# *Smart Vehicle Analyzing System with service center alliance using mobile computing (VASS).*

**Prof. V. R. Ghule[1], Hrushikesh Gujar[2], Shubham Gaikwad[3], Parmeshwar Bhande[4]**

[1]*Assistant Professor, STES Smt. Kashibai Navale college of Engineering Pune, Maharashtra, India.*

[2,3,4]*Computer Engineering student, STES Smt. Kashibai Navale college of Engineering Pune, Maharashtra, India.*

---

**Abstract-** *This application is an android app which can be run on any android compatible tablets and mobile phones. The main purpose of this project is to provide ease and convenience to the clients for the servicing of their vehicles. According to current location of the customer whom need to service his/her vehicle at service center will be suggested location of best service center near to the customer current location.*

*For security, 4D security will be provided on the parameters of name, contact number, photo and unique id of the person who will take the bike from the client. Once the client request for the service to the specified service center, service man accepts the request and will take the bike from his home. On spot service also will be provided if some accident happened and damaged to the vehicle. This provides convenience to the clients to service their bikes/cars.*

*Now a days multiple service center are available under specific brand but they are not connected and also local garages are not online. So this project helps to connect them.*

***Key Words:*** **Two-tier architecture, GPS, SQLite database, Mutual Authentication, Push Messages, 4D security, Android.**

## 1. INTODUCTION:

There are various service sites & many brand bikes are available which provide the service to vehicle. They provide the service to only the bikes of their brand & also there is no local garage concept. When customer want the service of vehicle at any time & any place then such service site not able to provide service. It takes lot of time to provide vehicle service. If customer go out of his area and in that area he want service to his vehicle. When we take example of this sites then it is not possible to provide the service at that time & at that place. The system of vehicle repair service is a problem that needs to research on the location of repair centers according to the service needs and available resource based on the maximal covering location and priority queuing theory. Considering the effect of waiting time due to rush jobs, some system models are there that maximizes the service covering, and restrains the service level of uncovered zones. But such a system this system also not provide the service as early as possible.

Some site provide the vehicle service but that site does not cover total area. It provide service for limited area and that's why people facing problem. When customer want vehicle service at any time & at any place there are not any site. So people face lots of problem. There is need of service site which provide the vehicle service easily without wasting lot of time and at any place. When customer face the vehicle problem then there is need of app or website which provide the garage information that are nearest. According to current location of customer whom need to service his/her vehicle at service center that application suggested location of best service center near to the customer's current location. So that customer does not face the problem of vehicle service.

To develop an android business application which provide services to those customers requiring servicing of their vehicles. Our application will connect the customers to the local garages with ease, providing security and authorization at the customer end. This will facilitate customers by giving them the benefit of good servicing by trustworthy service centers in the vicinity of the customer. The main purpose of this project is to provide ease and convenience to the clients for the servicing of their vehicles. According to current location of the customer whom need to service his/her vehicle at service center will be suggested location of best service center near to the customer's current location. Online payment or cash payment will be accepted after servicing your vehicle. For security, 4D security will be provided on the parameters of name, contact number, photo and unique id of the person who will take the bike from the client.

## 2. TECHNICAL STUDIES:

### 2.1. Android:

Android is a mobile operating system developed by Google, based on the Linux kernel and designed primarily for touchscreen mobile devices such as smartphones and tablets. Android's user interface is mainly based on direct manipulation, using touch gestures that loosely correspond to real-world actions, such as swiping, tapping and pinching, to manipulate on-screen objects, along with a virtual keyboard for text input. In addition to touchscreen devices, Google has further developed Android TV for televisions, Android Auto for cars, and Android Wear for wrist watches, each with a specialized user interface. Variants of Android are also used on notebooks, game consoles, digital cameras, and other electronics.

Now many versions of android available up to marshmallow. Now a day's android is trending operating system for mobile phones. Here we use android 6.0 and API level 21.

### 2.2. SQLite database:

SQLite is a relational database management system contained in a C programming library. In contrast to many other database management systems, SQLite is not a client–server database engine. Rather, it is embedded into the end program. SQLite is ACID-compliant and implements most of the SQL standard, using a dynamically and weakly typed SQL syntax that does not guarantee the domain integrity.

