Design of automated look-up system for vehicles safe journey

[1]Subhasis Banerjee, [2]Sudip Kumar Chand,
[1]Asst. Professor, [2]Student,
[1]Department of Computer and System Sciences,
[1]Visva-Bharati, Bolpur, India

Abstract:
This paper is based on an automated system which tracks vehicles and updates different information about vehicles and road. It provides wireless services through different mediums. The tracking of the vehicles is handled by the Road Side Unit (RSU). The RSU stores different information in a database. It updates traffic information, about blocked road, accident-prone area, shortest path, and other traffic related information which helps vehicles for a safe journey in an unknown area. RSU acts as agents between vehicles and the system, whose interaction helps a system to track vehicles and help vehicles to get updated information about the area.

***Index Terms* – LAN, WAN, MANET, AODV, DSR, VANET.**

I. INTRODUCTION

A network is defined as a group of two or more computers linked together. Many types of networks are available in nature such as LAN (Local Area Network), WAN (Wide Area Network), CAN (Campus Area Network), MAN (Metropolitan Area Network), HAN (Home Area Network) etc. Networks are also categorized by topology and protocol. Network communication can be done through wired as well as wireless mediums. In wired mediums, different cables are used to communicate from one point to another. We use switch, hub, etc for communication. In wireless medium devices like Wi-Fi, Wi-Max, Satellite and Infrared, Radio wave communications are available.

Ad hoc network is a decentralized wireless network that does not depend on a predefined infrastructure. Different types of ad hoc networks are Mobile Ad hoc Networks, Wireless Mesh Networks, Wireless sensor networks etc. Several routing protocols of MANET are AODV, DSR, DSDV & OLSR etc. Proactive, Reactive Routing and Hybrid are three types of routing protocols (F Cunha, July 2016). The Ad hoc On-Demand Distance Vector (AODV) routing enables dynamic, self-starting, multi-hop routing between participating mobile nodes allowing them free to move, establish and maintain the network. It only takes active nodes during routing and sending packets. One important feature of AODV is its use of a destination sequence number for each route entry. The destination sequence number is created by the destination to be included along with any route information it sends to requesting nodes. If there are two routes of choice to a destination, then a requesting node always selects the route with the greatest sequence number [1].

1.1 Overview

There are several routing techniques in a wireless network that depend on the structure and the algorithm used. Out of all protocols, AODV (Ad-hoc On-Demand Distance Vector) and DSR (Dynamic Source Routing) are important. Some of their techniques are taken care in this paper. AODV protocol creates routes between nodes only if they are requested by source nodes. So, it is called an on-demand algorithm and it does not create any extra traffic. The routes are maintained as long as they are required by the sources. It is used to maintain the sequence number for each node for route freshness. They are self-starting and loop-free besides scaling to many mobile nodes. In this protocol, network remains silent until connections are established. Nodes, who required connections, broadcast a connection request. Other nodes forward that message and store which node requested for the connection, which creates a temporary route for communication. The other nodes on the temporary path send a backward message to requesting node. The node that initiated the request uses the route containing the least number of hops through other nodes. The entries that are not used in routing tables are recycled after some time. If a link fails, the error is passed back to the transmitting node and the whole process is repeated. DSR also works on a similar fashion like AODV. It uses source routing instead of relying on the routing table at each immediate device. Determining the source route needs accumulating the address of every device between source and destination during route discovery. The accumulated path information is stored by nodes in the processing of the route discovery to transfer packets. To apply the source routing technique, the routed packet contains the address of each device the packet will traverse through. It may cause high overhead for long paths or large addresses. To avoid using source routing, DSR optionally defines a flow-id option that allows packets to be forwarded on hop-by-hop basis. In this protocol, all routing information are maintained at the mobile nodes. It has two phases, Route discovery, and Route maintenance. Route reply is generated only when a message has reached the intended destination node. While sending a reply the sender node sends the reply in the reverse path in which it receives the message to the message sender node. In case of any error occurred, the route path maintenance phase is initiated in which the route error packets are generated at a node. The erroneous hop will be removed from the node's route list, all routes containing that hop are truncated at that point. Again, the Route discovery phase is initiated to find the most viable route.

II. VANET

Vehicular ad-hoc networks (VANETs) are created by applying the principles of mobile ad hoc networks (MANETs) and wireless network for vehicle-to-vehicle data exchange. VANETs were first defined and introduced under "car-to-car ad-hoc mobile networking" applications, where networks can be formed and information can be relayed among cars. In VANETs Vehicle-to-vehicle and vehicle-to-roadside communications, architectures can co-exist to provide road safety, navigation, and other roadside services. VANETs are playing a key part of intelligent transportation systems [2].

