AUSTRALIAN JOURNAL OF BASIC AND APPLIED SCIENCES

ISSN:1991-8178 EISSN: 2309-8414
Journal home page: www.ajbasweb.com

A Mobile Cloud Application for Bus Tracking System

Vijayalakshmi Muthuswamy, S.Sandhya, S. Preethi and M.C.Ponni

CEG Campus, Anna University, Chennai, India

Address For Correspondence:
Vijayalakshmi Muthuswamy, CEG Campus, Anna University, Chennai, India.
E-mail: vijim@annauniv.edu

ARTICLE INFO

Article history:
Received 04 December 2015
Accepted 22 January 2016
Available online 14 February 2016

Keywords:
Cloud Computing; Location based tracking; GPS

ABSTRACT

This paper focuses on the implementation of a Bus Tracking system, by installing GPS devices on city buses. If passengers had an easy way to see which bus is near to their location and the time it would take to reach the stop, they can make a decision of whether or not to wait at a bus stop. Buswatch is an application designed to display the real-time location(s) of the buses in Chennai city. The GPS data of the bus locations will be transferred to a centralized server where the manipulations happen and the user is provided with a list of possible buses he can travel,along with the timeline diagram and a route map

INTRODUCTION

The Metropolitan Transport Corporation of Chennai has an operating area of 3,929 square kilometers with a fleet strength of 3,798 having 5.184 million passengers each day, generating Rs.30,034,000 with an occupancy ratio of 75.83%. Each MTC bus can carry 72 people, including 24 standing passengers. The occupancy ratio in Chennai is amongst the highest for the 38 transport corporations in the country. Such an important component of our country does not have smart applications to help passengers use this service easily. As users of the MTC bus service, we never know when the bus will come. We tend to miss crowded buses thinking that the next bus will come. But the next bus never comes so early. In such cases, the commuter drops the plan of going by bus and switches to auto instead. Such occurrences may reduce the revenue for MTC. So a better way to find a win-win situation for both the MTC and the public is to develop an application that tells the current location of the buses. This requires the buses to be fitted with a phone with internet connection and our application installed. The application will send the coordinates frequently to our server,thereby we can easily find the location of the bus. Similarly, we also have an application for the public that helps them to not only find which bus to take,but also the current location of the buses and how long they need to wait to catch the bus. This helps the user to take a wise decision on which bus to board to reach their destination on time.

This application is intended to help the general public who require the details of the buses to board, in order to reach their desired destination. The user needs to install this application in their android device. The user's current location will be automatically detected by GPS. The user needs to enter the destination he/she wants to go. This request will be sent to the server. The server has the details of all the buses, their routes and the current locations of all the buses through the GPS installed in each bus. The server filters the buses that can take the user to the destination. From this filtered list, we apply our algorithm to find the best possible list of buses. So the user gets a list of the desired buses that are approaching him along with the time required for the bus to reach the user and the travel time to reach the destination. Based on this list, the user can take a decision on which bus to choose and reach his destination within shortest time. Suppose, if an unusual event occurs, such as stopping by a freight train at a crossing, in such case bus is unable to move, so arrival estimates can't be calculated, it will

To Cite This Article: Vijayalakshmi Muthuswamy, S.Sandhya, S. Preethi and M.C.Ponni., A Mobile Cloud Application for Bus Tracking System. ***Aust. J. Basic & Appl. Sci.,*** *10(2): 110-113, 2016*

switch from counting down the minutes to a "DELAYED" notification. Once the bus begins to move again, arrival estimates will appear again. Integration with Google Maps also helps the user to see the travel route.

Literature Review:

A considerable amount of money is spent on IT-based applications such as real-time, at-stop displays on public transport, but actual knowledge about the behavioral effects these have on customers or potential customers in real life is quite sparse. In the daily operation of a bus system, the movement of vehicles is affected by the uncertain conditions as the day progresses, such as traffic congestion, unexpected delays, and randomness in passenger demand, irregular vehicle dispatching times, and incidents. In a real-time setting, researchers have devoted significant effort to developing flexible control strategies, depending on the specific features of public transport systems. Lin and Zeng (1999) proposed a set of bus arrival time prediction algorithms for a transit traveler information system implemented in Blacksburg, Virginia. Four algorithms were introduced with different assumptions on input data and were shown to outperform several algorithms from the literature. Their algorithms, however, did not consider the effect of traffic congestion and dwell time at bus stations.

EasyTracker (Biagioni, J., 2011) presents an automatic system for lowcost, real-time transit tracking, mapping and arrival time prediction using GPS traces collected by in-vehicle smartphones. Using EasyTracker, a transit agency can implement a sophisticated bus-tracking and arrival time prediction system by simply purchasing a number of smartphones and downloading the bus-tracking app to each phone.

