SEMESTER 2020-21

SLOT: F2

SOFTWARE ENGINEERING

FINAL REPORT

Submitted to-

Prof.Deepikaa S

Submitted by,

Vibhuti Srivastava 18BCI0111

Rakshit Singh Bisen 18BCI0149

Raj Adroja 18BCE0754

PROJECT TITLE-

GET SET TRAVEL- A TRAVELLING GUIDE

PROJECT ABSTRACT :

This is a system for the people travelling across different cities. This software will take the details of the customers in the beginning and authenticate the customer with the Adhar number. The software will also provide the current state of congestion of traffic on a particular route, and a notification will be raised which would be verified andthe nearest traffic police will be informed. People can also report the accidents in that area and the police and a hospital will be notified. The aspects include information about public transportation options, schedules, fares, real time location ofthe vehicle will be provided. The Routes of the bus, trains and metro will be shown to the user according to the desired location of visit. This will also allow the user to report the transportation medium in case of any problems, feedbacks will also be taken from the user and handled by our team if it is a software requirement or if it is some transportation requirement then the concerned government body will be reported. It will be a 24x7facility. The nearest bus stops, railway stations and other landmarks where the required transport could be taken. Based upon the experiences of the person the best mode of transport to travel to a particular place would be identified and shown as the priority mode of transportation to be opted by the future customers. Whenever an ambulance is in emergency an alert is raised and the notification to all the policeman on the way is given to get the way cleared out for the ambulance well in advance.

AIM-

- work in collaboration with the users for it continuous development to ensure its working for a long run. Any sort of feedbacks and suggestion will be viewed seriously by the team and will make best effort in making the suggested changes necessary

- it will give users to give their own views about how the system should work along with it, gives the users feel of more security and reliably on the proposed model

- that almost any type of user can make the use of it from Ambulance driver to policeman and even by some common people which will surely not focus on only one organization like any other models presently working in the market.

INTRODUCTION

Theoretical Background:

It is software which will guide people to travel across different places. The aspects include information about public transportation options, schedules, fares, real time location of the vehicle will be provided. Based upon the experiences of the person the best mode of transport to travel to a particular place would be identified and shown as the priority mode of transportation to be opted by the future customers. The Routes of the bus, trains and metro will be shown to the user according to the desired location of visit. This software will take the details of the customers in the beginning and authenticate the customer with the Aadhaar number.

The software will also provide the current state of congestion of traffic on a particular route, and a notification will be raised which would be verified and the nearest traffic police will be informed. People can also report the accidents in that area and the police and a hospital will be notified. This will also allow the user to report the transportation medium in case of any problems, feedbacks will also be taken from the user and those feedbacks will be taken positively by our team. The nearest bus stops, railway stations and other landmarks where the required transport could be taken by the user using our application. Based upon the experiences of the person the best mode of transport to travel to a particular place would be identified and shown as the priority mode of transportation to be opted by the future customers.

Also, whenever an ambulance is in a hurry an alert is raised and the notification to the entire policeman is given to get the way cleared out for the ambulance well in advance. In this way the ambulance carrying the patient which should be priority at all times will always reach on time.

Motivation:

- The motivation behind the working of this project is to provide user different and unique type of experience and make it better and better with time with the involvement of their feedbacks and improvements suggested by them which in turn will result in the huge scale development of the project with time.

- It gives users a unique and different type of experience as compared to the other present working models by asking the users to be active in terms of giving feedbacks and suggestions. Also asking them to how improve our current system.

- Also the model will be reliable as it will be ensured by thorough unit testing and tests after building of each model, secure which means all user data will be kept confidential and a backup will also be created in case of any unexpected loss and will be highly maintainable as modules will be quite straight forward and are easy to understand hence can be upgraded from time to time.

Objective of proposed Model:

- Objective of the proposed model is that it will work in collaboration with the users for it continuous development to ensure its working for a long run .

- Any sort of feedbacks and suggestion will be viewed seriously by the team and will make best effort in making the suggested changes necessary .

- The other objective of the proposed model is that it will give users to give their own views about how the system should work along with it, gives the users feel of more security and reliably on the proposed model .

- Along with all the above objectives this proposed model main objective is that almost any type of user can make the use of it from Ambulance driver to policeman and even by some common people which will surely not focus on only one organization like any other models presently working in the market.

Existing system problem:

There aren't many websites for real time travelling system. This is why our project is unique. Those websites that are there lags complete reference about traffic reports, accident reports, best possible routes, daily schedules and finding best routes between location via any other place.

Novel idea behind our project :

- Our project gives a platform where a user can get various types information required for his travelling, like route, modes of transportation, timings, traffic, accident in past , ambulance and police station facility between his starting and ending destination, everything on a single platform.
- It is software which will guide people to travel across different places. The aspects include information about public transportation options, schedules, fares, real time location of the vehicle will be provided.
- Based upon the experiences of the person the best mode of transport to travel to a particular place would be identified and shown as the priority mode of transportation to be opted by the future customers.
- Also it will focus on the mishap that happens with us during the journey. Here one can complain regarding the traffic jams and can also lend a helping hand towards the unknown by initiating an ambulance service for the accident on the way . Also traffic problem can be resolved as user can directly send mail to police.

PROCESS MODEL CHOSEN WITH JUSTIFICATION

The process model chosen is:
Rapid application development model (RAD)

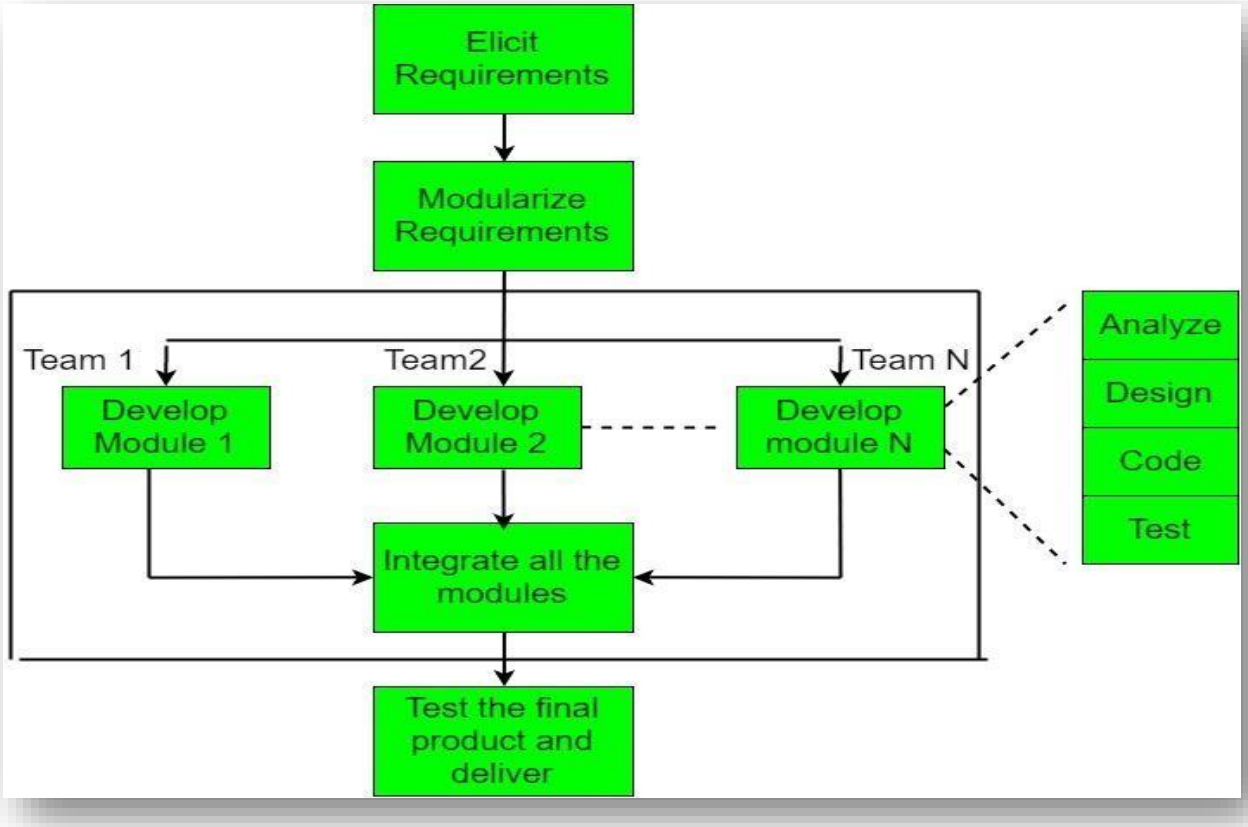
We are using this model as: (JUSTIFICATION)

- our project contains different small modules . the project can be broken down into small modules wherein each module can be assigned independently to separate teams. These modules can finally be combined to form the final product. This is the basic criteria of our RAD model
- . Development of each module involves the various basic steps as in waterfall model i.e analyzing, designing, coding and then testing, etc.
- We also have a self organising team.
- Another striking feature of our project which matches with this model is a short time span i.e the time frame for delivery(time-box) is generally 60-90 days.
- Feedback from the customer is available at initial stages
- Reduced costs as fewer developers are required.

- Use of powerful development tools results in better quality products in comparatively shorter time spans.

- The progress and development of the project can be measured through the various stages.

- It is easier to accommodate changing requirements due to the short iteration time spans.


Elicit Requirements
Modularize Requirements
Team 1
Team2
Team N
Develop Module 1
Develop Module 2
Develop module N
Analyze
Design
Code
Test
Integrate all the modules
Test the final product and deliver

❖ STAKEHOLDERS , PROJECT AND PRODUCT SCOPE:

Various stakeholders of our project are:

a) **THE DEVELOPER- :**. The project developers are the creators of the project. They write the codes and designs and integrate the various components together. In our project we assigned the role of developer to Vibhuti Srivastava and Rakshit Singh Bisen. The testing of the various components is also done by some member of the developer team. The team has the technical expertise to convert the demands of the customers (clients or end users) into a software. The developer team has regular meetings with the other stakeholders to notify them about progress and also take the requirements of the other stakeholders into their account.

b) **RESOURCE MANAGER-** Because project managers are in the position of borrowing resources, other managers control their resources. In our project, we assigned this job to Rakshit . So their relationships with people are especially important. If their relationship is good, they may be able to consistently acquire the best staff and the best equipment for their projects. If relationships aren't good, they may find themselves not able to get good people or equipment needed on the project.