VANETs support a wide range of applications, simple one-hop information dissemination e.g. cooperative awareness messages (CAMs), and multi-hop dissemination of messages over long distances [3]. Rather than moving at random, vehicles tend to move in an organized fashion in Reality. The interactions with roadside equipment like RSU can be characterized fairly accurately to most of the vehicles which are restricted in their range of motion i.e. they generally follow a pathway. Some of the important applications of VANETs are Electronic brake lights, Platooning, Traffic information systems, Road Transportation Emergency Services, and On-The-Road Services [4].

2.1 Previous Work

2.1.1 Complete multi-hop broadcast

In this one of the starting papers where the complete version of a multi-hop broadcast protocol for vehicular ad hoc network (VANET) was reported. Its result clearly shows that broadcasting in VANET is very different from routing in mobile ad hoc networks (MANET) due to several reasons such as network topology, mobility patterns, demographics, traffic patterns at different times of the day etc. These differences imply that conventional ad hoc routing protocols such as DSR and AODV will not be appropriate in VANETs for most vehicular broadcast applications. It is identified by three very different regimes that a vehicular broadcast protocol needs to work. These categories are dense traffic regime, sparse traffic regime, and regular traffic regime. So, it was built upon previously proposed routing solutions for each regime and it is showed that the broadcast message can be disseminated efficiently. The proposed design of the distributed vehicular broadcast (DV-CAST) protocol integrates the use of various routing solutions that have been previously proposed [5].

2.2.2 Routing problem on disconnected VANET

A vehicular ad hoc wireless network (VANET) exhibits a bipolar behavior in terms of network topology: fully connected topology with high traffic volume and sparsely connected topology when traffic volume is low. In this project, a statistical traffic model was developed based on the data collected to study key performance metrics of interest in disconnected VANETs, such as average re-healing time or the network restoration time. The results showed that, depending on the sparsity of vehicles, the network re-healing time can vary from a few seconds to several minutes which suggests that, a new ad hoc routing protocol will be needed as the conventional ad hoc routing protocols such as dynamic source routing (DSR) and ad hoc on-demand distance vector routing (AODV) will not work with such long re-healing times [6].

2.2.3 Early alert system in the positioning of VANET nodes

In a Vehicular Ad Hoc Network (VANET), before reaching a potentially dangerous zone on road, the wireless Collision Avoidance (CA) system issues warnings to drivers. A global positioning system can be used to get the vehicle position and which can be shared with other vehicles in the network. In case of emergency not all the vehicle gets affected, so broadcasting of alert packet is not feasible rather it has to be multicast. Calculating the list of relative vehicle's positions is depending on the traveling direction, bearing angle, and distance. Recent studies have shown that sparse vehicle traffic leads to network fragmentation which poses a crucial research challenge for safety application. Hence in this paper, a discussion about a system is done which will find out the relative positions between multiple vehicles by using Great Circle Algorithm. The improvement in VANET connectivity is made by a roadside unit which will manage all the vehicle information and detect the failure vehicle and calculate the detail of the vehicles that get affected by the failure vehicle and multicast alert packet to identified vehicles. This will avoid the broadcasting problem [7].

2.2.4 Traffic congestion along routes in VANET

One of the most interesting challenges for the Intelligent Transportation System (ITS) consists of the traffic congestion problem. Congestion is a big obstacle for transportation since it reduces the efficiency of the infrastructure and increases travel time, air pollution, and fuel consumption. Nowadays, the most promising technology in support of ITS is Vehicular Ad Hoc Networks (VANETs). In this paper, three protocols were proposed which can transmit traffic information for routes of interest on VANETs without any Road Side Unit (RSU) support. The protocols make a comparison between a reactive and a proactive approach and present strategies to improve the routing of the packets based on the density and location of the vehicles. The objective is to keep high values of delivery ratio and accuracy using the smallest number of transmissions to guarantee scalability and not to saturate the bandwidth with only this type of packet [8].

III. OUR APPROACH

3.0.1 Objective

In this paper, we have used techniques whose objective is to track all vehicles which are moving within a certain area, along with other details. Vehicles can move in any area, globally they cannot be followed with the help of an ad hoc network, but our approach is limited to within a small region with greater accuracy in storing vehicular information. Our model uses a better update mechanism to manage inter-vehicle communication, Traffic information about routes, Emergency road services, etc. It manages the communication model using a server located in the center of an area. It provides its service through Road Side Units, Wi-Fi, and Wi-Max [9].