VTrack (Thiagarajan, A., 2009) estimates road travel time based on a sequence of WiFi-based positioning samples using an HMM-based algorithm for map matching. By using hidden Markov based map matching scheme and travel time estimation method, interpolates sparse data to identify the most probable road segments driven by the user and to attribute travel times to those segments.

Systems used in Singapore bus transport system: Singapore bus transport system uses an in built SMS system available for the bus transport using GSM technology from the telecom provider Singtel. This system records the bus arrival timings. Subscriber could also flag bus route where he or she would be alerted with details about the buses through SMS (Ching & Garg, 2002). Katrin Dziekan has also done significant work with rider reactions to real-time arrival information via at-stop displays. In one paper, she summarizes that real-time arrival displays increase feelings of security, reduce uncertainty, increase ease-of-use, adjust travel behavior and improve customer satisfaction. Most importantly to this investigation, permanent real-time arrival signage at transit stations showed that the ability to determine when the next vehicle is coming brings a traveler's perception of wait time in line with the true time spent waiting (Dziekan and Kottenhoff, 2007).

Kidwell [04] presented an algorithm for predicting bus arrival times based on real-time vehicle location. The algorithm worked by dividing each route into zones and recording the time that each bus passed through each zone. Predictions were based on the most recent observation of a bus passing through each zone. However, this algorithm was not suitable for large cities where both travel time and dwell time could be subject to large variations. Generally speaking, these models are reliable only when the traffic pattern in the area of interest is relatively stable. One of their main limitations is that it requires an extensive set of historical data, which may not be available in practice, especially when the traffic pattern varies significantly over time [05]. The difference between this application and Google Maps is quite evident. Suppose you want to go to Anna University from your location, Google maps gives you all possible buses and their frequency along with the travel time. However, it does not give you any information about the current location of the buses and hence it is just a static data. The marked difference between our application and Google Maps is that our application calculates the exact time for the buses to reach us based on the current location of the buses and the travel time to reach your current location.

This app helps people to save their travelling time with optimum choice. It may also increase the number of people using public transport as they know their estimated journey time. It is also likely to bring down the number of people using their own vehicles for commuting which in effect saves fuel. This enables maintaining proper record of buses in the city and its frequency. It also helps in tracking if a bus is held up somewhere due to breakdown as the GPS location will point to the same area for a long time.

System Design:

The features of the system are given in five steps. First, this is a user-friendly application because the user needs to enter the destination alone and hence can be used by novice smart phone users. Next, the user need not enter his current location. It will be automatically detected using GPS. Later, the user enters the destination and in a single button click, it generates all possible bus choices suiting our requirements. Fourth, the server takes care of the operations involving calculation of estimated time to reach the user using the algorithm defined by us. Finally, the following details are displayed as output to the user such as the bus numbers and bus names, time taken for buses to reach the user, time taken for the buses to drop the user in their destination, walking time for the user, effective total journey time ignoring traffic constraints, corresponding travel distance and walking distance, map to show the route taken and timeline diagram to show the flow

Australian Journal of Basic and Applied Sciences, 10(2) Special 2016, Pages: 110-113

System Description:

There are 2 Android applications deployed, namely Bus side Application and User/Client side Application.

Bus side Application:

This is a one screen application that asks the driver to enter the bus name (eg., 23C) and bus number (eg., TN 09 AB 1743). Once the submit button is clicked, the details either create/update the record in the table present in our database. Every 10 seconds, the application sends the coordinates of the bus to the database to know the current location of the bus.

User side Application:

This is a user-friendly application that gives the user with two options. The user can either enter a source or let the application find his source. The user enters the destination where he wants to go and clicks Search. The request hits the server sending the source and destination. The server replies back with the JSON file of the necessary details to be displayed to the user. The user is presented with a list of all buses with the corresponding distances and times. Once a list item is clicked, the user can either view a map or a timeline to show the flow. The map view shows the location of the bus in the map. Clicking on the timeline button aids him in easily understanding his position at various instances of time. This gives him a clear picture that helps him decide to take the bus or not to reach his destination on time.

Server side Bus side Application:

The central server makes use of the PLACES WEB SERVICE to find all the nearby bus stops from the user's location. The algorithm sorts the results from the PLACES WEB SERVICE API to find the closest bus stops. The Google Places API Web Service allows to query for place information on a variety of categories, such as: establishments, prominent points of interest, geographic locations, and more. Search can be done for places either by proximity or a text string. A Place Search returns a list of places along with summary information about each place. The results are fed to the "DIRECTIONS API" from which all the bus names between the end locations are fetched. The Google Directions API is a service that calculates directions between locations using an HTTP request. We can specify source and destination as text strings (e.g. "Guindy") or as latitude/longitude coordinates. Then the respective buses coordinates are fetched from the database, a backend which periodically updates the changes in the coordinates of the buses, and are processed to find their distances from the users. The algorithm sorts the distances and send the corresponding results to the user.

