

Secure Outstation Cab Service



Gowtham Mamidiseti, Ankitha A.K., Deepa Sarker, D. Siva Naga Sheshi Reddy, C.H. Prudhvi Raj

Abstract- The intent of this thesis was to develop an application that would make booking easier for outstation commutes with guaranteed security. Secure Outstation cab service is an automated prototype which depicts the actual working of an organization that deals with the transport domain. It is a web based platform that allows customers to book their cabs from their comfort of their own home or office. The proposed Secure Outstation Cab Service ensures that the customers can book the cab as per their requirements by logging on to the website. The main aim of Secure Outstation Cab Service is that the users are provided with security, unlike the other cab service systems. On the other hand, we have also developed a mobile based application which is mainly meant for support system that is whenever a ticket is raised, the support staff would handle the queries related to booking, commute, food availability, company, etc. by the users. The prototype which we have developed clearly shows how the software acts as a SaaS model in delivering business models with the customers. The paper focuses on the objective of the application, its problem statement, our analysis on the research work done, proposed work and methodology followed by the workflow, conclusions and future work.

Keywords— Security code, location triggering, android, firebase, real time database, ticket generation, Google map API, support system.

I. INTRODUCTION

A. Introduction

Outstation cab booking service basically deals with the type of service when the customers want to travel to the back of beyond meaning the drop location must be beyond the city boundaries, no matter how far picks up and drop location are. The application which we have designed is mainly meant for outstation rides with security Travel is fun and when it comes to travel by a cab, we get an experience to see the several beauties of nature very closely, and travel explorations to tell the world the glories of our journey. Sometimes all you wish for is a relaxing cab ride from your doorstep to your destination. Firstly, the user has to create an account by registering on the organization's website and login using those credentials. After entering to his account he will find various cab details and their prices after specifying the source and destination.

Revised Manuscript Received on June 30, 2020.

* Correspondence Author

Ms. Deepa Sarker*, Student, Department of Computer Science and Engineering, Presidency University, Bangalore, India.

Ms. Ankitha A.K., Student, Department of Computer Science and Engineering, Presidency University, Bangalore, India.

Mr. C. H. Prudhvi Raj, Student, Department of Computer Science and Engineering, Presidency University, Bangalore, India.

Mr. D. Siva Naga Sheshi Reddy, Student, Department of Computer Science and Engineering, Presidency University, Bangalore, India.

Mr. Gowtham, Assistant Professor, Department of Computer Science and Engineering, Presidency University, Bangalore, India.

© The Authors. Published by Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP). This is an [open access](https://creativecommons.org/licenses/by-nc-nd/4.0/) article under the CC BY-NC-ND license (<http://creativecommons.org/licenses/by-nc-nd/4.0/>)

Finally, he will get the information of the cab that he had booked along with booking ID, car number and other parameters. The primary motive of this project is that the users are provided with security, unlike the other cab service systems.

This system is useful for passengers who want to know about cab routes for travelling between two or many places and how much it would cost for traveling. The mobile based application designed is mainly meant for support information, where the users can ask their queries related to commute, bookings, food service, and about our company by raising tickets. These tickets are then checked upon and resolved by our support team (Ticket resolution).

B. Objective

- **The Objectives include:**

- Live tracking of the cab using a unique security code.
 - Guiding the customers to few places or checkpoints, so as to where they can buy their food from.
- Other specific goals include:
- To produce a web based system that allows customers to register and reserve a cab online and for the company to effectively manage their cab booking business.
 - To ease customer's task whenever they need to book a cab.
 - To produce a mobile based application mainly for customers to seek support by raising tickets.

C. Problem Statement and Approach to problem

The problem statement of our project is to develop an application that could be free from all interferences of internet connectivity provided it is secured and meant for outstation commutes. Present system does not provide security and often lacks connectivity making it difficult to trace the customers when they are on their ride.