3.1 Model architecture

The basic design of the model is shown in the following E-R diagram. The model works with the help of the response of a server. The server communication is handled using HTTP client-server based on communication. It contains databases to store the information of the vehicular nodes within it. Web-based tools such as HTML, PHP, CSS, and JavaScript are used to handle the query and responses of the model. For handling the core database MySQL is used

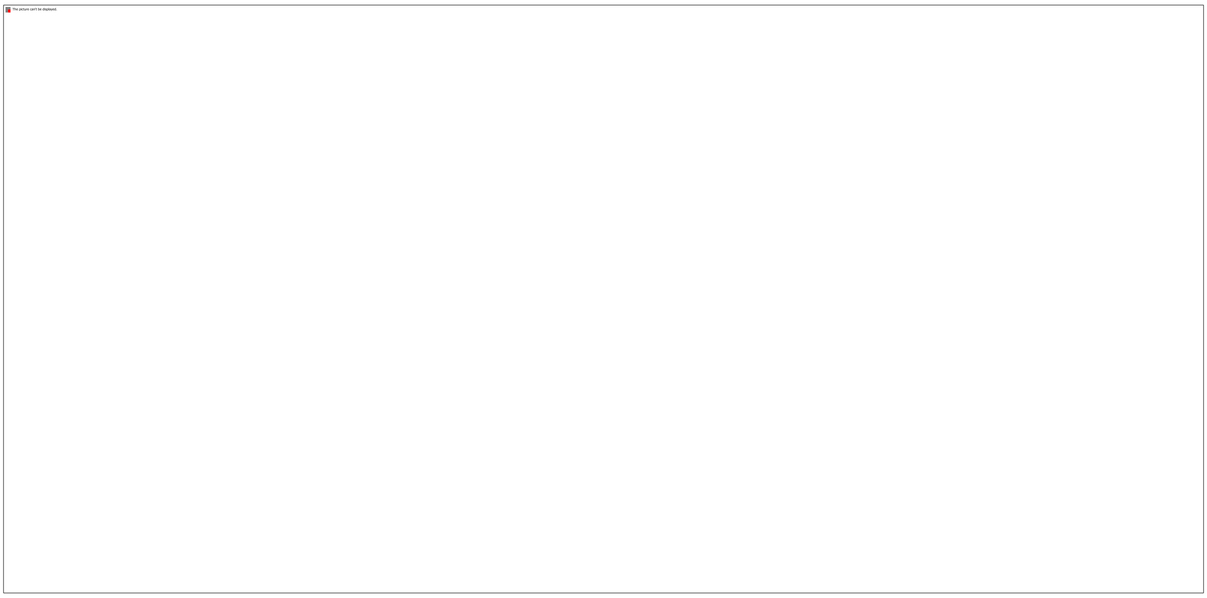


Fig-1

3.1.2 Model elements

This model contains few elements which perform different tasks during communication. The elements are as follows:

Vehicle_db

A database containing all information of vehicles about their location, Car id, Vehicle status, and other information in the form of a table in the form of rows and columns where each row contains details of a particular vehicle.

User_account

A database containing all information related to the user who has access to the vehicle database. It stores the name, email, phone no., username, and password of all users. It also stores the active status of a user. After creating an account by a user to access the vehicle database, if the admin does not allow that user then that user cannot log in to his account. User can access vehicle database only if admin permits to do so.

Add_record

A page that helps users to add records to the vehicle database. It also allows updating a record automatically when the same car id entered to enter a record of which car id is stored in the database previously. It takes location information of a car in the form of latitude and longitude, car id of a vehicle, and store that information along with its movement status.

Delete_record

A page that helps users to delete a record from the vehicle database. If the user found a vehicle has gone out of its network region then he or she will delete its record from the database.

Show_all_Records

A page which shows all record stored in the vehicle database. It shows all vehicle information in the form of a table. An important feature of this page is if any user did notice that a car record is not updated for at least 3 hours which indicates the car is not connected to the server till 3 hours then it will automatically delete that record before showing all records.

Search_Record

A page which allows user to search particular car details by entering its car id. If car details are available in the vehicle database it shows that record, otherwise gives a message that the database does not contain that record.

Profile

A page that shows details of a user. It contains the name, email, phone no., username, password as entered by the user at the time of creating an account. It allows the admin to go into the administrative control zone where the admin can handle the whole model.

Admin_page

A page that contains all details about the model and admin only has permission to change anything. It can perform various operations like adding a vehicle record, updating a vehicle record, deleting a vehicle record, viewing all vehicle records, searching a vehicle record, viewing all user details, update user details, delete user from the User account database, add a user to User account database, Allow/does not allow a user to access vehicle database, etc.