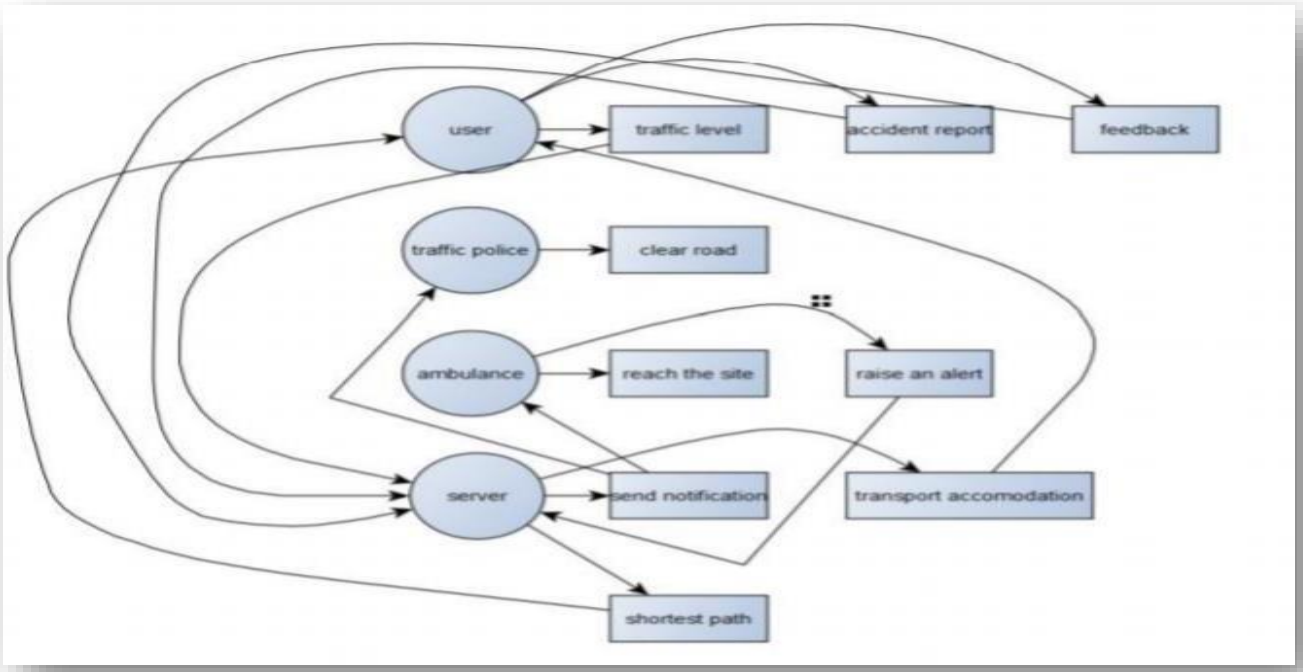
c) **THE END USERS:** External customers are the customers when projects could be marketed to outside customers. In the case of our project ,the end user/customers would be the people of all ages who uses our website whenever they need to travel from one place to another .They can view the route, have prior knowledge of traffic on that route, any accident in past and all the types of transportation available between those 2 places along with their timings .

d) **THE PROJECT MANAGER-:**In our case, our faculty Prof Nalini N Ma'am is the project manager. The project manager has the task of monitoring the overall functioning of the project, the project manager sets the deadlines for various deliverable submissions and the project manager also serves as link between the client and the developers of the project. The manager keeps track of the progress of each developer in the team and also assigns work to different members of the team. The project manager presides over review meetings held at regular intervals during the project development in order to keep track of overall project progress and to get an idea of estimated time to complete the project.

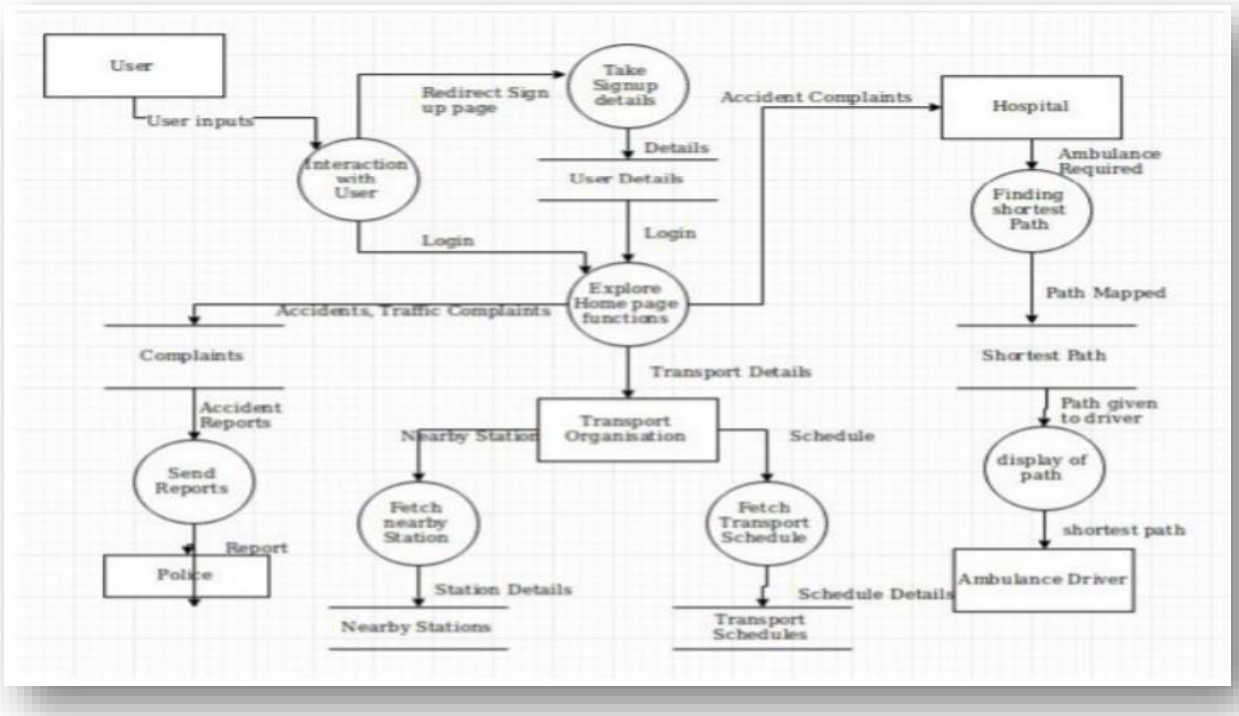
e) **GOVERNMENT:** Project managers working in certain heavily regulated environments will have to deal with government regulators and departments. These can include all or some levels of government from municipal, provincial, federal, to international. As we are taking