SQLite is a popular choice as embedded database software for local/client storage in application software such as web browsers. It is arguably the most widely deployed database engine, as it is used today by several widespread browsers, operating systems, and embedded systems (such as mobile phones), among others. SQLite has bindings to many programming languages.

### 2.3. Google map:

Google maps is a desktop web mapping service developed by Google. It offers satellite imagery, street maps, 360° panoramic views of streets (Street View), real-time traffic conditions, and route planning for traveling by foot, car, bicycle, or public transportation. We use google map to get the longitude and latitude values of customer and garage/service center location. The google map also helps to get neighbor's node locations.

After getting longitude and latitude locations of customer and service centers we will use Haversine formula to calculate distance between service centers and customer.

### 3. SYSTEM ARCHITECTURE:

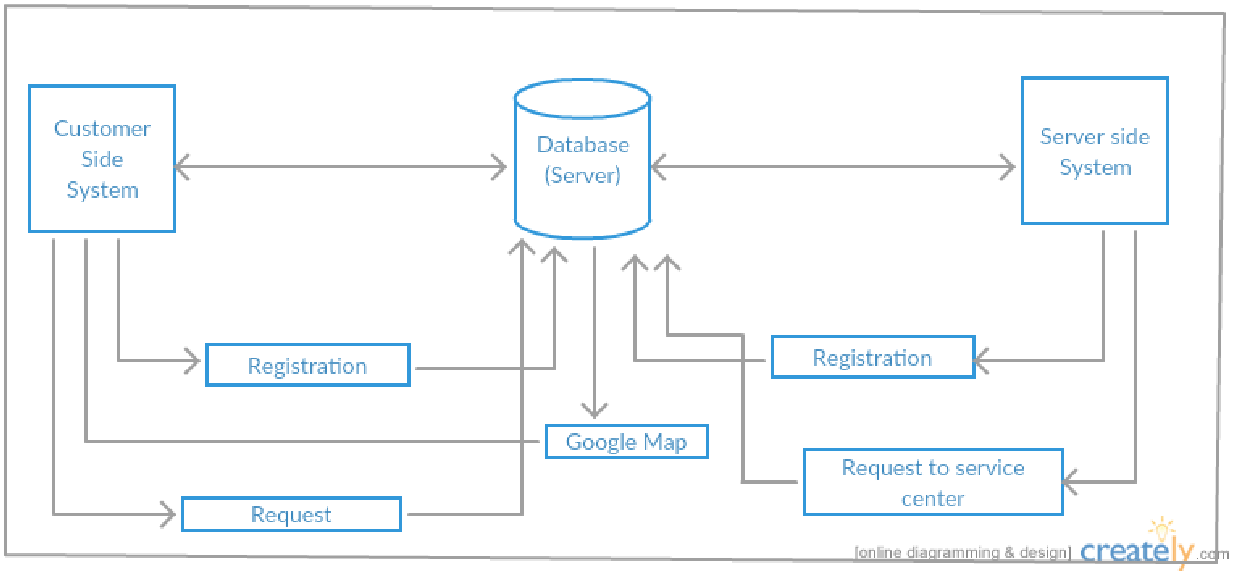


**Fig. -1:** System Architecture diagram

3.1. Customer side App:

Android app at customer side is installed and registration of customer who need to service his bike/car is registered. While registering name, contact number, gender, bike type and current location is registered. All the service centers available near within 5 km from his current location will be shown on the map.

According to ratings of the service center customer can choose any of the service center and request to that service center is registered and same will be shown at shop owner side. All the details of servicing of bike and bill payment is shown on the app. Feedback and ratings to the service provider for the servicing experience can be given by the user.

**3.2.** Shop owner side App:

Android app at shop owner side is installed and registration of shop who will provide service to the customer for servicing of bike is registered. While registering owner name, shop name, contact number, servicing type and location is registered. Request from the customer side is shown on the app and can be accepted by shop owner. Request approval message delivered at the customer side. Bill payment details and customer information as well as his bike information is shown at shop owner side.