Sign_Up

A page that allows any user to create an account to access the vehicle database. It takes the name, email, phone no., username, and password as input. If another record available in the User account database with the same username then it does not store the record informing user to change the username.

3.1.3 Working of functions

Add_record algorithm:

{

1. Take all data from the user as input (Car id, Latitude, Longitude)
2. Search all records of vehicle database using input Car id
3. If input Car id is available in the vehicle database
   Then check whether the input car location (latitude, longitude) is the same as the database record of that Car id
   If both locations are the same, set car status as not moving and update the record other is set car status as moving and update the record and Alert user as "Record updated"
4. If input Car id is not available in the vehicle database
   Then set car status as not moving and add the record to the database and Alert user as "Record added"

}

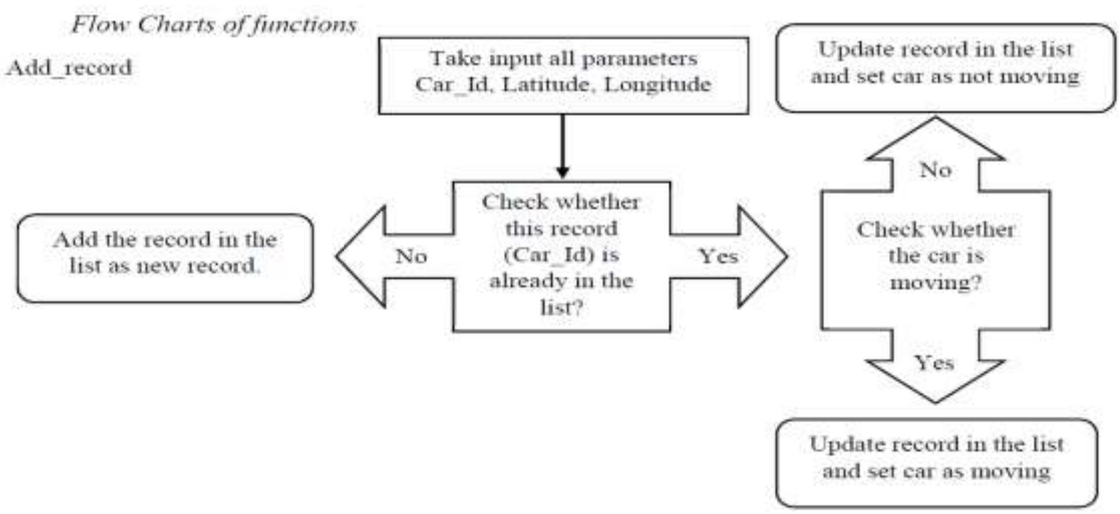


Fig-2

Time Complexity: O(n) where n is the number of records stored in the database.

Reason: In the above algorithm, statement 1 takes O(1) time to take all inputs, Statement 2 takes O(n) times to search a particular record out of n records, Statement 3 takes O(1) time to compare between two records and Statement 4 takes O(1) time to store the record in the database. So overall time is O(n).

Delete_record algorithm:

{
1. Take Car id as input from the user
2. Search vehicle database using that Car id
3. If input Car id is not available in the vehicle database
   Print "Record is not available in the database"
4. Otherwise if the record is available
   Delete that Record having the same Car id as input and
   Alert user as "Record deleted"
}

Flow chart:

Delete_Record

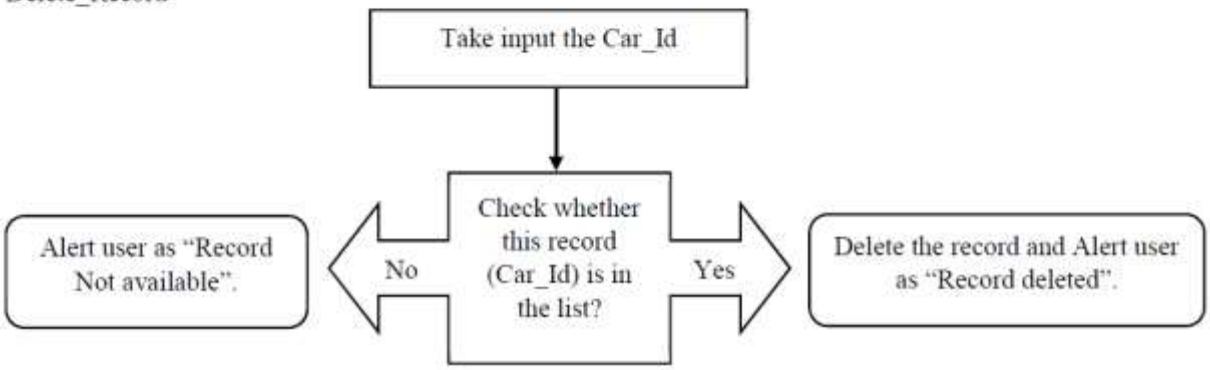


Fig-3

Time Complexity: O(n) where n is the no. of records stored in the database.