All Diagrams-

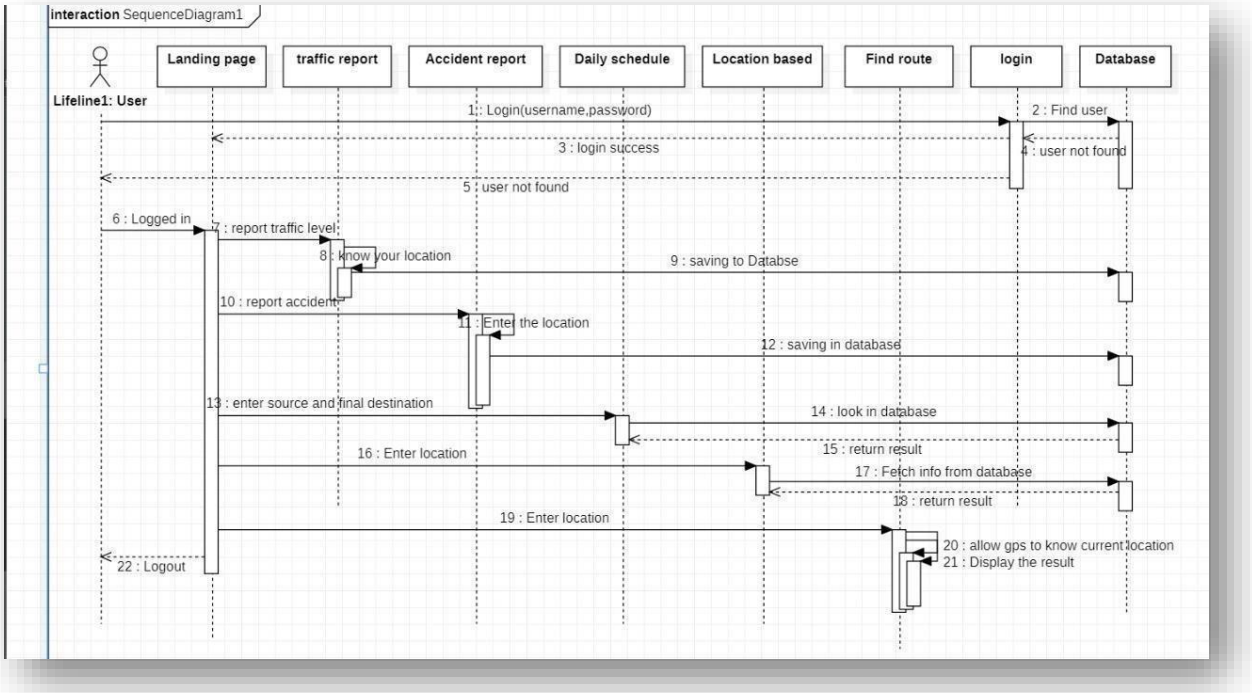
1) MODULE DEPENDENCY DIAGRAM-



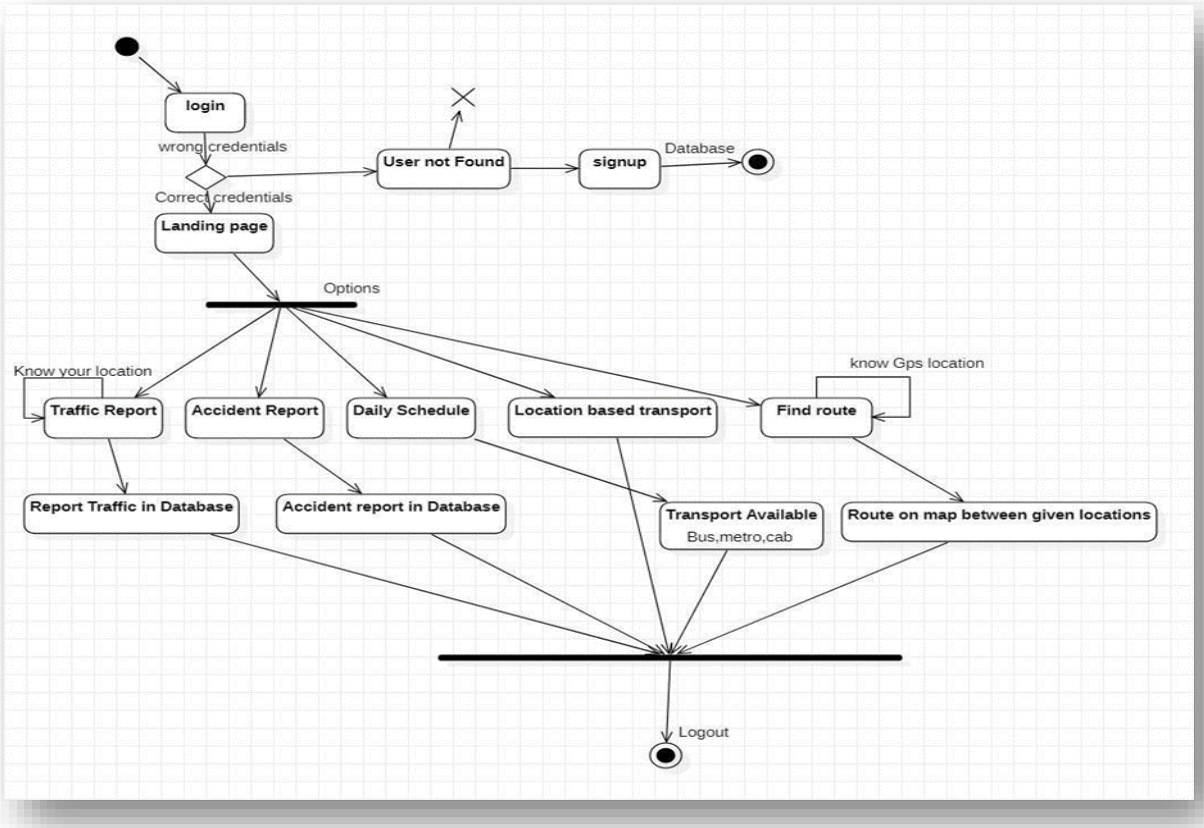
2) WBS-



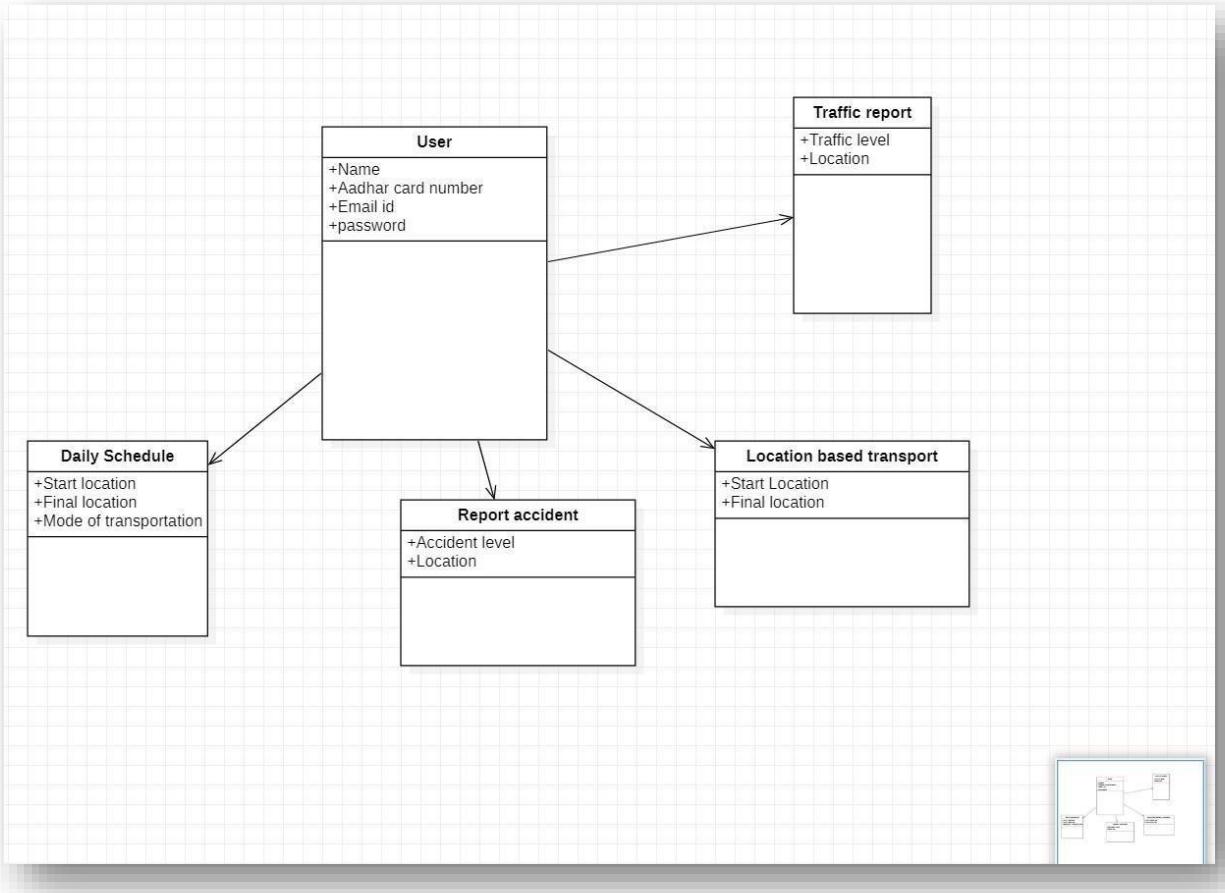
3) SEQUENCE DIAGRAM-



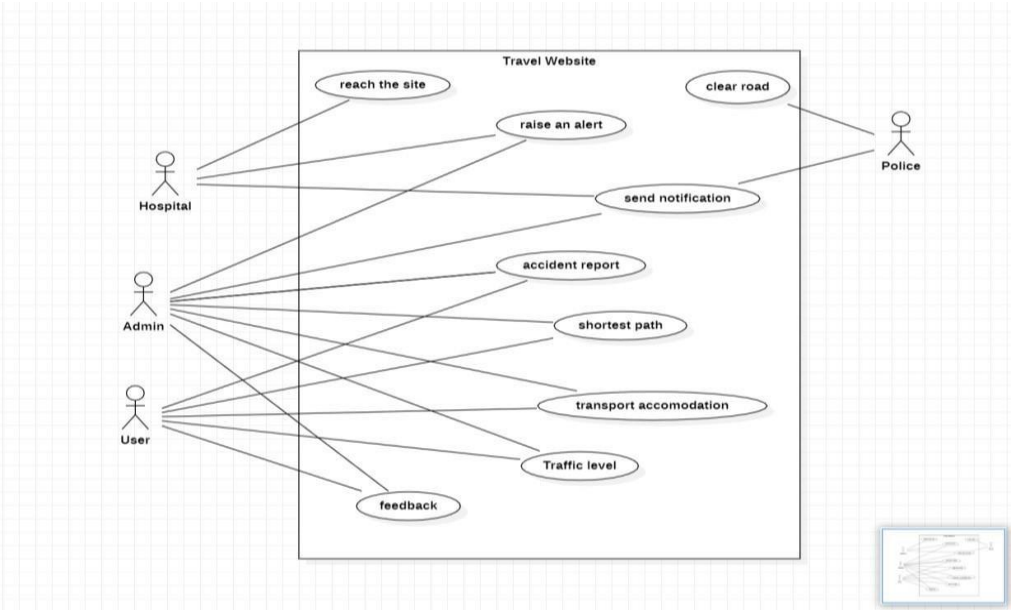
4) STATE DIAGRAM –



5)STATE DIAGRAM-



6) USE CASE



7)USE CASE DESCRIPTION

USER CASE ID	Registration/login
PRIMARY ACTOR	User
SECONDARY ACTOR	Database administrator
TRIGGER	When the user clicks the signup button
PRECONDITIONS	User is not already registered
BASIC FLOW	The user clicks the sign-up button, then when the registration page opens up. The user enters the various details such as name, Aadhar card number, email, password

USER CASE ID	Report Traffic
PRIMARY ACTOR	User
SECONDARY ACTOR	Database administrator
TRIGGER	When the user clicks the Report Traffic button
PRECONDITIONS	User is a valid customer and logged in
BASIC FLOW	The user clicks the report traffic button and can report the traffic that he/she faced during the journey and can give the level of traffic and the location of the traffic

USER CASE ID	Location Based Transport
PRIMARY ACTOR	User
SECONDARY ACTOR	Database administrator
TRIGGER	When the user clicks Location based Transport

PRECONDITIONS	User is a valid customer and logged in
BASIC FLOW	The user clicks the Location Based transport he/she can give the start and destination location and search for available mode of transport

USER CASE ID	Report accident
PRIMARY ACTOR	User
SECONDARY ACTOR	Database administrator
TRIGGER	When the user clicks report accident button
PRECONDITIONS	User is a valid customer and logged in
BASIC FLOW	The user can report an accident, the level of accident and the location of accident

USER CASE ID	Daily schedules
PRIMARY ACTOR	User

SECONDARY ACTOR	Database administrator
TRIGGER	When the user clicks daily schedules
PRECONDITIONS	User is a valid customer and logged in
BASIC FLOW	The user can view the timetable of all mode of transportation

USER CASE ID	Check Weather
PRIMARY ACTOR	User
SECONDARY ACTOR	Database administrator
TRIGGER	When the user clicks Check Weather
PRECONDITIONS	User is a valid customer and logged in
BASIC FLOW	The user can view the Weather details of any place of his choice

PRODUCT SCOPE:

1) First a login page will open up which contains a form with fields name and password to login. This page will also contain a link to a sign up page for first time users.

2) The sign up page consists of a form which contains fields such as name, gender, email, password and confirm password, upon filling the details, the details are stored in the database and the user is redirected to the login page.

3) After logging in the user is sent to the landing page and on left side user can see options such as report accident,report traffic,location based transport,daily schedules and find route.

4) In report accident user can enter the location or use the option of choose current location and select accident level(high, medium ,low) and report.

5) In report traffic user can enter the location or use the option of choose current location and select traffic level(high,medium,low) and report

6) In location based transport user can Enter starting location and Destination location and find the mode of transport you can take to reach.This is mostly for short distances.

7) In daily schedules user can Enter starting location and Destination location and find the mode of transport and find the train/metro schedule or bus schedule to travel.

8) In find route user can find route to two different places.

9) The UI for the product will be made as simple as possible so users of every background can easily use our product.

PROJECT SCOPE:

- Our project gives a platform where a user can get various types information required for his travelling, like route, modes of transportation, timings, traffic, accident in past , ambulance and police station facility between his starting and ending destination, everything on a single platform.
- It is software which will guide people to travel across different places. The aspects include information about public transportation options, schedules, fares, real time location of the vehicle will be provided.
- Based upon the experiences of the person the best mode of transport to travel to a particular place would be identified and shown as the priority mode of transportation to be opted by the future customers.
- Also it will focus on the mishap that happens with us during the journey. Here one can complain regarding the traffic jams and can also lend a helping hand towards the unknown by initiating an ambulance service for the accident on the way

❖LITERATURE SURVEY/ RESRAECH PAPERS-

1) A literature review on NoSQL database for big data processing ,June 2018 , At International Journal of Engineering and Technology

Authors- Md. Asraf Ali, Md. Razu Ahmed , Kenneth Sundaraj

Aim of the present study was to literature review on the NoSQL Database for Big Data processing including the structural issues and the real-time data mining techniques to extract the estimated valuable information. Methods: We searched the Springer Link and IEEE Xplore online databases for articles published in English language during the last seven years (between January 2011 and December 2017). We specifically searched for two keywords ("NoSQL" and "Big Data") to find the articles. The inclusion criteria were articles on the use of performance comparison on valuable information processing in the field of Big Data through NoSQL databases. Results: In the 18 selected articles, this review identified 8 articles which provided various suitable recommendations on NoSQL databases for specific area focus on the value chain of Big Data, 5 articles described the performance comparison of different NoSQL databases, 2 articles presented the background of basics characteristics data model for NoSQL, 1 article denoted the storage in respect of cloud computing and 2 articles focused the transactions of NoSQL. Conclusion: In this literature, we presented the NoSQL databases for Big Data processing including its transactional and structural issues. Additionally, we highlight research directions and challenges in relation to Big Data processing. Therefore, we believe that the information contained in this review will incredible support and guide the progress of the Big Data

2) A Review Paper on MERN Stack for Web Development Pragati Bhardwaj 1 , Dr Dinesh Kumar 2 P.G. Scholar, Department of Computer Engineering, SRCEM, Palwal, Haryana, India1 HOD & Associate Professor, Department of Computer Engineering, SRCEM, Palwal, Haryana, India

Today Developers around the world are making efforts to enhance user experience of using application as well as they are trying to enhance the developer's workflow of designing applications to deliver projects and rollout change requests under strict timeline. Stacks can be used to build web applications in the shortest span of time. The advantage of such JavaScript stacks helps to build an integrated solution by using only JavaScript. This paper provides the introduction to concept and describes the MERN stack open source and explains the meaning of them along with their future scope. This paper also describe that React is preferred more to Angular in front-end development.