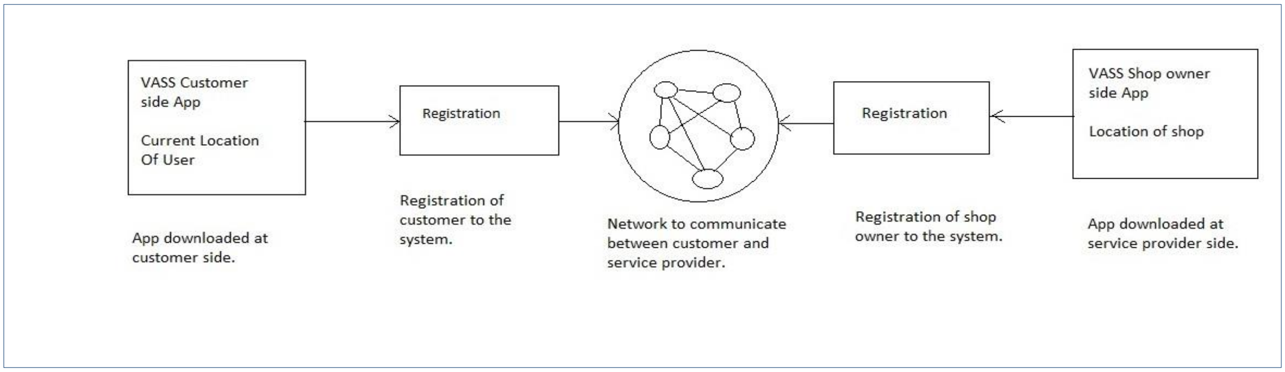


**Fig. -2:** Customer side and Service center side app

### 3.2. Intermediate Network:

Network is created among all the users and shop owners. So that they can communicate between each other for making request for bike servicing and accepting request from shop owner side for the same. All the shops nearby customer location is shown by algorithmic approach of Warshall's algorithm. The algorithm is design such that it will calculate best possible shortest distance from current location of customer provide the appropriate shops nearby. The network is three tier architecture between customer, service provider and server. The server is to be best suitable for handling all the request from customer and shop owner side and will provide best faster service. The database for the system is always provide from the server to communicate with customer and shop owners.

All the details of customers, Shops, servicing of bikes, bill payments will be always reflected to database which live on three tier network.

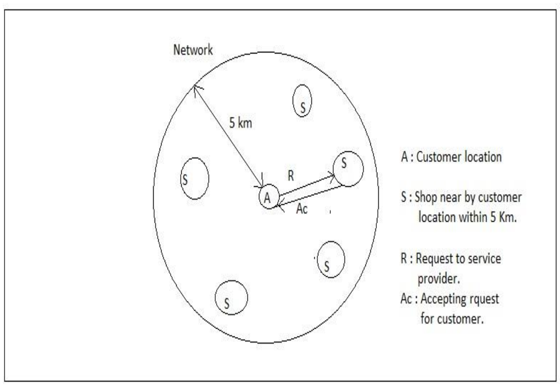


**Fig. -3:** Map structure.

In computer science, the Floyd Warshall algorithm is an algorithm for finding shortest paths in a weighted graph with positive or negative edge weights (but with no negative cycles)

### 4. MATHEMATICAL MODEL:

A mathematical model is a description of a system using mathematical concepts and language.
The process of developing a mathematical model is termed mathematical modeling.
Mathematical model consist of three parts:

1. Mapping
2. State Diagram
3. Set theory

In proposed project there 'N' number of users and system. Therefore our mapping is many to many. The users are represented as {U1, U2, U3…..Un }. And our system represented as {S1, S2, S3 …Sn}.

User set: {U1, U2, U3 …..Un}

System set: {S1,S2,S3…Sn}.

The 'Haversine' formula is used to calculate the distance between 2 points.

Haversine formula:

$a = \sin^2(\Delta\varphi/2) + \cos(\varphi 1)*\cos(\varphi 2)*\sin^2(\Delta\lambda/2)$

$c = 2*a*\tan2(\sqrt{a}\sqrt{(1-a)})$

$d = R*c$

Where

$\varphi$ is latitude

$\lambda$ is longitude

R is earth's radius (mean radius = 6,371km)

Note: Angles are in radians

## 5. ALGORITHM:

Floyd's Algorithm

Stephen Warshall and Robert Floyd independently discovered Floyd's algorithm in 1962. In addition,
Bernard Roy discovered this algorithm in 1959. This algorithm is sometimes referred to as the Warshall-Floyd algorithm or the Roy-Floyd algorithm. The algorithm solves a type of problem call the all-pairs shortest-path problem, meaning that it finds the shortest path between all the vertices of a given graph. Actually, the Warshall version of the algorithm finds the transitive closure of a graph but it does not use weights when finding a path. The Floyd algorithm is essentially the same as the Warshall algorithm except it adds weight to the distance calculation.