Reason: In the above algorithm, statement 1 takes O(1) time to take input Car_id, Statement 2 takes O(n) times to search a particular record out of n records, Statement 3 and Statement 4 takes O(1) time to complete their task. So overall time is O(n).

Show_all_Records Algorithm:

{
1. Pick up all record from the vehicle database
2. If records are available
   Print all records in the form of a table containing Car id, Latitude, Longitude, and Car status.
3. Otherwise if no record available
   Alert user as "No record available"
}

Flow chart:

Show_All_Record

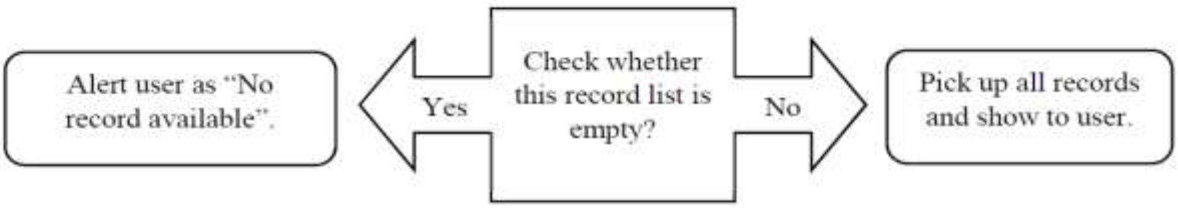


Fig-4

Time Complexity: O(n) where n is the no. of records stored in the database.

Reason: In the above algorithm, statement 1 and Statement 2 takes O(n) times to print all records if available. Statement 3 takes O(1) time to send an alert message. So overall time is O(n).

Search_Record Algorithm:
{
1. Take Car id as input from the user
2. Search vehicle database using that Car id
3. If input Car id is not available in the vehicle database
   Print "Record is not available in the database"
4. Otherwise if a record is available
   Show that Record having same Car id as input
}

Flow chart:

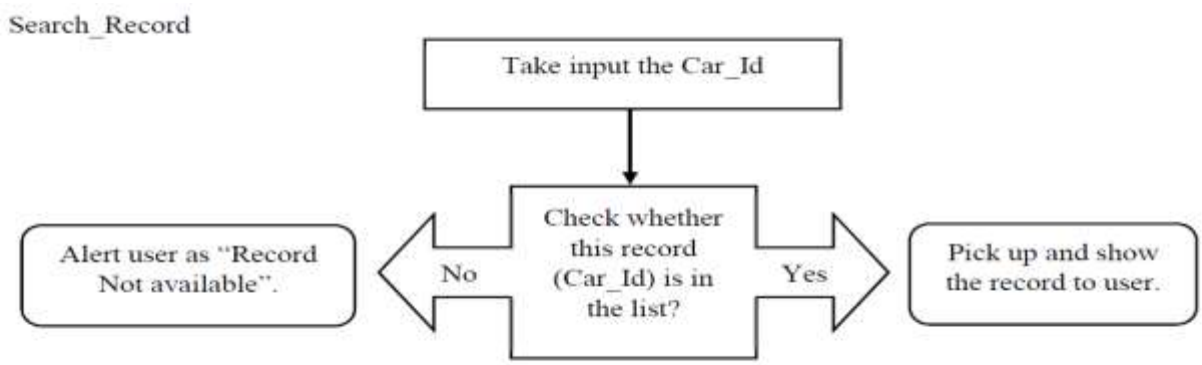


Fig-5

Time Complexity: O(n) where n is the no. of records stored in the database.

Reason: In the above algorithm, statement 1 takes O(1) time to take input Car_id, Statement 2 takes O(n) times to search a particular record out of n records, Statement 3 and Statement 4 takes O(1) time to complete their task. So overall time is O(n).