3) Smart tourism: State of the art and literature

In the modern era, the tourism sector has grown to be one of the dominant sectors globally while technology continues to evolve. These facts have given birth to the "Smart Tourism" concept which can be characterized as a progression from traditional tourism. In order to realize an actual Smart Tourism experience, the proper services need to be delivered to the right user at the right time with the best possible way. During the last six years, there has been a significant amount of research in the Smart Tourism field which, to the best of our knowledge, have not yet been presented in a thorough literature review. In this paper, after carefully reviewing a wide number of papers, we have managed to identify the most commonly used approaches and basic concepts in the Smart Tourism sector and present them in detail along with the papers that focus on them. In this study, "key concepts" include: Privacy Preserving, Context Awareness, Cultural Heritage, Recommender Systems, Social Media, Internet of Things, User Experience, Real Time, User Modeling, Augmented Reality and Big Data. At the same time, major Smart Tourism challenges are presented so as to lay the foundations for future researches in the field.

4) Management of tourism group and technology of the personalized tour based on RFID BY IEEE

Published in: **2011 Chinese Control and Decision Conference (CCDC)**

In today's world, tourism has taken its place as one of the major sources of foreign exchange for both developed and developing economies. Therefore, efficient management of tourists and tourism has become important in accessing the benefits accruing from this sector of the world economy. This paper proposes a research program about group travel management and personalized guide technology based on RFID, to improve the management efficiency and enhance the quality of tourism. The paper presents the overall design of the tourism group management system and the working principle, and then describes the design of the hardware and the software of the system in detail, as well as introduces the hardware schematic diagram of each module and the software design flow of the system.

5) Design and Development of Tour Management System using Android

Authors- Aishwarya Bhat , Ayesha Thasneema

Tour management android application is an emerging technology in today's world. Cell phones have become a necessary tool for many people throughout the world. This application guides the tourists, offers them with all the relevant information such as images, weather condition and description about the places they want to visit. It is especially useful for the users to visit the places having no idea regarding the place. The users can get a better guidance about the places they want to visit by making use of the Google map service provided in the application. Users can view various tour and travel destinations.

TOOLS FOR PROJECT:

Mlab

Heroku

HTML CSS

JavaScript

NodeJs

- ***HTML:***

Hypertext Mark-up Language (HTML) is the standard mark-up language for reports intended to be shown in an internet browser. It very well may be helped by innovations,

for example, Cascading Style Sheets (CSS) and scripting dialects, for example, JavaScript. Web programs get HTML archives from a web worker or from neighbourhood stockpiling and render the records into media site pages. HTML depicts the structure of a site page semantically and initially included signs for the presence of the document.HTML components are the structure squares of HTML pages. With HTML builds, pictures and different items, for example, intelligent structures might be installed into the delivered page. HTML gives a way to make organized archives by signifying basic semantics for text, for example, headings, passages, records, connections, cites and different things. HTML components are outlined by labels, composed utilizing point sections.

❖ ***CSS:***

Cascading Style Sheets (CSS) is a template language utilized for depicting the introduction of a report written in a mark-up language, for example, HTML. CSS is a foundation innovation of the World Wide Web, close by HTML and JavaScript. CSS is intended to empower the detachment of introduction and substance, including design, tones, and fonts. This partition can improve content openness, give greater adaptability and control in the particular of introduction qualities, empower various site pages to share organizing by determining the important CSS in a different.CSS record which lessens multifaceted
nature and reiteration in the auxiliary substance just as empowering the.CSS document to be reserved to improve the page load speed between the pages that share the document and its arranging. Division of arranging and substance likewise makes it achievable to introduce a similar mark-up page in various styles for various delivering strategies, for example, on-screen, on paper, by voice (by means of discourse-based program or screen peruse), and on Braille-based material gadgets. CSS additionally has rules for substitute designing if the substance is gotten to on a cell phone.

❖ ***JavaScript:***

JavaScript often abbreviated as JS, is a programming language that conforms to the ECMA Script specification. JavaScript is high-level, often just-in-time compiled, and multi-paradigm. It has curly-bracket syntax, dynamic typing, prototype-based object-orientation, and first-class functions. Alongside HTML and CSS, JavaScript is one of the core technologies of the World Wide Web. JavaScript enables interactive web pages and is an essential part of web applications. The vast majority of websites use it for client-side page behaviour, and all major web browsers have a dedicated JavaScript engine to execute it. As a multi-paradigm language, JavaScript supports event-driven, functional, and imperative programming styles. It has application programming interfaces (APIs) for working with text, dates, regular expressions, standard data structures, and the Document Object Model (DOM). However, the language itself does not include any input/output (I/O), such as networking, storage, or graphics facilities, as the host environment (usually a web browser) provides those APIs.

❖ ***Node.js:***

Node.js is an open-source, cross-platform, back-end, JavaScript runtime environment that executes JavaScript code outside a web browser. Node.js lets developers use JavaScript to write command line tools and for server-side scripting running scripts server-side to produce dynamic web page content before the page is sent to the user's web browser. Consequently, Node.js represents a "JavaScript everywhere" paradigm, unifying web-application development around a single programming language, rather than different languages for server- and client-side scripts.

MODULE DESCRIPTION-

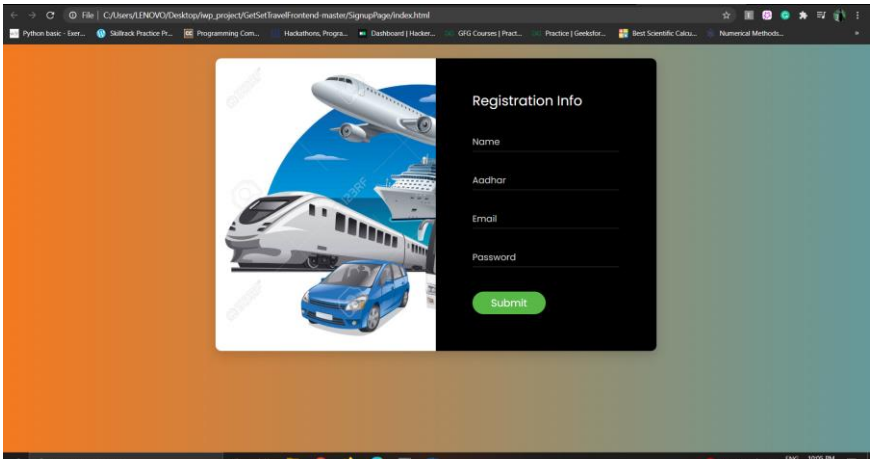
1) ***LOGIN PAGE***- First a login page will open up which contains a form with fields name and password to login. This page will also contain a link to a sign up page for first time users.

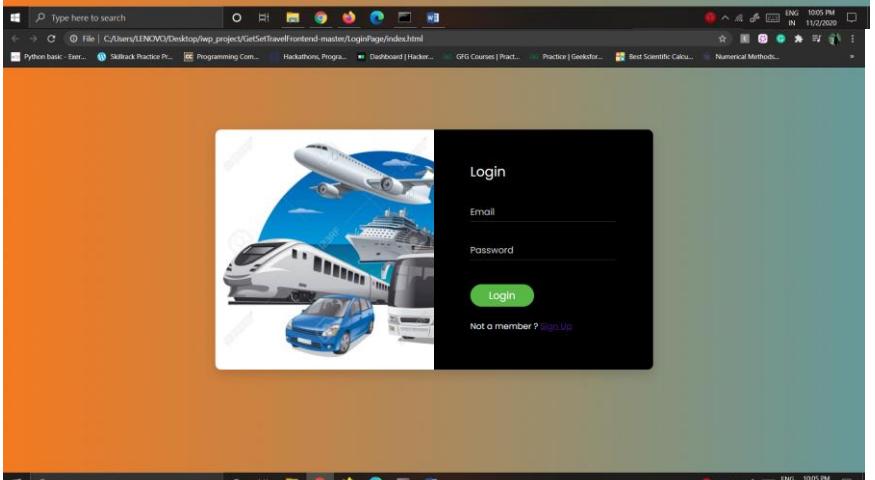
2) ***SIGNUP PAGE-*** The sign up page consists of a form which contains fields such as name, gender, email, password and confirm password, upon filling the details, the details are stored in the database and the user is redirected to the login page.

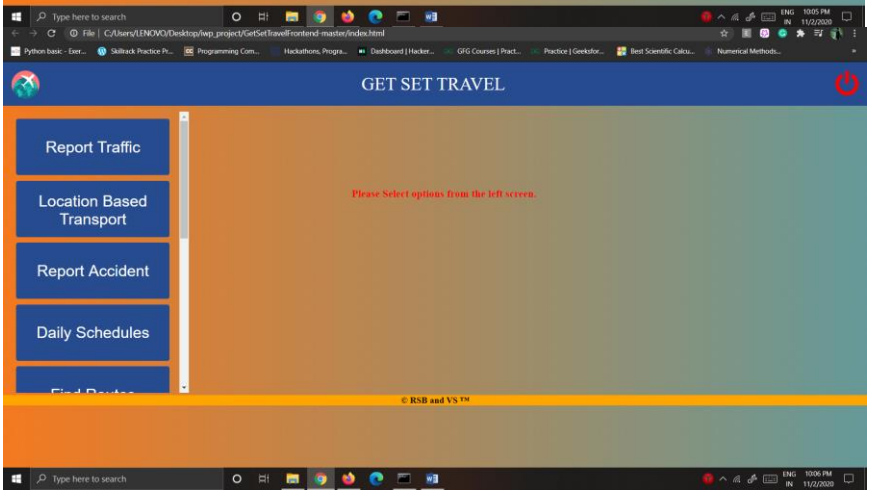
3) After logging in the user is sent to the landing page and on left side user can see options such as report accident,report traffic,location based transport,daily schedules and find route.