This algorithm works by estimating the shortest path between two vertices and further improving that estimate until it is optimum. Consider a graph G, with Vertices V numbered 1 to n. The algorithm first finds the shortest path from i to j, using only vertices 1 to k, where k<=n. Next, using the previous result the algorithm finds the shortest path from i to j, using vertices 1 to k+1. We continue using this method until k=n, at which time we have the shortest path between all vertices. This algorithm has a time complexity of O(n3), where n is the number of vertices in the graph. This is noteworthy because we must test up to n2 edge combinations.

The Floyd–Warshall algorithm compares all possible paths through the graph between each pair of vertices. It is able to do this with comparisons in a graph. This is remarkable considering that there may be up to edges in the graph, and every combination of edges is tested. It does so by incrementally improving an estimate on the shortest path between two vertices, until the estimate is optimal.

The pseudo code foe warshall:

```
1 let dist be a |V| × |V| array of minimum distances initialized to ∞ (infinity)
2 for each vertex v
3    dist[v][v] ← 0
4 for each edge (u,v)
5    dist[u][v] ← w(u,v)  // the weight of the edge (u,v)
6 for k from 1 to |V|
```

```
for i from 1 to |V|
  for j from 1 to |V|
    if dist[i][j] > dist[i][k] + dist[k][j]
        dist[i][j] ← dist[i][k] + dist[k][j]
     end if
```

The one catch here is that this assumes that you have Dijkstra's algorithm backed by a Fibonacci heap. If you don't have Fibonacci heap available and aren't willing to put in the 72 hours necessary to build, debug, and test one, then you can still use a binary heap for Dijkstra's algorithm; it just increases the runtime to O(m log n), so this version of Johnson's algorithm runs in O(mn log n). This is no longer always asymptotically faster than Floyd-Warshall, because if $m = \Omega(n^2)$ then Floyd-Warshall runs in $O(n^3)$ while Johnson's algorithm runs in $O(n^3 \log n)$. However, for sparse graphs, where $m = o(n^2 / \log n)$, this implementation of Johnson's algorithm is still asymptotically better than Floyd-Warshall

In short:

With a Fibonacci heap, Johnson's algorithm is always asymptotically at least as good as Floyd-Warshall, though it's harder to code up. With a binary heap, Johnson's algorithm is usually asymptotically at least as good as Floyd-Warshall, but is not a good option when dealing with large, dense graphs. In computer science, the Floyd Warshall algorithm is an algorithm for finding shortest paths in a weighted graph with positive or negative edge weights (but with no negative cycles)

A single execution of the algorithm will find the lengths (summed weights) of the shortest paths between all pairs of vertices, though it does not return details of the paths themselves. Versions of the algorithm can also be used for finding the transitive closure of a relation.

Prior to the first iteration of the outer loop, labeled k=0 above, the only known paths correspond to the single edges in the graph. At k=1, paths that go through the vertex 1 are found: in particular, the path [2,1,3] is found, replacing the path [2,3] which has fewer edges but is longer (in terms of weight). At k=2, paths going through the vertices {1,2} are found. The red and blue boxes show how the path [4,2,1,3] is assembled from the two known paths [4,2] and [2,1,3] encountered in previous iterations, with 2 in the intersection. The path [4,2,3] is not considered, because [2,1,3] is the shortest path encountered so far from 2 to 3. At k=3, paths going through the vertices {1,2,3} are found. Finally, at k=4, all shortest paths are found.

### 6. FEASIBILITY STUDY:

Feasibility Analysis is the process of determination of whether or not a project is worth doing. Feasibility studies are undertaken within tight time constraints and normally culminate in a written and oral feasibility report. It helped in taking decisions such as which software to use etc.