Reasons why users will like our online cab service:

- Providing security to customers.
 - No negotiation with the cab driver regarding the price of the ride.
 - Attractive offers for the users.
 - Does not face any hindrances from internet connectivity.
 - Providing the locations of the nearest offered service (restaurant) for the customer.
 - Providing support system to customers in terms of booking, food, commute, destination etc through our mobile app.
- This cab booking system is developed to offer the subsequent services:
- **Online Vehicle Reservation** - Customers can reserve offered cabs online before their expected pick-up date or time.

- *Guaranteed security*- Security for customers, particularly females are provided with the accessibility to live track the locations through the unique security code that is provided to the customer during their start of travel.

II. LITERATURE REVIEW

At present there are many web and mobile applications developed for cab booking system such as Ola, Uber, Savaari etc. but they aren't sure about their security. All of them have hit the market at their launch but even today there are many cases against those systems in terms of their security to customers' especially young children and females. This was mainly happening due to some blockages in their internet connectivity so the tracking of location became a problem which led to many issues. There have been incidents involving the abuse and harassment of women in case of both the services Ola and Uber. However, since these services work as intermediates between the cabbies and consumers, there are several issues that have been observed plaguing the sector. Apart from them, lack of support system, payment issues and lack of communication with the drivers were also some of the problems faced in the existing systems. So to come up with an application that would overcome the above issues, we did go through a few research works and few concepts that can be applied to our project. Some of the research papers that we studied are below: SMS Gateway Integration by Veena K, Katankar et.al, / (IJCSSE) International Journal on Computer Science and Engineering. In their journal, they have depicted that Short message service (SMS) will play a vital role in the future business areas popularly known as mobile banking, organizational marketing system etc. Main goal of the proposed system is to provide multilevel local authentication to the SMS gateway service. This service can be used in multiple departments in their organizations where SMS can be used as a service for getting notifications and for marketing purposes. Through this we could figure out how to use the SMS gateways and how to integrate them in order to send SMS during registration, login and booking as well as during commute for both the web application as well as the mobile application.

A similar work as our cab booking system was also of great help in developing our application especially the web application in terms of the database handling. The journal - Online cab booking system by Aditya Gupte, Anuja Goankar et.al, / (IJSRD) International Journal for Scientific Research and Development where they had showcased the development of an interactive website which operates as a Cab Booking System for customers to conveniently book cabs. A discrete database is maintained to handle all the information needed for error-free statement calculation and generations. It handles various activities such as record updating, maintenance, and searching using HTML, JavaScript and PHP for Front-end and Back-end development respectively. PHPMyAdmin software was used to support and check integrity of databases created.

Using firebase/real time database for the mobile application over the usual MySQL database was an advantage as it could avoid the storage of large unorganized data. In order to build a dynamic application we came

across firebase cloud integration through a previously done research work on firebase application in android. The journal- Application of Firebase in Android App Development-a study by Chunnu Khawas, Pritam Shah-el.al; / (IJCA) International Journal of Computer Application, indicates that the web application has become more dependent on large amount of database and unorganized data such as videos, images, audio, text, files and other arbitrary types. It is difficult for Relational Database Management System (RDBMS) to handle the unstructured data.

III. PROPOSED WORK

D. Requirements for the Proposed Work:

The System and the Software requirements for both the Web Application and Mobile Application include:

- Eclipse
- Java API
- Jar files
- SMS gateways
- Mail SMTP
- Apache Tomcat
- XAMPP Engine (SQL Server Engine)
- Content Delivery Network Integration with CJS(Cascading JavaScript)
- Firebase (Cloud Storage Server)
- Android Studio
- SDK

E. Metamorphosis:

1. Security using unique security code-

In Vecto Cabs web platform as well as mobile platform we have built this feature in order to provide security to the customers during their commute by location triggering and tracking by using the location SMS mechanism. After the user registers with his mail ID, phone number and Name, the user gets an OTP in SMS as well as a verification code to the mail following by which he/she has to enter their source and destination to get the route details and map view. Once the user is convenient with the route, distance and duration he/she can proceed with the booking. Invoice generation is done once the booking is confirmed after which the unique security code is sent to the customers during the commute by the admin itself from the APK file. It is a unique four digit security code which is to be kept confidential by the user and the guardian only. Once the user receives the security code, he/she can send that code to their parents/guardian. The location of the user is sent as a URL after the code is sent to the parent. We have deployed location triggering mechanism to implement this.