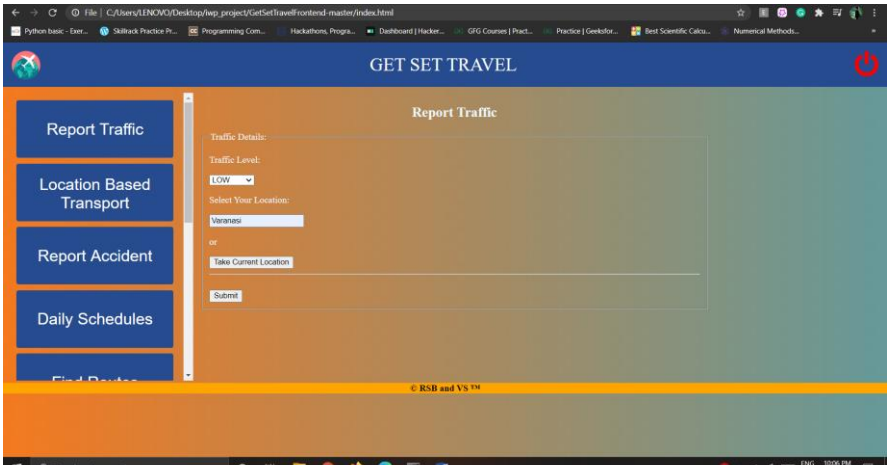
- ***There will be five models under the proposed model for its efficient working which are-***
- ***Traffic congestion Reporting module*** – This module will tell us about the data where the traffic is congested whenever user is stuck in traffic as his input
- ***Accident Reporting module*** – This software allows the user to send requests for medical assistance in case if any road accidents happen by asking about his location and the level of emergency .
- ***Daily transport schedules module***-This module will provide the daily transport schedules of buses , trains and metro to the users which will be in according to the various transport organizations.
- ***Location wise transport module-*** This module will take input as current location and destination and allows any verified user to travel at a particular instant.
- ***Route module-*** In find route user can find route to two different places including types of terrains on their path.
- ***Weather forecast module-*** It will tell the weather report of a place with an weather API.
- ***Feedback module-*** It gives customer a chance to give feedback of their journey so that future customer can review the tour before going on same route.

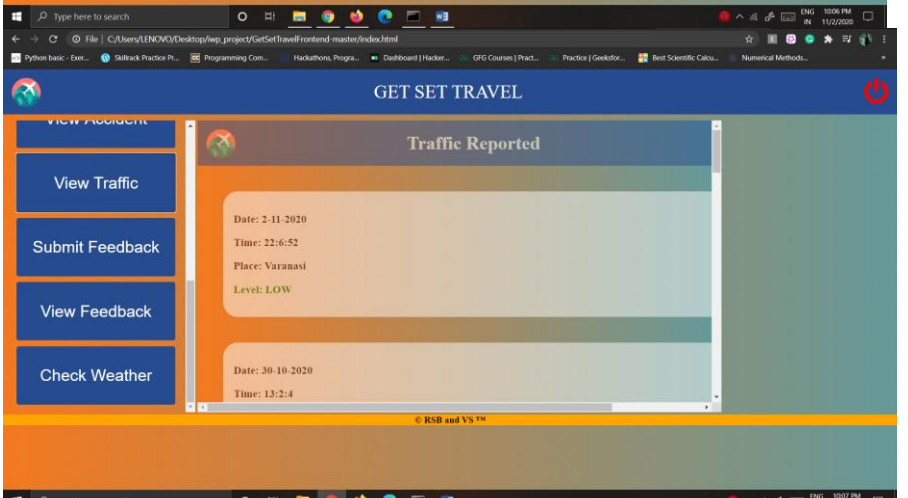
Results

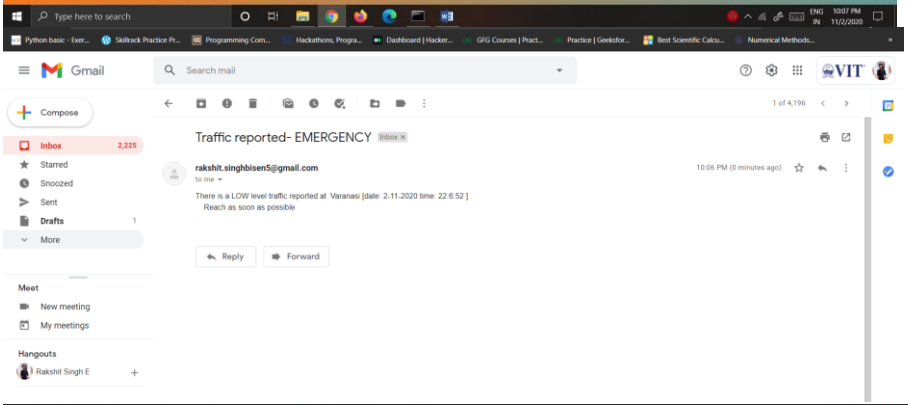

Registration Info
Name
Aadhar
Email
Password
Submit

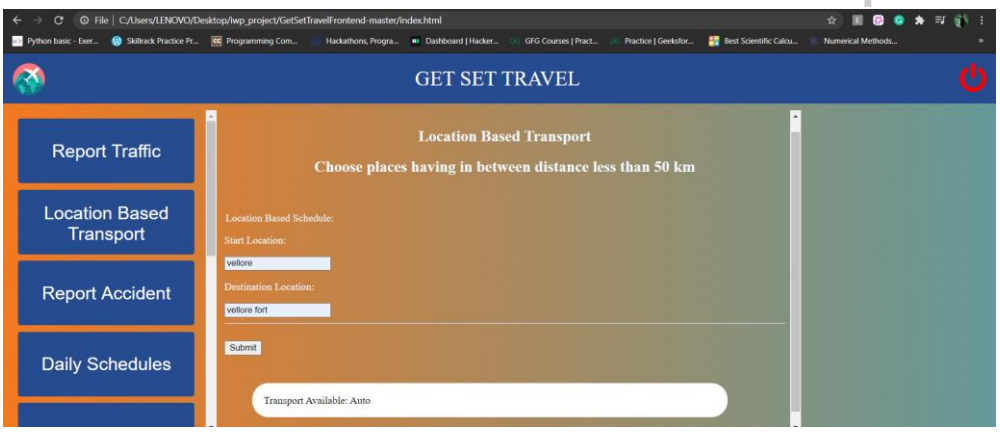

Login
Email
Password
LogIn
Not a member ? Sign Up

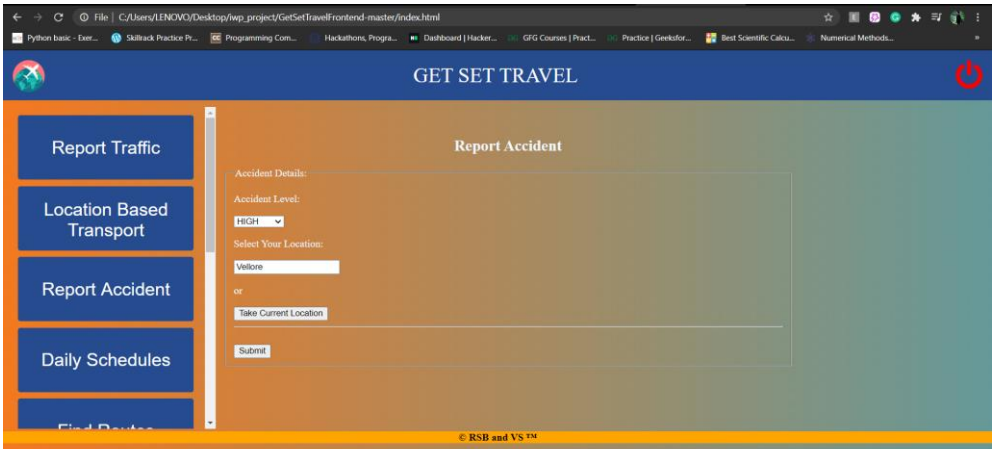

GET SET TRAVEL
Report Traffic
Location Based Transport
Report Accident
Daily Schedules
Please Select options from the left screen.
© RSB and VS ™

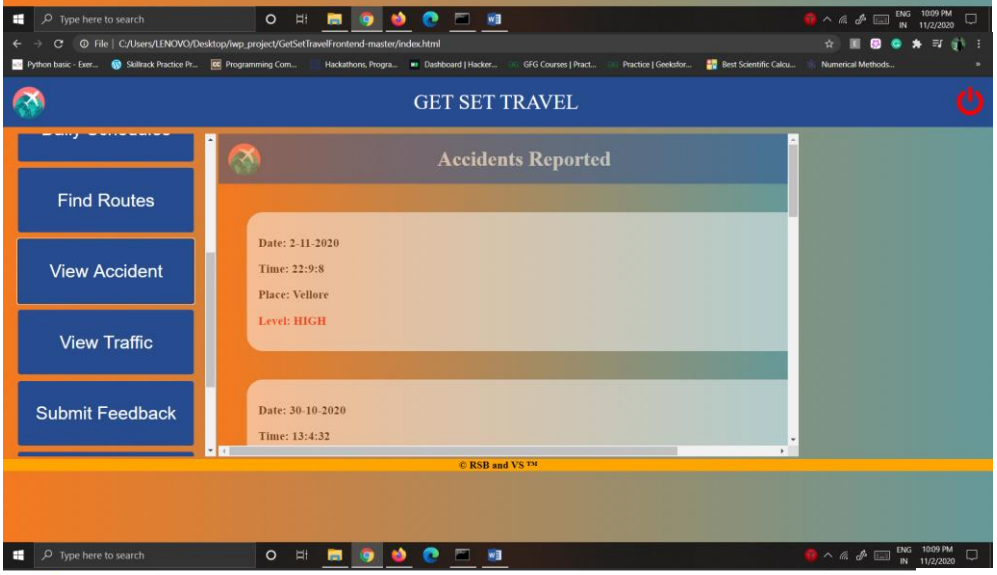

GET SET TRAVEL
Report Traffic
Location Based Transport
Report Accident
Daily Schedules
Report Traffic
Traffic Details:
Traffic Level:
LOW
Select Your Location:
Varanasi
or
Take Current Location
Submit
© RSB and VS ™