Sign_Up Algorithm:
{
1. Take all data from the user as input (area_name, position, rsu_id, and password)
2. If password and confirm_passwords are not the same
   Discard the record and alert the user to "Passwords do not match"
3. Search all records of RSU_Account database using input rsu_id
4. If input rsu_id is available in the RSU_Account database
   Discard the input record and Alert the user to "Id already exist, change Id"
5. Otherwise if rsu_id is not available in the RSU_Account database
   Add the record in the database and Alert user "Record added"
}

Flow chart:

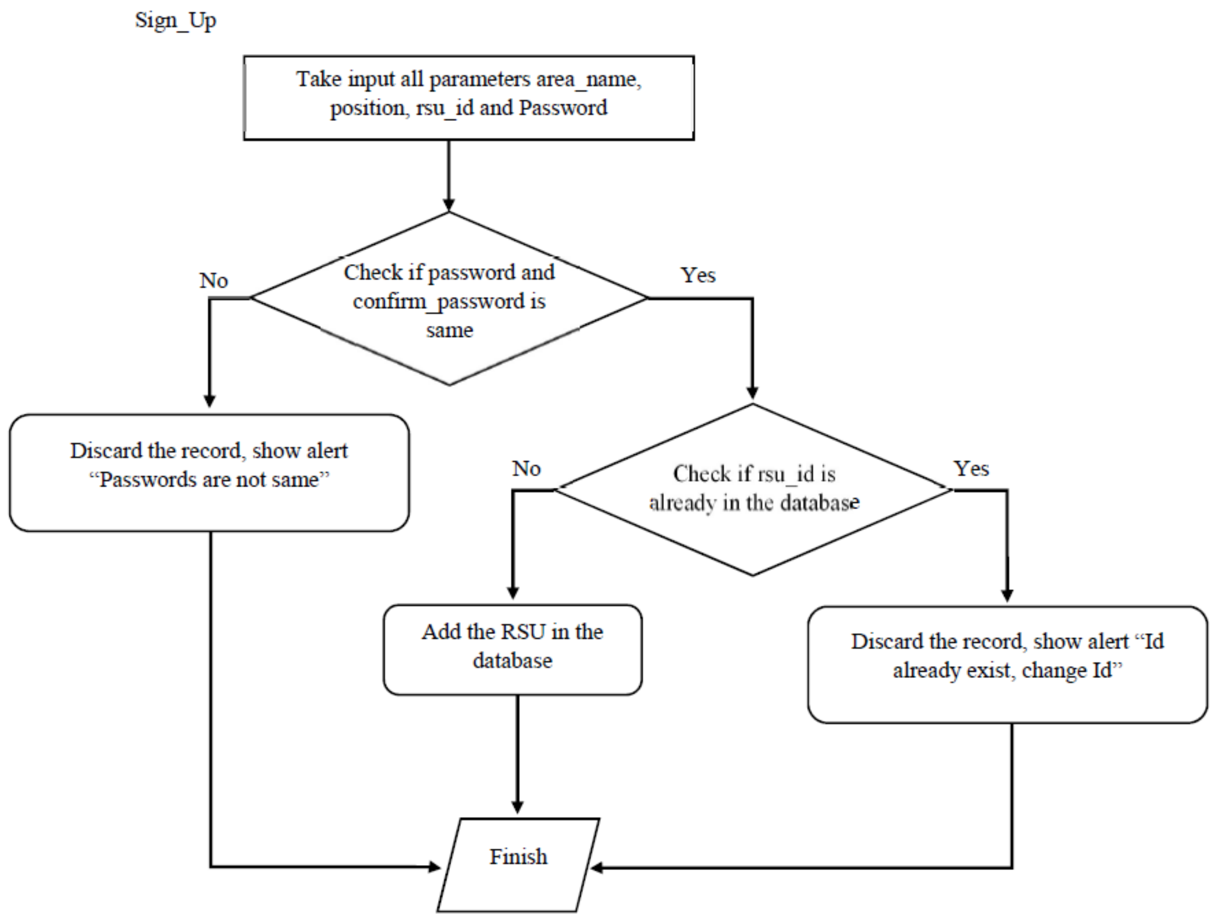
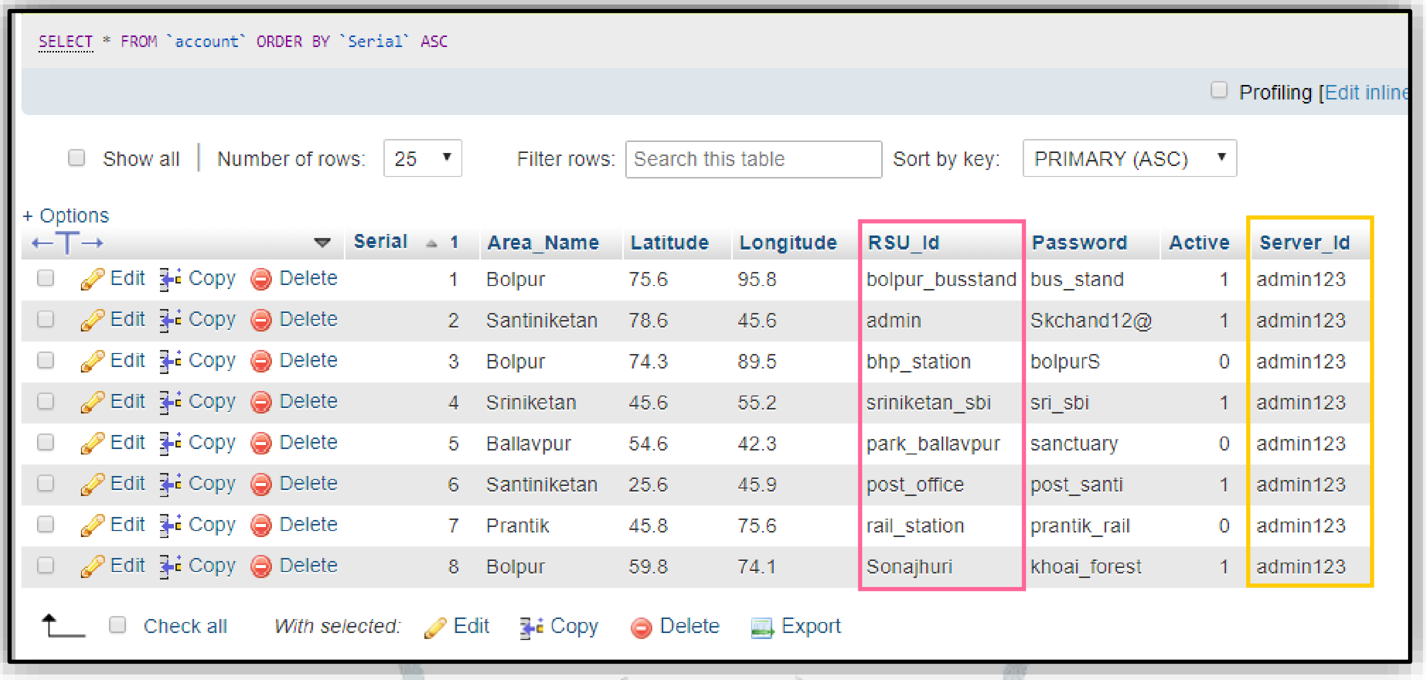