Algorithm:

The algorithm has the following steps. First, places APIs take a geographic location as input and return nearby places. To increase the efficiency of the results from places API, filtration is done on the results fetched and the two bus stops which are close to the geographical location are returned. Next, while calculating the distance with respect to all possible buses, from all the distances that is being displayed the results are sorted using quick sort and shortest distances are extracted discarding other values.

```
TIMELINE ALGORITHM(arrival time,travel time)
 {
U=getUserLatLong();
S=getBustopLatLong();
 B=getBusLatLong();
 Walkingtime=distancecalc(U,S);
 Waitingtime=distancecalc(S,B);
 BusStopReachingTime=ArrivalTime-WaitingTime; DestinationReachingTime=ArrivalTime+TravelTime;
 }
```

Using a gps the buses register their bus name, bus number ,their latitude and longitude to the central server periodically. The user's query to find the bus to a destination hits the central server and the server invokes Places Web services (Places API) to find all the closest bus stops . The results are fed to the Direction API from which all the bus names between the end locations are fetched. Then the corresponding buses current coordinates are fetched from the database. Algorithm sorts the distances and send only those buses and their corresponding distances to the client side which are less than a particular threshold value from the user. Hence the result (all buses with corresponding distance and time) is displayed on the user side mobile application in a list view and route is displayed on a map.

Performance Chart and Timeline Output:

On running the application we found that the time taken for the entire process (hitting the server, calling the APIs, sorting the displaying the results) to complete is less than a minute.

T-nagar to Anna university-47sec

West-mambalam to Adyar-49sec
Guindy to Ashok nagar-45 sec

Architecture Diagram:

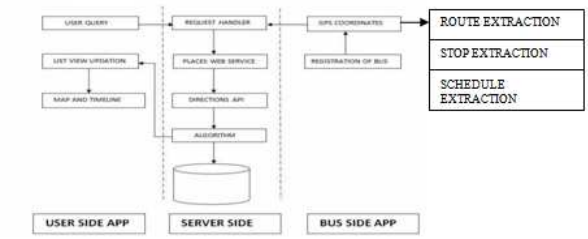
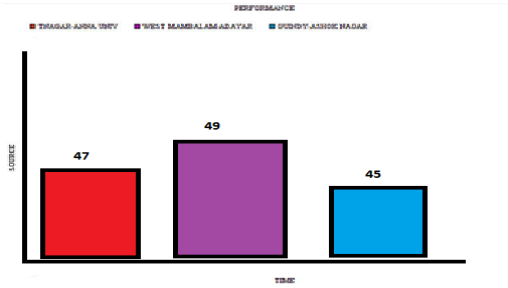
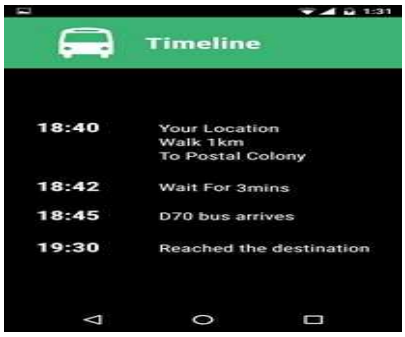


Fig. 1: Framework of bus tracking.





Conclusion And Future Work:

The system tracks every bus around the city using the current location of all the buses and evaluates their arrival time at different stops in their respective routes. System also tracks the distance that needs to be travelled, the time taken to reach the bus stop if the user is located else where and the overall time taken for the entire journey. Finally, it distributes this information to passengers using BusWatch application which is Android based. As discussed this work aids the user in getting his search queries within a stipulated time. The network usage and number of API requests per query are likely to be the overheads of the project. This real time bus tracking app will serve as a viable system that will effectively assist pedestrians in making the decision of whether to wait for the bus or walk and helps them avail the bus services.

REFERENCES

Lin, W.H., J. Zeng, 1999. Experimental study of real-time bus arrival time prediction with GPS data, Transportation Research Record: Journal of the Transportation Research Board, (1666): 101-109.

Biagioni, J., T. Gerlich, T. Merrifield, J. Eriksson, 2011. “Easytracker: Automatic transit tracking, mapping, and arrival time prediction using smartphones,” in Proc. ACM SenSys, 1–14.

Thiagarajan, A., 2009. “VTrack: Accurate, energy-aware road traffic delay estimation using mobile phones,” in Proc. ACM SenSys, 85–98.

Kidwell, B., 2001. Predicting transit vehicle arrival times. GeoGraphics Laboratory, Bridgewater State College, Bridgewater, Mass.

Chandurkar, M.S., S. Mugade, S. Sinha, M. Misal, P. Borekar, 2013. Implementation of Real Time Bus Monitoring and Passenger Information System. International Journal of Scientific and Research Publications, 3(5): 1-5.