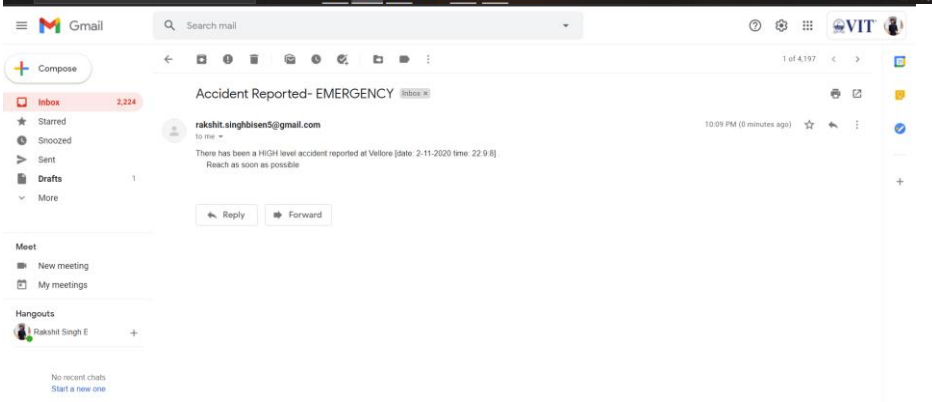

GET SET TRAVEL
View Traffic
Submit Feedback
View Feedback
Check Weather
Traffic Reported
Date: 2-11-2020
Time: 22:6:52
Place: Varanasi
Level: LOW
Date: 30-10-2020
Time: 13:2:4
© RSB and VS ™

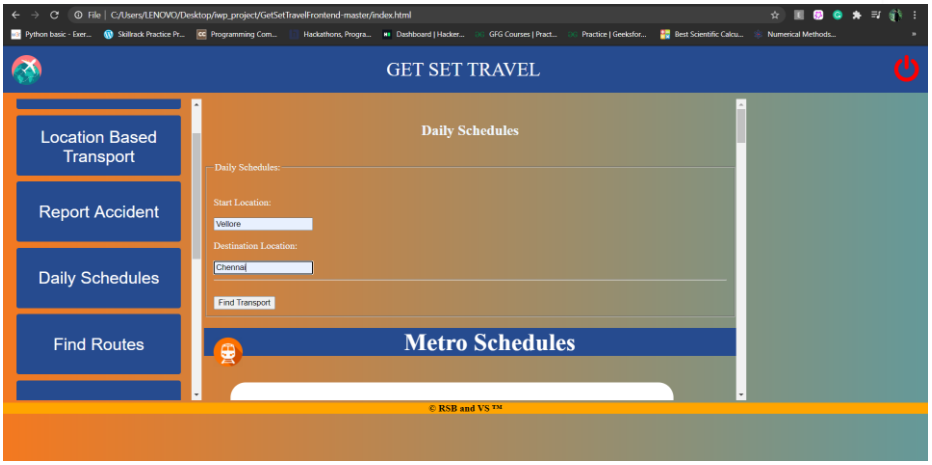

Gmail
Search mail
Compose
Inbox 2,225
Starred
Snoozed
Sent
Drafts 1
More
Meet
New meeting
My meetings
Hangouts
Traffic reported- EMERGENCY
rakshit.singhbisen5@gmail.com
10:06 PM (0 minutes ago)
There is a LOW level traffic reported at Varanasi [date: 2-11-2020 time: 22:6:52]
Reach as soon as possible
Reply
Forward

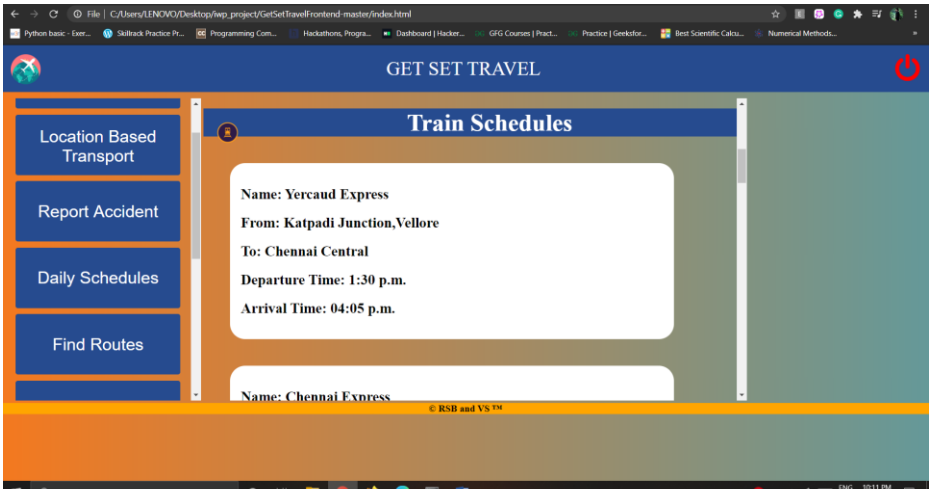

GET SET TRAVEL
Report Traffic
Location Based Transport
Report Accident
Daily Schedules
Location Based Transport
Choose places having in between distance less than 50 km
Location Based Schedule:
Start Location:
vellore
Destination Location:
vellore fort
Submit
Transport Available: Auto

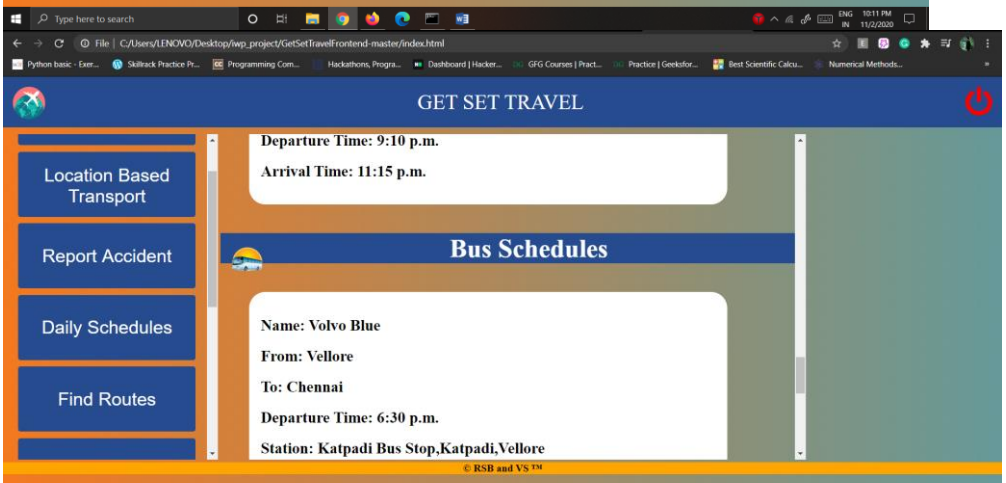

GET SET TRAVEL
Report Traffic
Location Based Transport
Report Accident
Daily Schedules
Report Accident
Accident Details:
Accident Level:
HIGH
Select Your Location:
Vellore
or
Take Current Location
Submit
© RSB and VS ™

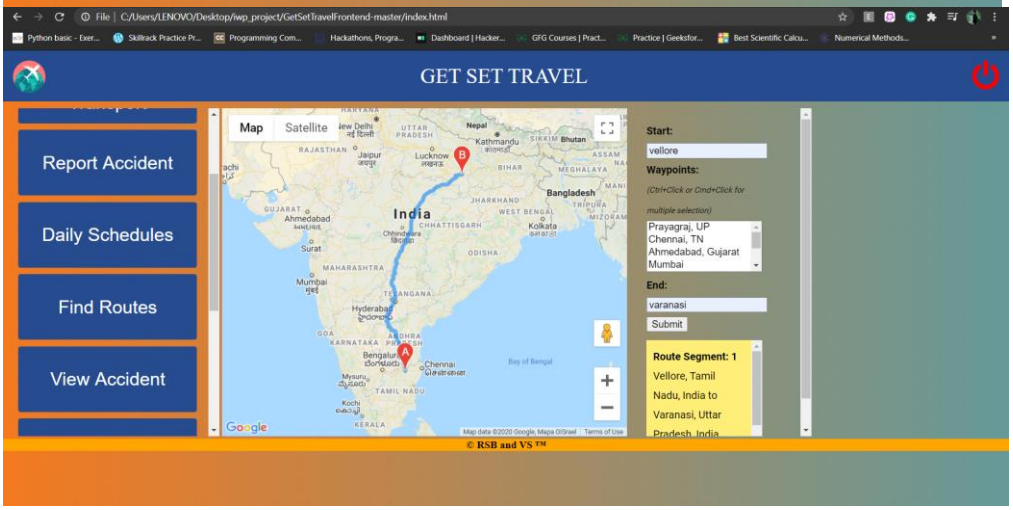

GET SET TRAVEL
Find Routes
View Accident
View Traffic
Submit Feedback
Accidents Reported
Date: 2-11-2020
Time: 22:9:8
Place: Vellore
Level: HIGH
Date: 30-10-2020
Time: 13:4:32
© RSB and VS ™

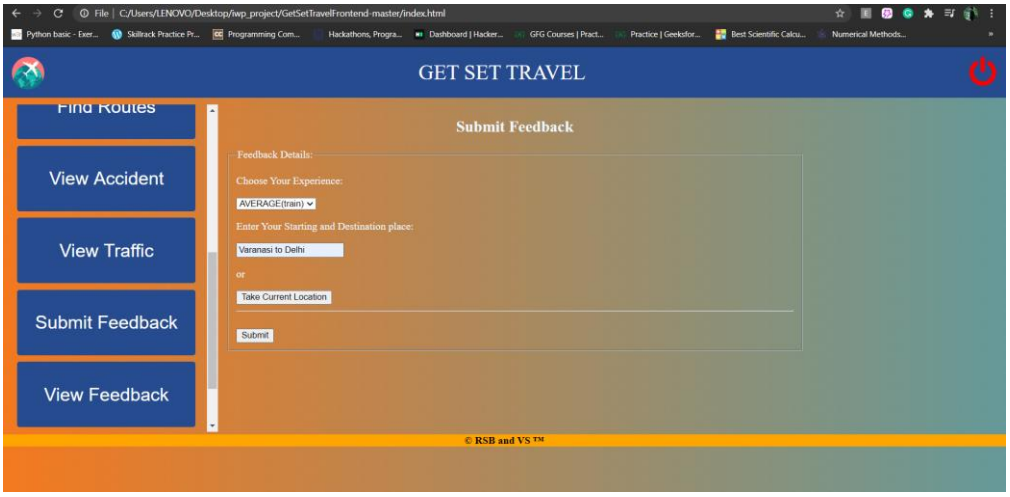

Gmail
Compose
Inbox 2,224
Starred
Snoozed
Sent
Drafts
More
Accident Reported- EMERGENCY
rakshit.singhbisen5@gmail.com
There has been a HIGH level accident reported at Vellore [date: 2-11-2020 time: 22:9:8]
Reach as soon as possible
Reply
Forward
Meet
New meeting
My meetings
Hangouts