2. Location Triggering using unique security code-

The source and destination from the user is and sent to the Google maps API. It searches by latitude and longitude and fetches them for the source as well as destination. After which the map view is available.

The map is generated by rendering the source as point ‘a’ and the destination as point ‘b’. The red point on the map is named as point ‘a’ for the source which will be mapped to the latitude and longitude returned by the Google map API to that particular point. Similarly, for point which is marked in green (point ‘b’) will be mapped to the latitude-longitude returned by the API.

The shortest routes will be highlighted in blue color. So basically we make use of the Google maps API to generate the location, distance, estimated journey time, cost etc.

Various Google map APIs used – Maps SDK for Android, Maps JavaScript API, Maps embedded API, Polyline Utility in the Maps SDK for Android Utility Library and Geometry Library in the Maps JavaScript API, Universal, cross platform Maps URLs, Intents in the Maps SDK for Android, Google maximum zoom imagery service in the Maps JavaScript API.

3. Ticket Generation and Ticket Resolution –

An online ticketing system streamlines how you communicate with your customers. It brings in customer conversations from multiple channels like phone, email, and social media into a single, easily accessible location where you get the entire context you need. An efficient help desk ticketing system enables us to automate the routine support tasks to reduce the excessive workload and to ease the efficiency of the team. Our mobile application is built to ensure ticket generation mainly to-

1. Master multi-channel communication
2. Increase efficiency with automation
3. Make self-service easier for customers
4. Increase customer loyalty

For a ticketing system, a document or ticket is created that records all the interactions and activities taking place in a service desk or support team. The tickets generated will be accessible to both admin/rep and to the customer to log their communications to one continuous thread. If at all any confusion or mismatch occurs, then both the parties can refer the thread anytime and review the previous information. Once the ticket is created, the admin/reps can then analyze the issue and work on it. When they have any notifications, updates or resolution to be given then they can alert that to the customer through the ticket. Similarly, if the customer has any questions/queries, they also can use the same ticket to communicate or can verbally contact the team. When the queries/issues from the customer side are resolved, then the ticket status will be updated from “initiated” to “closed” which is nothing but ticket resolution. If the same customer wants to reopen the ticket, that can be possible by accessing the same person that they worked with previously rather than raising a fresh ticket again. In our project we have used Firebase as the cloud storage database where all our ticket generated data will be viewed by the admin at the Admin Interface or the real-time firebase portal. Once the queries are resolved either over verbal conversations or online the ticket status will be updated. The customer can raise their queries by selecting their queries based on commute, food, support, SMS and mail services etc. after which their ticket will be generated with a ticket ID and the query status.

IV. SYSTEM ARCHITECTURE

System Architecture for both Web Application and Mobile Application are as follows:

F. System Architecture for Web application.

The system architecture that our web application follows is 3-tier architecture as depicted in Fig. 1.

