNOWLEDGMENT

We have proposed and utilized a new way of tracking public buses with the aid of the usage of MQTT as an choice of trendy HTTP. The proposed solution is scalable and future ready. The use of proposed structure approves horizontal scalability for the solution. Use of MQTT will result to a good deal much less consumption of bandwidth and on demand get admission to to vicinity information of the buses. Use of MQTT makes the whole answer information environment pleasant and approves moderate weight data transfer. The distributed backend of expand in vary of customers makes it scalable. This system will help in making the public use of transport buildings dependable and efficient.

XI. REFERENCES

[1] Googlemaps,https://developers.google.com/maps/

[2] Chheda Gaurav, Gajra Niket, Chhaya Manal, Deshpande Jitesh, GhargeSaylee, “Real Time Bus monitoring and Passenger Information System”, in“International Journal of Soft Computing and Engineering (IJSCE)”, ISSN:2231-2307, Volume-1, Issue-6, January 201234

[3] Maruthi R.,Jayakumari C., “SMS based Bus Tracking System using OpenSource Technologies”, in International Journal of Computer Applications (0975 – 8887) “, Volume 86 – No 9, January2014

[4] Salim A., Idrees Ibrahim , “Design and Implementation of Web-Based GPS-GPRS Vehicle Tracking System”, in International Journal of Computer Science and Information Technologies , Vol 3, Issue 12, 443- 448, December2013.

[5] Vasileios Karagiannis, Periklis Chatzimisios, Francisco Vazquez-Gallego and Jesus Alonso-Zarate, “A Survey on Application Layer Protocols for the Internet of Things”, Transaction on IoT and Cloud computing,2015