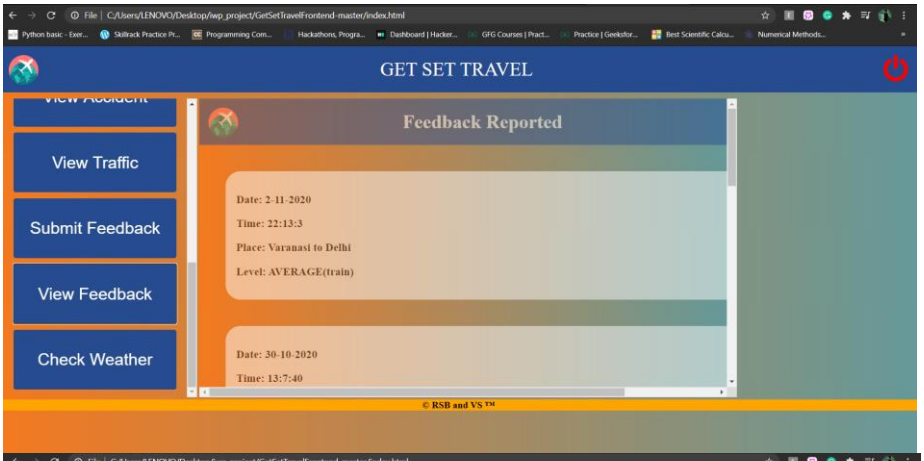

GET SET TRAVEL
Location Based Transport
Report Accident
Daily Schedules
Find Routes
Daily Schedules
Daily Schedules:
Start Location:
Vellore
Destination Location:
Chennai
Find Transport
Metro Schedules
© RSB and VS ™

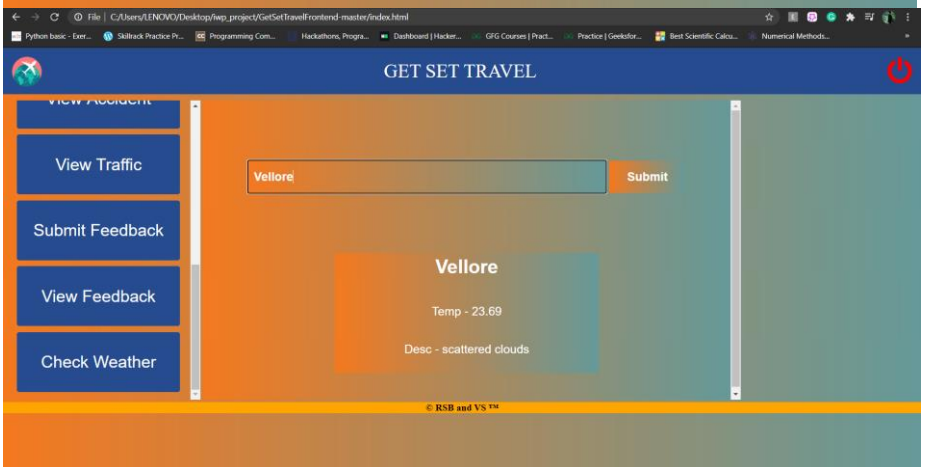

GET SET TRAVEL
Location Based Transport
Report Accident
Daily Schedules
Find Routes
Train Schedules
Name: Yercaud Express
From: Katpadi Junction,Vellore
To: Chennai Central
Departure Time: 1:30 p.m.
Arrival Time: 04:05 p.m.
Name: Chennai Express
© RSB and VS ™

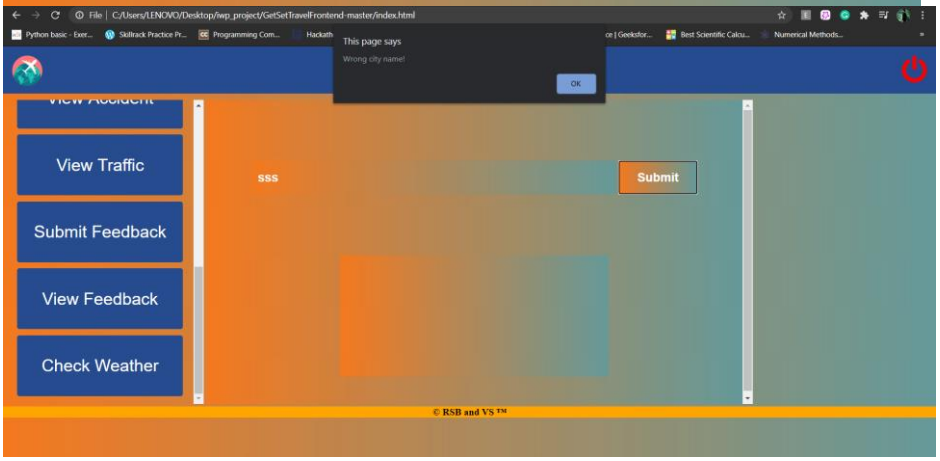

GET SET TRAVEL
Location Based Transport
Report Accident
Daily Schedules
Find Routes
Departure Time: 9:10 p.m.
Arrival Time: 11:15 p.m.
Bus Schedules
Name: Volvo Blue
From: Vellore
To: Chennai
Departure Time: 6:30 p.m.
Station: Katpadi Bus Stop,Katpadi,Vellore
© RSB and VS ™


GET SET TRAVEL
Report Accident
Daily Schedules
Find Routes
View Accident
Map
Satellite
India
Start:
vellore
Waypoints:
(Ctrl+Click or Cmd+Click for multiple selection)
Prayagraj, UP
Chennai, TN
Ahmedabad, Gujarat
Mumbai
End:
varanasi
Submit
Route Segment: 1
Vellore, Tamil Nadu, India to Varanasi, Uttar Pradesh, India
© RSB and VS ™


GET SET TRAVEL
Find Routes
View Accident
View Traffic
Submit Feedback
View Feedback
Submit Feedback
Feedback Details:
Choose Your Experience:
AVERAGE(train)
Enter Your Starting and Destination place:
Varanasi to Delhi
or
Take Current Location
Submit
© RSB and VS ™


GET SET TRAVEL
View Traffic
Submit Feedback
View Feedback
Check Weather
Feedback Reported
Date: 2-11-2020
Time: 22:13:3
Place: Varanasi to Delhi
Level: AVERAGE(train)
Date: 30-10-2020
Time: 13:7:40
© RSB and VS ™


GET SET TRAVEL
View Traffic
Submit Feedback
View Feedback
Check Weather
Vellore
Submit
Vellore
Temp - 23.69
Desc - scattered clouds
© RSB and VS ™


This page says
Wrong city name!
OK
View Traffic
Submit Feedback
View Feedback
Check Weather
sss
Submit
© RSB and VS ™

CONCLUSION

It is software which will guide people to travel across different places. The aspects include information about public transportation options, schedules, fares, real time location of the vehicle will be provided. Based upon the experiences of the person the best mode of transport to travel to a particular place would be identified and shown as the priority mode of transportation to be opted by the future customers. Also it will focus on the mishap that happens with us during the journey. Here one can complain regarding the traffic jams and can also lend a helping hand towards the unknown by initiating an ambulance service for the accident on the way

Future Work

1) In this project we can further extend/ create database, which could be authorized and governed by Government.
2) We can approach hospitals and police stations for their cooperation with our website. We can create a software to verify that the report of traffic or accident was legitimate by a real user or fraud, before sending mail to police and hospitals
3) We can create a IOT devices so that more clear and exact description of places will be provided
4) We can embed a chat bot to clear the enquiries and doubts of our customer
5) We can give Discount Coupons for other app and websites like for ticket booking or online food order on the basis of the loyalty points of our customers.
6) We can make GUI more interactive.

References

Razu Ahmed, M., Arifa Khatun, M., Asraf Ali, M., & Sundaraj, K. (2018). A literature review on NoSQL database for big data processing. International Journal of Engineering & Technology, 7(2), 902. https://doi.org/10.14419/ijet.v7i2.12113

Bhardwaj, P., & Kumar, D. D. (2018). A Review Paper on MERN Stack for Web Development. A Review Paper on MERN Stack for Web Development, 6(4), 5. http://www.ijircce.com/upload/2018/april/67_A%20Review.pdf

Kontogianni, A. (2020, February 3). Smart tourism: State of the art and literature review for the last six years. ScienceDirect. https://www.sciencedirect.com/science/article/pii/S2590005620300059

Security Implications for Json web Token Used in MERN Stack for Developing E-CommerceWeb Application. (2020). Regular, 10(1), 39–45. https://doi.org/10.35940/ijeat.a1663.1010120

Sawsan Alshattnawi" Building Mobile Tourist Guide Applications using DifferentDevelopment Mobile Platforms" Jordan, Irbid, Yarmouk University, http://www.sersc.orgjournals/IJAST/ vol54/ 2.

Li Liu; Yanfang Jing "Android city tour guide system based on Web service" 2nd International Conference on Consumer Electronics, Communications and Networks(CECNet),pp.3118-3121, 2012, DOI:10.1109/CECNet.2012.6201621.

Alexander Smirnov; Alexey Kashevnik; Andrew Ponomarev; Maksim Shchekotov; Kirill Kulakov "Application for eTourism: Intelligent Mobile Tourist Guide" IIAI 4th International Congress on Advanced Applied Informatics, pp. 40 - 45, 2015, DOI: 10.1109/IIAI-AAI.2015.190.

Hamzah Alghamdi; Shiai Zhu; Abdulmotaleb El Saddik "ETourism:Mobile Dynamic Trip Planner" IEEE International Symposium on Multimedia(ISM),pp.185-188, 2016, DOI:10.1109/ ISM.2016. 0044.

Dadape Jinendra R. Jadhav Bhagyashri R. Gaidhani Pranav Y.Vyavahare Seema U. Achaliya Parag N. "Smart Travel Guide: Application for Android Mobile", http://www.ijecscse.org/papers/ SpecialIssue/comp2/171

Lalita R. Pawar1, Sarvesh S. Patwardhan "Problems & Suggestions for Android City Tour Guide System Based on Web Services for Mumbai" http://ijarcet.org/wpcontent/uploads/IJARCET-VOL-4-ISSUE-6-2668-2672