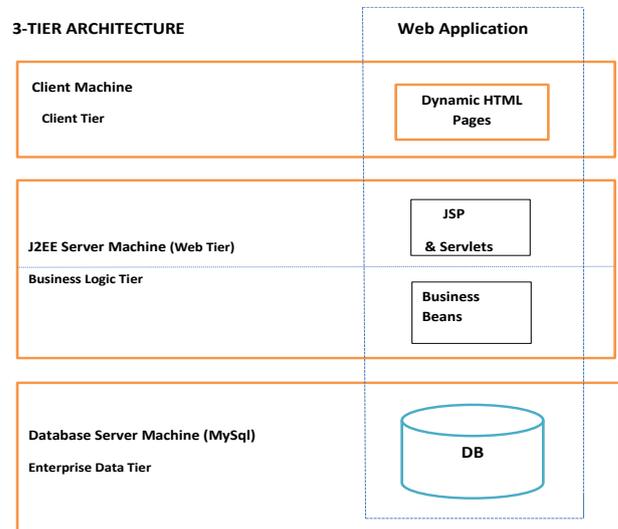


Fig. 1. System Architecture for Web Application

The System architecture we followed is single page application architecture. It is the era of minimalism, where a single-page web app is more popular. The topmost level of the web application is the presentation layer which displays information such as browsing parking lot, client, system user login, booking, etc. It communicates with other layers by which it puts out the results to the browser/client tier and all other tiers in the network. The Business Access Layer Project is responsible for all the business logic of the application by performing detailed processing; it interfaces the Data Access Layer and the presentation Layer. The Data Access Layer subsystem is responsible for accessing the database for data retrieval and insertion. Working of the web application - With web applications, we have the server vs. the client side. In essence, there are two programs running concurrently: The code which lives in the browser and responds to user input and the code which lives on the server and responds to HTTP requests. For writing server-side code, Java is used. Any code that is able to respond to HTTP requests has the ability to run on a server. The server-side code is responsible for creating the page that the user requested as well as storing different types of data, including user profiles and user input. It is never seen by the end-user. A combination of CSS, HTML, and JavaScript is used for writing the client-side code. This code is parsed by the web browser. Unlike the server-side code, client-side code can be seen as well as modified by the user. It reacts to user input. The client-side code communicates only via HTTP requests and is not able to read files off a server directly. The role is to accede to the requests of clients, including browsers and mobile apps via secure protocols. The requests could pertain to page resources or could also be related to a REST API. Web servers are intrinsic to the working of web apps,

mandating the need for increased emphasis on web server architecture, including the server’s physical capacity – storage, memory, computing power, and performance, apart from the application tiers. One of the distinct advantages of this architecture is the ability to combine and rely on the Java native tools, and frameworks for creating applications straddling the entire spectrum – from simple to the most complex of applications.

G. System Architecture for Mobile application

The system architecture that Vecto Cabs for Mobile application follows is as depicted in Fig. 2.

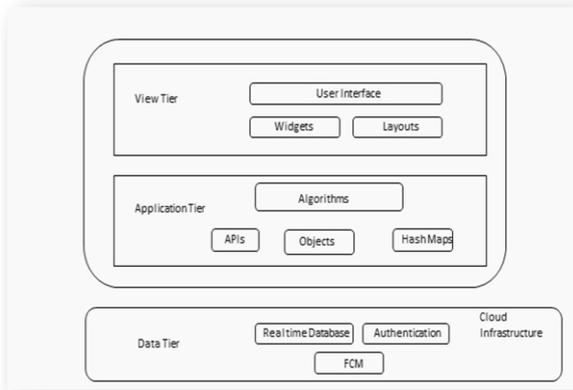


Fig. 2. System Architecture for Mobile Application

Our mobile app architecture falls under the Server-less architecture using the BAAS (back end as a service) Firebase which takes care of managing server side logic and state. Here Server-less does not mean that the app is not dependent on a server, but rather means that the app is reliant on third party services commonly known as FaaS (Function as a Service) or BaaS. In the view tier, we have the UI Interface with the widgets and layouts. In the Application Layer, we have used Algorithms nothing but APIs, objects and hash maps since firebase is a key-value pair. So hash maps are efficient for locating a value based on a key and also inserting and deleting values based on the key. It will also take care of ensuring that the authentication is done successfully, either through email and password or through third party authentication services, such as Google, Facebook, etc.

