ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 5, May 2015

Cloud Based Public Transport Information System for Android Devices Using Crowd Sourcing

Tejas N A[1], Younus Zain[2], Panimozhi K[3]

UG Student, Dept. of CSE, BMSCE, Bangalore, India[1,2]

Assistant Professor Dept. of CSE, BMSCE, Bangalore, India[3]

ABSTRACT: Daily transport is affected by a number of conditions which are highly uncertain such as congestions, delays, passenger demand and accidents. This calls for a real time and dynamic system that is flexible to meet the needs of the operator and user which is affordable and efficient. An important problem in creating efficient public transport system is obtaining data about passengers' end to end journey. Obtaining this data is problematic and expensive since buses do not have an onboard ticketing system to record where and when passengers get on and off the bus. This paper aims at trouble-free and convenient travel for individuals who commute daily using the public bus transport of a city by making real time information of buses available, without any extra costs right at their fingertips. This is achieved by the concept of Crowdsourcing.

KEYWORDS: GPS, GSM, Cloud, Android, Public Transport, Crowdsourcing.

I. INTRODUCTION

Transportation demands in urban areas continue to increase rapidly as a result of both population growth and changes in travel patterns. This requires planning a system, which is affordable, efficient and reliable from the users' as well as operator's perspectives. This Paper is to make transport much convenient for individuals who commute daily using the public bus transport of the city, for effective time management and making it trouble-free, not just for the commuters but the Transport Department to create an efficient public transport system.

Today in the market there are applications available which specifies the route and the timings, predict arrival times of different buses But this paper builds an application that takes it to the next step by making information about the vacant seats and the current location of any bus in Real-Time, accessible to the daily commuters with a novel and economical wireless system.

II. RELATED WORK

Menon et al [4] propose an app where Internet of Things infrastructure can be used to predict arrival timings of buses as well as the crowd inside each bus. It uses protocols for communication between the devices. The architecture proposed by the research would establish a connection between the bus and its information and the passenger through the means of sensors to calculate the vacant seating capacity of a particular bus, embedded devices are used to collect the temporal information, geographic location and how fast the bus is moving. These details are also sent to the cloud server after every minute through any standard protocols that use 3G/4G, satellite transmits signals to the bus on ground as well as a very accurate time reference which is provided by blocks of atoms, phone app is used to access all the bus information and a cloud server including a database to save and classify the multiple bus information. Finally, they conclude saying bus transportation improves in a number of parameters including management of time, efficiency management, crowd management and in the number of options being offered to users.

Abhishek Dilip Bhonge et al [3] try to make local bus transport easier for everyone by coming up with a mobile application on the android OS. This app has all the information about buses and their respective routes. The user has to enter the source and destination or just enter the destination as the app gets his/her current location and shows all

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 5, May 2015

possible routes to his destination. The app also has an emergency button to send an alert along with their exact location to the authorities or the updated emergency contacts if the traveller feels uneasy while travelling. The buses also have RFID's to track their current capacity which in turn is showed as red or green on the map indicating whether its board able or not.

There have been a variety of location-based services and almost all have failed to gain widespread use or to be exquisitely useful. Thomas Sheppard et al [2] developed this technology which has the capability of listening to the signal your smartphone broadcasts as it searches for Wi-Fi networks, and provides an aggregated and anonymized open data stream showing, accurately, where every person is in real-time. With the software of Presence Orb installed on the ever-increasing number of public Wi-Fi hotspots, you could know exactly how a place will be before you arrive. Presence Orb records only required amount of data in order to provide an aggregate picture of group of people, not individual persons, and that the identification of specific devices would only be with the explicit consent of the owner in an opt-in model. With the presence Orb installed citywide, you could practically know how full a bus would be before it reaches a particular stop.

To tackle the decrease in frequency of estimation of the buses arriving at their right timings, Gunjal Sunil et al [5], proposed an android application which has the ability to obtain accurate prediction of bus arrival time on real time basis which can be viewed by the transport department as well as the user. Three privileged access levels are provided to avoid unauthorized access, namely; Admin, who add/remove buses from the timetable and controls the same for of all buses and, Conductor, who selects the route and starts the app on his phone and therefore the User, who will read the timetable of solely the desired buses and can't modify timetable of buses. The whole system is divided into 3 models; The android application, the website which has the information of the database for customers not having access to an internet-connected android device and a Remote database which holds all of the information in a MYSQL database.