[1] Technical Feasibility

[2] Economical Feasibility

[3]Operational Feasibility

6.1. Technical Feasibility:

The technology which we are going to use is android SQLite. It is widely used and have proven to be reliable. It is also very efficient and easy to use.

6.2. Economic Feasibility:
In our project we use SQLite database which is locally and freely available database server. Our project is based on Open source tools, so there are very few factors that would affect the economic feasibility.

6.3. Time Feasibility:
This project will take 6 to 7 months for implementation.
6.4. Privacy feasibility:
For security, 4D security will be provided on the parameters of name, contact number, photo and unique id of the person who will take the bike from the client. As we use the authentication of customer/service provider, the acquired information is confidential and thus it eliminate the problems related to privacy.

## 7. SCOPE:

The main purpose of this project is to provide ease and convenience to the clients for the servicing of their vehicles. For the practical implementation of the idea we are developing an android application i.e. Garage Connector. Basically first user need to download the application from play store, using this application the current location of user is displayed on the screen according to the altitude and latitude & also the nearby garages in the range of 5Km from the location of client are displayed. Now the client will choose the garage according to the reviews, rating & his convenience. Then the client will fill the details on app for the sake of registration. Then one message will send to the garage person which includes the details of client, his 2 wheeler & simultaneously client will also get message which includes the details of garage person.

## 8. CONCLUSION:

The proposed paper shows the flow, structure and working of the VASS system. VASS is user friendly i.e. easy to use. It is free of cost on android store. Thus, it is time a time saving as well as cost efficient application. So, we can conclude that the proposed system can be used to reduce human efforts and luxuriate human lives, hand in hand, with the modern technology.

**ACKNOWLEDGEMENT:**

We thank our colleagues who provided insight and expertise that greatly assisted the research, although they may not agree with all of the interpretations/conclusions of this paper.

We are grateful to Prof V. R. Ghule for assistance and moderated this paper and in that line improved the manuscript significantly.

## REFERENCES:

[1] An Emergency Service Center Location Model for Vehicle Repair with Priority
Queuing Rules and Service Level Constrains. Li-si CAO1, Zi-xian LIU1 1School of Management and Economics, Tianjin University ,Tianjin, China
[2] Automatic Construction of Garage Maps for Future Vehicle Navigation Service.

Qian Zhouâˆ—, Fan Yeâˆ—, Xiaoge Wang , Yuanyuan Yang —Department of Electrical and Computer Engineering, Stony Brook University, Stony Brook, NY 11794, USA
{qian.zhou,fan.ye,yuanyuan.yang}@stonybrook.edu Institute for Cyber-Enabled Research, Michigan State University, East Lansing, MI 48824, USA.

[3] Automobile Service Center Management System. Prof. Shilpa Chavan Saket Adhav, Rushikesh Gujar, Mayur Jadhav, Tushar Limbore (Padmabhooshan Vasantdada Patil Institute of Technology, Pune)

[4] GPS and Map Matching Based Vehicle Accident Detection System Md. Syedul Amin, Mohammad Arif Sobhan Bhuiyan Dept. of Electrical, Electronic and Systems Engineering,Universiti Kebangsaan Malaysia 43600 UKM, Bangi, Selangor, Malaysia Mamun Bin Ibne Reaz, Salwa Sheikh Nasir Dept. of Electrical, Electronic and Systems Engineering, Universiti Kebangsaan Malaysia 43600 UKM, Bangi, Selangor, Malaysia

[5]Remote Vehicle Tracking System using GSM Modem and Google Map Muhammad Ridhwan Ahmad Fuad and Micheal Drieberg Electrical and Electronics Engineering Department, Universiti Teknologi PETRONAS, Malaysia E-mail: mdrieberg@petronas.com.my

[6] Low-Altitude Unmanned Aerial Vehicles-Based Internet of Things Services: Comprehensive Survey and Future Perspectives Naser Hossein Motlagh, Tarik Taleb , Osama Arouk¡ Aalto University, Finland,naser.hossein.motlagh@aalto.fi Sejong University and Aalto University, Finland, talebtarik@ieee.org ¡INRIA Sophia Antipolis and Eurecom, France, osama.arouk@inria.fr