V. WORK FLOW

H. Workflow of Web Application

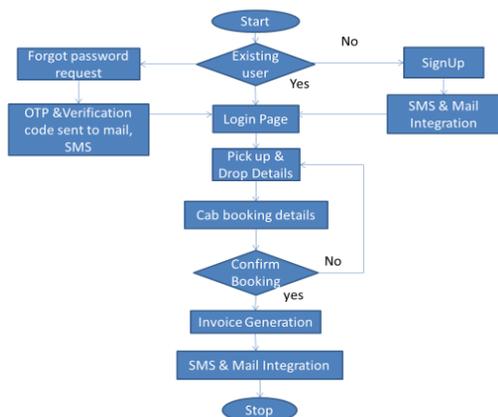


Fig. 3. Workflow of Web Application

- 1) User opens the Application leading to the Home Page of Vecto Cabs.
- 2) If the user has an existing account in the application or if/she has previously registered then the user can directly login using his email ID and password created.
- 3) If the user is a new user or has no account registered in the application, then the user has Sign Up/Register using his/her Email ID, Name and set the password. Immediately, OTP to set the password will be sent via SMS and a verification code via email will be sent to the user done by the TextLocal API and Gmail SMTP protocol.
- 4) In case if the user forgets his/her password during Login, then forgot password request can be activated after which OTP and verification code will be sent to the user to set his/her new password and then proceed to login with the new password.
- 5) After the login is successful, the page redirects to the main portal where the route for the journey has to be chosen by the user by giving his source and destination. The map is rendered by Google maps API showing the location, distance and view of the destination.
- 6) Once the route is displayed, the user now has to choose his date and day for the commute as well as the type of car he prefers (mini, prime, sedan etc.). He then gets the booking details as invoice where in all the details such as the Booking ID, destination, source, date of journey, distance, cost, driver, driver contact number, car number, car name and type. If the user is to agreeable to the details then he can confirm the booking.
- 7) After the booking confirmation, a mail will be processed to the registered customer with all his booking details and also a confirmation SMS with the booking ID. After all the process is completed, the page redirects to the Home page where he can view his booking history, cancel the ride and also update the user’s profile.

I. Workflow of Mobile Application

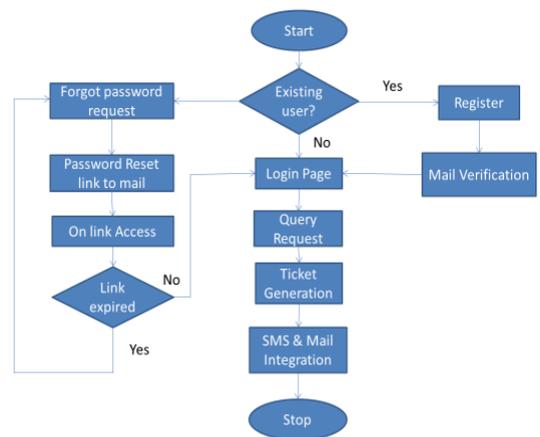


Fig. 4. Workflow of Mobile Application

- 1) Unlike the web application, our mobile application is exclusively for the support system of the customers.

- 2) Application opens up with the Login Page where the user has to login using their email ID and password. In case, if the user has forgotten the password, then he can request for password reset for which a password reset link will be sent to the registered mail. The user has to access that link, set the new password and login.
- 3) If the user has not registered yet or if is a new user, then he/she has to register using Name, email ID, mobile number, set password and confirm.
- 4) A validation link will be sent to the registered mail ID in order to verify the mail address after which the user can sign in into the application using the registered mail ID and password.
- 5) The page will be redirected to the ticket generation page where the user can raise tickets based on their queries related to commute, food, support and SMS and mail services.
- 6) Details regarding the ticket ID and query status will be sent to the customer through SMS and mail.
- 7) The tickets can be viewed by the admin at the firebase admin interface and the support team can contact the customer to resolve the queries through verbal conversation. If the query is solved then the ticket status will be updated to as closed which means that ticket is closed or resolved.
- 8) After the ticket resolution is done, the customer gets a mail and SMS which indicates that about their ticket status.