III. PROPOSED SYSTEM

The proposed system completely eliminates hardware using the concept of Crowdsourcing. Crowdsourcing is the process of obtaining needed services, ideas, or content by soliciting contributions from a large group of people, and especially from an online community, rather than from traditional employees or suppliers. [7]

The system is divided into 3 major modules, each module having its own intricate function.

- The android module
- The cloud module
- The QR codes

1. ANDROID MODULE

One of the most widely used mobile operating system used these days is android. Android is a software bunch, comprising not only operating system but also middleware and key applications. Android is a mobile operating system (OS) based on the Linux kernel and currently developed by Google.[6] With a user interface based on direct manipulation, Android is designed primarily for touchscreen mobile devices such as smartphones and tablet computers, with specialized user interfaces for televisions (Android TV), cars (Android Auto), and wrist watches (Android Wear).Android's source code is released by Google under open source licenses, although most Android devices ultimately ship with a combination of open source and proprietary software, including proprietary software developed and licensed by Google. Initially developed by Android, Inc. [7] in this paper we have considered the development of our application on Android. The application is built on Android Studio 1.1, using two main libraries Zxing and Parse. User permissions like coarse location, fine location, and camera are granted to provide basic functionality like access to Google Maps API and Play services. Zxing is an open-source, multi-format 1D/2D barcode image processing library implemented in Java, with ports to other languages [9]. Scanning is performed either by an intent integrator, URL browser or manually. It scans the QR codes on the bus and it also uses the gps of the phone to get the current position of the user. This crowd sourced data is uploaded onto a cloud server making a

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 5, May 2015

whole network of people sharing their location to get real time bus information thus eliminating the need of any hardware. Bus locations are plotted in the form of markers across Google maps, further clicking on a marker reveals an info window with key bus details.

2. CLOUD MODULE

The cloud module used in our system is called "parse" which is essentials a backend as a service (BaaS). BaaS is a module that allows developers to easily link their application to a cloud storage and exposing API's by backend applications while also providing features such as push notifications and social network integration. Parse provides developers to have total control over their cloud databases providing three major functionalities, core, analytics and push. Their cloud engine working with a webpage with a simple UI makes it easy to create different classes of different types such as files, geo points, arrays and so on. Storing data on Parse is built around the "ParseObject". Each ParseObject contains key-value pairs of JSON-compatible data. This data is schema less, which means that you don't need to specify the keys that match each object ahead of time. "ParseQuery" offers different methods to retrieve a list of objects rather than just a single object .The general pattern is to create a ParseQuery, put conditions on it, and then retrieve a List of matching ParseObjects using the findInBackground method with a FindCallback.[8] User and bus information is stored using this software stack and each bus is identified by its unique object ID and registration number. Along with the bus number and a unique registration number for each bus, the capacity and the stop information for all buses are also stored in the database using a JSON Array. Locations are stored as geo points giving a number of inbuilt functions provided by the Parse infrastructure to maintain, update and easily find relationships between each geo location User details are stored within a user class. Parse provides a specialized user class called "ParseUser" that automatically handles much of the functionality required for user account management. ParseUser is a subclass of the ParseObject, and has all the same features, such as flexible schema, automatic persistence, and a key value interface. The database administrator is responsible for handling the classes and making any timetable changes using the simple UI provided by Parse.

3. QR CODES

A QR (Quick Response) code is, as known, to encrypt and store a greater capacity of data compared to usual barcodes. We use a QR code to store the bus number along with the registration number of the bus. These QR codes are then planted inside buses (each having its own unique QR code) for the user to scan and upload information about that bus to the cloud service. These QR codes provide error checking and a security while ensuring that data is uploaded about a particular bus only when the commuter is inside a particular bus. The registration number of each bus being the unique id to refer them, a simple plain text with the bus number and its unique registration number together is used to generate the QR codes using a secure pattern.