Fig-6

Time Complexity: O(n) where n is the no. of records stored in the RSU database.

Reason: In the above algorithm, statement 1 takes O(1) time to take all inputs, Statement 2 takes O(1) times to compare two strings, Statement 3 takes O(n) time to search a record out of n records, Statement 4 and Statement 5 takes O(1) time to complete their task. So overall time is O(n).

3.1.4 Database Instances of RSU Database


SELECT * FROM `account` ORDER BY `Serial` ASC

	Serial	Area_Name	Latitude	Longitude	RSU_Id	Password	Active	Server_Id
Edit Copy Delete	1	Bolpur	75.6	95.8	bolpur_busstand	bus_stand	1	admin123
Edit Copy Delete	2	Santiniketan	78.6	45.6	admin	Skchand12@	1	admin123
Edit Copy Delete	3	Bolpur	74.3	89.5	bhp_station	bolpurS	0	admin123
Edit Copy Delete	4	Sriniketan	45.6	55.2	sriniketan_sbi	sri_sbi	1	admin123
Edit Copy Delete	5	Ballavpur	54.6	42.3	park_ballavpur	sanctuary	0	admin123
Edit Copy Delete	6	Santiniketan	25.6	45.9	post_office	post_santi	1	admin123
Edit Copy Delete	7	Prantik	45.8	75.6	rail_station	prantik_rail	0	admin123
Edit Copy Delete	8	Bolpur	59.8	74.1	Sonajhuri	khoai_forest	1	admin123

Fig-7

3.2 Advantages of this model

- This model maintains a simple easy to use structure.
- Special care is taken to remove the redundancy of data in both the database.
- It always provides proper alert messages to help a naïve user to interact with it.
- Management of data in databases is done in a proper way such that it does not provide any wrong information and does not modify or remove any data by itself.
- Any user cannot access the database without logging in. Users are only allowed to add, update, track or delete vehicle records from the database.
- More than one user cannot get access from the same browser simultaneously.
- If any user closes the browser knowingly or unknowingly, then his or her account will be automatically logged out.
- A superuser "ADMIN" keeps track of all activities in the model.
- To get access in the database user should create an account and require permission from "ADMIN".
- If "ADMIN" finds anyone wrong then he or she can block that user from accessing the database, if required he or she can remove that user from the user database.
- "ADMIN" is allowed to make any change in the databases.
- If any vehicle goes out of region i.e. it does not interact with the server for a predefined amount of time then the server automatically deletes its record from the database.