VI. CONCLUSION AND FUTURE WORK

All the requirements that were elucidated with respect to the mobile application and the web application were successfully done with all the possible testing and were fully functional without bugs. The current application has been built bearing in mind the current demands and requirements of a decent web application that would get a cab to the required location with a minimal waiting time and minimal frustration, and also a mobile application that can be used for support system, that is whenever ticket is raised support staff would immediately assist the queries raised by the customers, and the queries would be related to booking, commute, company by the users. The outcome of the Vecto Cabs application was to provide an easy to use interface for scheduling a pickup by the user and to view all the pickups that the user made using the application. Additionally, the other main features of the app include- Obtaining the current location using the unique security code sent during the commute, showing the estimated duration, distance of the ride, showing the route map along with the 3D view and satellite view of the location, sending invoice and booking details via mail and SMS, support to the customers through ticket generation and resolution. Apart from all this, the main importance was given to the fact that we are providing outstation cabs along with guaranteed security as the present systems face a lot of security issues especially females travelling alone for long distances. So in order to overcome the present security issues, our application meets the crisis by providing a unique security code which can be sent to the guardians or

parents to track their location without any hindrances in internet connectivity.

As part of future enhancements, this application can be extended in an extensive basis. In the future, the web application as well as the mobile application can both be made as a common platform. Apart from that we would also develop a separate Admin Dashboard for the Mobile Application to fetch tokens by typing the email-ID. This application has a large scope of enhancement.

ACKNOWLEDGEMENT

First of all, we are indebted to the GOD ALMIGHTY for giving me an opportunity to excel in our efforts to complete this project on time.

We are extremely grateful to **Dr. C. Prabhakar Reddy**, Dean, School of Engineering and **Dr. Mohan K G**, Head of the Department, Department of Computer Science and Engineering, for providing all the required resources for the successful completion of this project.

We sincerely thank our project guide, **Prof. Gowtham Mamidiseti**, for his guidance, help and motivation. Apart from the area of work, we learnt a lot from him, which we are sure, will be useful in different stages of our life. We would like to express our gratitude to Faculty Coordinators and Faculty, for their review and many helpful comments.

We would like to acknowledge the support and encouragement of our friends.

REFERENCES

1. Amit Vashistha, Rohit Goyal, Aman Chaudhary, Prabu S. (2018) 'Cab Booking Application', (*IJARIT*) *International Journal of Advance Research, Ideas and Innovations in Technology* vol. 4, issue 2.
2. Aditya Gupte, Anuja Gaonkar. (2017) 'Online Cab Booking System', (*IJSRD*) *International Journal of Scientific Research and Development* vol. 10, issue 10, pp. 679-683.
3. Chunnu Khawas, Pritam Shah (2018) 'Application of Firebase in Android App Development-A Study', *International Journal of Computer Applications (0975 – 8887)* vol. 179, issue 46.
4. Veena K. Katankar, Dr. V. M. Thakare. (2010) 'Short Message Service using SMS Gateway', (*IJCSE*) *International Journal on Computer Science and Engineering*, vol. 02, issue 04, pp. 1487-1491.

AUTHORS PROFILE

Ms. Deepa Sarker is pursuing her Bachelor of Technology in Computer Science and Engineering, 8th semester at Presidency University, Bangalore.

Ms. Ankitha A.K. is pursuing her Bachelor of Technology in Computer Science and Engineering, 8th semester at Presidency University, Bangalore.

Mr. C. H. Prudhvi Raj is pursuing his Bachelor of Technology in Computer Science and Engineering, 8th semester at Presidency University, Bangalore.

Mr. D. Siva Naga Sheshi Reddy is pursuing his Bachelor of Technology in Computer Science and Engineering, 8th semester at Presidency University, Bangalore.

Mr. Gowtham Mamidiseti is working as an Assistant Professor at Presidency University. He is currently pursuing his PhD from Acharya Nagarjuna University. He has also published many International Journals and was a GATE rank holder in 2010.