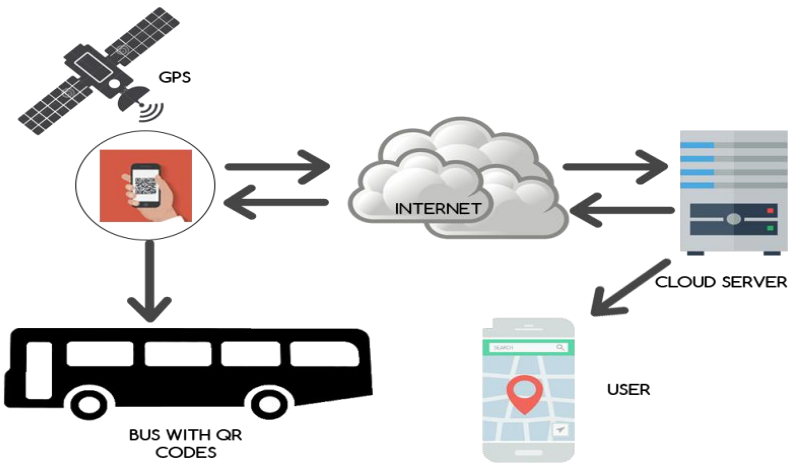

Fig.1 Architecture of the system

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 5, May 2015

The system architecture as shown in Fig. 1, presents how the user can access the Crowdsourced data. By scanning a QR code on the bus and by a tap the user's location is uploaded as the new location of the bus and stored to be accessed by anyone registered to the application. The use case diagram demonstrated in the Fig. 2 provides the different individual functionalities of the system. The "include" relationship of the "log in" use case checks for the new, existing and logged in users. Once the user logs in he can either "search" or "scan" which corresponds to finding a bus near the user or uploading data for a particular bus. QR scanner, Google maps API and the GPS of the phone together accomplishes the above task. Database administrator can manage users and buses through a simple web UI.

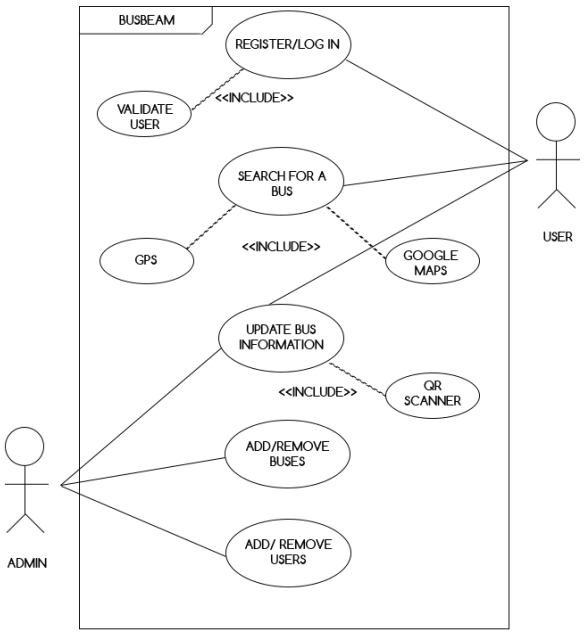

Fig.2 Use case diagram of the system

IV. SYSTEM DESCRIPTION

To elaborate more on the functioning of the system, the modules are explained using a dataflow diagram as shown in Fig. 3. After the user is validated, the main screen is displayed where he has two main options, search and upload, and two options on the action bar, about and logout.

1. UPLOAD

The QR scanner modulestartswhen a user clicks the "scan" option achieved by intents and bundles. On scanning a QR code within a bus, the bus no and the registration no, are verified against the data within the serverensuring that bus exists in the records. The user then has an option to enterthe boarding point and the occupancy of the bus and clicks on "Beam" to upload his location.