IV. CONCLUSION

In VANET, vehicles are mobile nodes, which communicate with a server or with other nodes (vehicles) through RSU. RSU keeps a strong connection with the server to update all information as soon as possible. This model provides many useful services such as traffic optimization, Road transportation works, Emergency services, On road services, route information, shortest paths to an unknown area, accidents or unnatural hazards, etc. It works on the existing position-based routing approach over a vehicle to vehicle infrastructure. In real life for a very busy traffic area, a frequent network disconnection caused by high-speed mobility of vehicles is identified as a major issue. It decreased packet delivery rate, increase packet delay and increase routing overhead in the environment. Successfully building systems from the knowledge of this paper make an intelligent machine to control traffic-based transportation systems. As this model is based on the real-time scenario so, it may have many limitations. For automatic communication with vehicles, some modules must be installed in each vehicle which must not be removable. As this model is based on the wireless network, some problems related to data packet transmissions such as frequency or bandwidth problems can be handled in Congestion control. In vehicle-to-vehicle communication, it can alert a car about the actions of the car in front of it such as whether the car is taking a turn, whether it stops, whether it overtakes, etc. Since the experiments of VANETs are expensive so, several simulators may be used to get a better result instead of hardware modules. In the future, it may provide highly accurate information. This type of system can be used to minimize traffic related problems which are available today in our day-to-day life. Till now this paper does not includes map, but for better optimization it must requires a high-quality map with all important traffic points marks in it such as where traffic signal exist, about joining of roads, nearest buildings like hospitals of school, traffic checking points etc. which will help the system to better manage the traffic situations and help common public travelling through vehicles in a better way.

V. ACKNOWLEDGMENT

We would like to express our sincere gratitude to Dr. Utpal Roy, Professor, Dept. of Computer and System Sciences, Visva-Bharati for providing his valuable guidance, comments and suggestions throughout the working session of this paper. He supported and encouraged us a lot for working on this paper. We offer a heartfelt thanks to him for providing his valuable time to us.

REFERENCES

[1] Singh, Meeta and Jigyasa Sharma. "Performance analysis of secure & efficient AODV (SE-AODV) with AODV routing protocol using NS2." *Proceedings of 3rd International Conference on Reliability, Infocom Technologies and Optimization* (2014): 1-6.

[2] Sivasakthi M., Suresh S., Research on Vehicular ad hoc Networks (VANETs): An overview Journal of Applied Science and Engineering Research , Journal of Wireless Networking and Communications, Published – 2013: 29-38

[3] F. Li and Y. Wang, "Routing in vehicular ad hoc networks: A survey," in *IEEE Vehicular Technology Magazine*, vol. 2, no. 2, June 2007, pp. 12-22

[4] Moustafa, H. and Zhang, Y. (2009) Vehicular Networks: Techniques, Standards, and Applications. CRC Press, Boca Raton. http://dx.doi.org/10.1201/9781420085723

[5] N Wisitpongphan, OK Tonguz, JS Parikh, P Mudalige, F Bai, V Sadekar, "Broadcast storm mitigation techniques in vehicular ad hoc networks", IEEE Wireless Communications 14 (6), 84-94

[6] N. Wisitpongphan, O. Tonguz, F. Bai, P. Mudalige, V. Sadekar, "On the Routing Problem Disconnected Vehicular Ad Hoc Networks", Proceedings of the IEEE INFOCOM 2007 - 26th IEEE International Conference on Computer Communications, May 2007 Pages 2291–2295

[7] Y. C. Raut, V. N. Katkar, S. M. Sarode, "Early alert system using relative positioning in Vehicular Ad-hoc Network", Emerging Research Areas: Magnetics Machines and Drives (AICERA/iCMMD) 2014 Annual International Conference on, pp. 1-8.

[8] Giuseppe Martuscelli, Azzedine Boukerche, Paolo Bellavista, "Discovering traffic congestion along routes of interest using VANETs", Global Communications Conference (GLOBECOM) 2013 IEEE, pp. 528-533, 2013.

[9] H. Hartenstein and L. P. Laberteaux, "A tutorial survey on vehicular ad hoc networks," in *IEEE Communications Magazine*, vol. 46, no. 6, June 2008, pp. 164-171.