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 5, May 2015

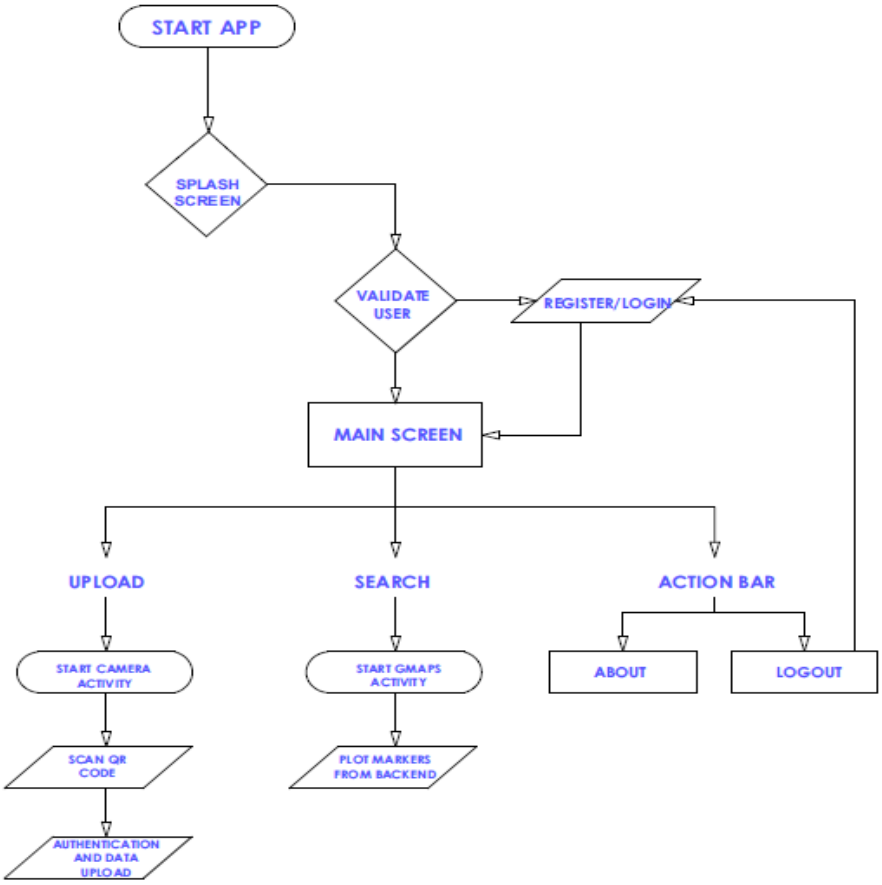

Fig.3 Data flow diagram of the system

2. SEARCH

There is a call to the Google maps API when the search option is selected. The map is sprawled with markers for each bus with their location queried from the server .Tapping on a marker opens an information pop up box that has vital details like the last known bus stop, next stop, occupancy, location of the bus and the time stamp of the uploaded information. Moreover, there is a search box where when a user enters a bus number, the map refreshes to show only those buses on the entire map. Courtesy of the new Google maps API, the user can also find the shortest route to the next bus stop to reach in time to board the bus.

3. ACTIONBAR

The action bar is the topmost bar or border of any application. We have two options here namely, about and logout. The "about" option gives a brief walkthrough on how the application works. The "logout" option helps the user to logout and returns him/her to the login screen.

V. SCREENSHOTS

The successful build of the application that targets the Min SDK of 18 API(ICS) and target SDK of 21 API(Lollilop) was successful installed on a OnePlus one android device and the screenshots of the same are posted below.

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 5, May 2015

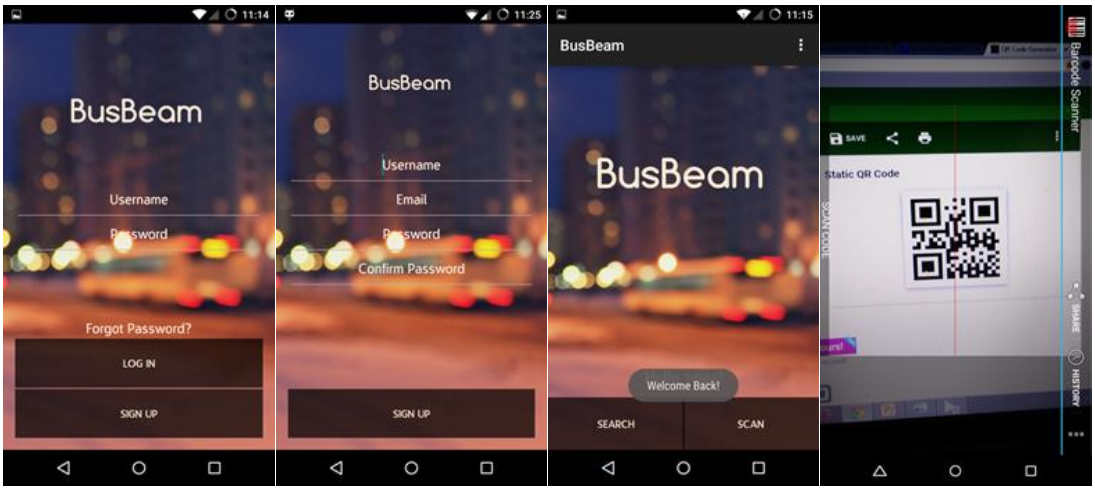

Fig.4 Main screen Fig.5 Login screen Fig.6 Menu Fig.7 QR scanner

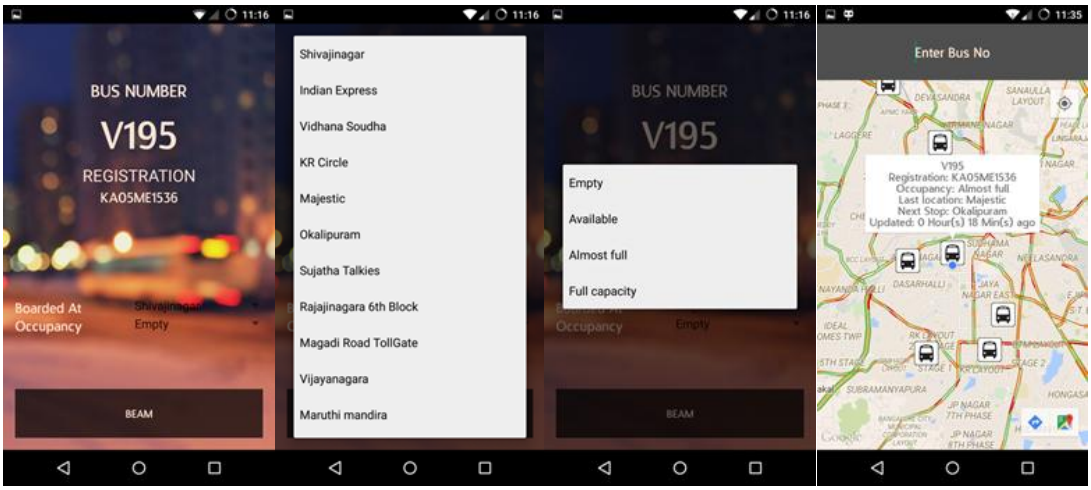

Fig.8 Upload screen Fig.5 Spinner with bus stops Fig.6 Spinner with occupancy Fig.7 Map with buses around

VI. CONCLUSION

With public transport being used by more than 90% of the people within India, it comes down to a single question as to how it can be well built with a solid organized structure to make it feasible to the users and the operators as well. To address these issues we eliminate total hardware dependency with a solution that uses an Android application which solely utilizes crowd sourcing to obtain temporal and geographical bus information and a QR code to perform security checks. Since this dynamic information is available right at the users fingertips it offers a reliable and an inexpensive solution providing answers to a number of problems.

VII. ACKNOWLEDGEMENT

The work reported in this paper is supported by the college through the Technical Education Quality Improvement Programme [TEQIP-II] of the MHRD, Government of India.

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 5, May 2015

REFERENCES

1. Tarek Yahiaoui, Cyril Meurie and Louahdi Khoudour, "Real-time passenger counting in buses using dense stereovision",Electron Imaging, Volume 19, Issue 3, July 2010.
2. Thomas Sheppard and Alan Graham, Wired, Technology (2014, April 22) "Presence Orb" using Wi-Fi to detect if buses are full".
3. Abhishek Dilip Bhonge, Deepak Dattatray Kankhare, Prasad Laxmanrao Takate, "gps enabled android application for bus schedule system", IJRET: International Journal of Research in Engineering and Technology eISSN: 2319-1163 | pISSN: 2321-7308
4. Menon, R. Singha, D. Ediga and Subba Iyer, "Implementation of Internet of Things in bus transport system of Singapore"Asian Journal of Engineering Research, Volume 1, Issue 4,July-Sept.2013.
5. Gunjal Sunil N, Joshi Ajinkya V, Gosavi Swapnil C and Kshirsagar Vyankatesh B, "Dynamic bus timetable using GPS". International Journal of Advanced Research in Computer Engineering & Technology (IJARCET), Volume 3, Issue 3, March 2014.
6. http://developer.android.com/index.html.
7. https://en.wikipedia.org/wiki/
8. https://parse.com/docs/unity/guide
9. https://github.com/zxing/zxing