APENDIX

CODING

Server

```
var nodemailer = require('nodemailer');
var smtpTransport = require('nodemailer-smtp-transport');
let express = require('express')
let bodyParser = require('body-parser')
let geocoding = new require('reverse-geocoding');
let geo=new require('reverse-geocoding-google')
let cors=require('cors')

let { mongoose } = require('./db/mongoose.js')

let { User } = require('./models/user.js')
let { TrafficCom } = require('./models/trafficcom.js')
let { AccidentCom } = require('./models/accidentcom.js')
let { DailySchedules } = require('./models/dailyschedules.js')
let { Ambulance } = require('./models/ambulance.js')
let { BusSchedules} = require('./models/bus.js')
let { TrainSchedules} = require('./models/train.js')
let { Feedback }= require('./models/feedback.js')

let app = express()
let port = process.env.PORT || 3000
```

```
app.use(bodyParser.json())
app.use(cors())
console.log("Server started")

app.post('/rgeo',(req,res)=>{
    let lat=req.body.lat
    let long=req.body.long
    let config={
        'latitude':lat,
        'longitude':long,
        'key':'AIzaSyCEF1MgrRBaktN47f3Xf_sb1POSHnDunY0'
    }

    geo.location(config, function (err, data){
        res.send(data.results[0].formatted_address)
    });
})

app.get('/users', (req, res) => {
    User.find().then((users) => {
        res.send(users)
    }).catch((err) => {
        res.status(400).send(err)
    })
})

app.post('/users', (req, res) => {
    console.log(req.body)

    let user = new User({
        name:req.body.name,
        email:req.body.email,
        aadhar:req.body.aadhar,
        password:req.body.password,
        user_type:req.body.user_type
    })

    user.save().then((doc) => {
        res.send(doc)
    }).catch((err) => {
        res.status(400).send(err)
    })
})

app.get('/traffic', (req, res) => {
    TrafficCom.find().then((traffic) => {
        res.send(traffic)
    }).catch((err) => {
        res.status(400).send(err)
```

```
  })
})

app.post('/traffic', (req, res) => {
   console.log(req.body)

   let traffic = new TrafficCom({
      date:req.body.date,
      time:req.body.time,
      place:req.body.place,
      level:req.body.level
   })

   traffic.save().then((doc) => {
      res.send(doc)
   }).catch((err) => {
      res.status(400).send(err)
   })
   var content=req.body.place
   var content1=req.body.date
   var content2=req.body.time
   var content3=req.body.level

   var transporter = nodemailer.createTransport(smtpTransport({
   service: 'gmail',
   host: 'smtp.gmail.com',
   auth: {
     user: 'rakshit.singhbisen5@gmail.com',
     pass: 'rakshit@511'
   }
 }));

  var mailOptions = {
   from: 'rakshit.singhbisen5@gmail.com',
   to: 'rakshitsingh.bisen2018@vitstudent.ac.in',
   subject: 'Traffic reported- EMERGENCY',

   text: `There is a ${content3} level traffic reported at  ${content} [date: ${content1} time: ${content2}
] .
   Reach as soon as possible`
  };

  transporter.sendMail(mailOptions, function(error, info){
   if (error) {
     console.log(error);
   } else {
     console.log('Email sent: ' + info.response);

   }
```

```
  });
})



app.get('/feed', (req, res) => {
   Feedback.find().then((feed) => {
      res.send(feed)
   }).catch((err) => {
      res.status(400).send(err)
   })
})

app.post('/feed', (req, res) => {
   console.log(req.body)

   let feed = new Feedback({
      date:req.body.date,
      time:req.body.time,
      place:req.body.place,
      level:req.body.level
   })

   feed.save().then((doc) => {
      res.send(doc)
   }).catch((err) => {
      res.status(400).send(err)
   })
})



app.get('/accident', (req, res) => {
   AccidentCom.find().then((accident) => {
      res.send(accident)
   }).catch((err) => {
      res.status(400).send(err)
   })
})

app.post('/accident', (req, res) => {
   console.log(req.body)

   let accident = new AccidentCom({
      date:req.body.date,
      time:req.body.time,
      place:req.body.place,
      level:req.body.level
   })
```

```
accident.save().then((doc) => {
        res.send(doc)
    }).catch((err) => {
        res.status(400).send(err)
    })
    var content=req.body.place
    var content1=req.body.date
    var content2=req.body.time
    var content3=req.body.level
    var transporter = nodemailer.createTransport(smtpTransport({
    service: 'gmail',
    host: 'smtp.gmail.com',
    auth: {
      user: 'rakshit.singhbisen5@gmail.com',
      pass: 'rakshit@511'
    }
  }));

  var mailOptions = {
    from: 'rakshit.singhbisen5@gmail.com',
    to: 'rakshitsingh.bisen2018@vitstudent.ac.in',
    subject: 'Accident Reported- EMERGENCY',
    text: `There has been a ${content3} level accident reported at ${content} [date: ${content1} time:
${content2}] .
     Reach as soon as possible`
  };

  transporter.sendMail(mailOptions, function(error, info){
    if (error) {
      console.log(error);
    } else {
      console.log('Email sent: ' + info.response);

    }
  });
})


app.get('/daily', (req, res) => {
    DailySchedules.find().then((daily) => {
        res.send(daily)
    }).catch((err) => {
        res.status(400).send(err)
    })
})
```

```
app.post('/daily', (req, res) => {
    let daily = new DailySchedules({
        date:req.body.date,
        bus:req.body.bus,
        train:req.body.train,
        metro:req.body.metro
    })

    daily.save().then((doc) => {
        res.send(doc)
    }).catch((err) => {
        res.status(400).send(err)
    })
})

app.post('/bus', (req,res)=> {
    let busSched = new BusSchedules({
        bus:req.body.bus
    })
    busSched.save().then((doc)=>{
        res.send(doc)
    }).catch((err)=>{
        res.status(400).send(err)
    })
})

app.get('/bus', (req, res) => {
    BusSchedules.find().then((daily) => {
        res.send(daily)
    }).catch((err) => {
        res.status(400).send(err)
    })
})

app.post('/train', (req,res)=> {
    let trainSched = new TrainSchedules({
        train:req.body.train
    })
    trainSched.save().then((doc)=>{
        res.send(doc)
    }).catch((err)=>{
        res.status(400).send(err)
    })
})

app.get('/train', (req, res) => {
    TrainSchedules.find().then((daily) => {
        res.send(daily)
```

```
  }).catch((err) => {
      res.status(400).send(err)
  })
})

app.listen(port)
```

Signup page

```
<!DOCTYPE html>
<html lang="en">

<head>
    <!-- Title Page-->
    <title>SignUp GetSetTravel</title>

    <!-- Icons font CSS-->
    <link href="vendor/mdi-font/css/material-design-iconic-font.min.css" rel="stylesheet" media="all">
    <link href="vendor/font-awesome-4.7/css/font-awesome.min.css" rel="stylesheet" media="all">
    <!-- Font special for pages-->
    <link
href="https://fonts.googleapis.com/css?family=Poppins:100,100i,200,200i,300,300i,400,400i,500,500
i,600,600i,700,700i,800,800i,900,900i" rel="stylesheet">

    <!-- Main CSS-->
    <link href="css/main.css" rel="stylesheet" media="all">
</head>

<body>
    <div class="page-wrapper bg-gra-01 p-t-180 p-b-100 font-poppins">
        <div class="wrapper wrapper--w780">
            <div class="card card-3">
                <div class="card-heading"></div>
                <div class="card-body">
                    <h2 class="title">Registration Info</h2>
                    <!-- <form> -->
                        <div class="input-group">
                            <input id="name" class="input--style-3" type="text" placeholder="Name"
name="name">
                        </div>
                        <div class="input-group">
                            <input id="aadhar"class="input--style-3" type="text" placeholder="Aadhar"
name="adhar">
                            <!-- <i class="zmdi zmdi-calendar-note input-icon js-btn-calendar"></i> -->
                        </div>

                        <div class="input-group">
                            <input id="email"class="input--style-3" type="email" placeholder="Email"
name="email">
                        </div>
```

```
 <div class="input-group">
                    <input id="password"class="input--style-3" type="password" placeholder="Password"
name="password">
                  </div>
                  <div class="p-t-10">
                    <button id="reg_b" class="btn btn--pill btn--green" type="submit">Submit</button>
                  </div>
                <!-- </form> -->
              </div>
            </div>
          </div>
    </div>


    <script type="text/javascript" src="signup.js"></script>
    <script src="https://unpkg.com/axios/dist/axios.min.js"></script>

</body>
</html>
```

Login page

```
<!DOCTYPE html>
<html lang="en">

<head>
    <!-- Title Page-->
    <title>GetSetTravel Login</title>

    <!-- Icons font CSS-->
    <link href="vendor/mdi-font/css/material-design-iconic-font.min.css" rel="stylesheet" media="all">
    <link href="vendor/font-awesome-4.7/css/font-awesome.min.css" rel="stylesheet" media="all">
    <!-- Font special for pages-->
    <link
href="https://fonts.googleapis.com/css?family=Poppins:100,100i,200,200i,300,300i,400,400i,500,500
i,600,600i,700,700i,800,800i,900,900i" rel="stylesheet">


    <!-- Main CSS-->
    <link href="css/main.css" rel="stylesheet" media="all">
</head>

<body>
    <!-- <div class="bg-gra-01 font-poppins" id = "siteName" >Get Set Travel</div> -->
    <div class="page-wrapper bg-gra-01 p-t-180 p-b-100 font-poppins">

        <div class="wrapper wrapper--w780">

            <div class="card card-3">
                <div class="card-heading"></div>
                <div class="card-body">
```

```
            <h2 class="title">Login</h2>
            <!-- <form method="POST"> -->

              <div class="input-group">
                <input id="email" class="input--style-3" type="email" placeholder="Email"
name="email">
              </div>
              <div class="input-group">
                <input id="password" class="input--style-3" type="password"
placeholder="Password" name="password">
              </div>
              <div class="p-t-10">
                <button id = "submit" class="btn btn--pill btn--green" type="submit">Login</button>
              </div>
              <div class="signuplink" >
                <br>
                <p>Not a member ? <a href="../SignupPage/index.html"> Sign Up</a></p>
              </div>

            <!-- </form> -->
          </div>
        </div>
      </div>
    </div>
    <script src="https://unpkg.com/axios/dist/axios.min.js"></script>


    <script type="text/javascript" src="login.js"></script>


</body>
</html>
```

Report Accident Page

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <!-- <meta http-equiv="X-UA-Compatible" content="ie=edge"> -->
    <link rel="stylesheet" href="report_traffic.css">
    <title>Document</title>
    <style>
        .sty{
            color: aliceblue;
        }

    </style>
</head>
```

```
<body>
  <br>
  <h2 style="text-align: center;" class="sty">Report Accident</h2>

  <!-- <form > -->
    <fieldset>
        <legend class="sty">Accident Details:</legend>
        <p class="sty">Accident Level:</p>
        <select id = "level">
            <option value="LOW">LOW</option>
            <option value="MEDIUM">MEDIUM</option>
            <option value="HIGH">HIGH</option>
        </select>          <br>
        <p class="sty">Select Your Location:</p>
        <input id="location" type="text" name="location" placeholder="Vellore">
        <br>
        <p class="sty">or</p>
        <button id ="geoLocation">Take Current Location</button>
        <br>
        <hr>
        <br>
        <input id="report"class ="button" type="submit" value="Submit">
    </fieldset>

  <!-- </form> -->
  <script type = "text/javascript" src="report_accident.js"></script>
  <script src="https://unpkg.com/axios/dist/axios.min.js"></script>

</body>
</html>
```

View Accident

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <!-- <meta http-equiv="X-UA-Compatible" content="ie=edge"> -->
  <title>GetSetTravel(Hospital)</title>
  <meta http-equiv="refresh" content="10"/>

  <link rel="stylesheet" href="hospital.css">
  <link rel="icon" href="travel.png">

</head>
<body>
    <div id = "heading">
        <img src="travel.png" alt="icon" style="float: left; height:50px; width:50px; padding-left:
15px; padding-top: 15px">
        <h1>Accidents Reported</h1>
        <!-- <button id = "logout">LOGOUT</button> -->
        <i class="fa fa-power-off" style="font-size:48px;color:red; float: right; padding: 15px;"></i>
```

```
 </div>
  <!-- <div class = "container">
     <h3>Date: </h3>
     <h3>Time: </h3>
     <h3>Place: </h3>
     <h3>Level: </h3>
  </div> -->
  <script src="https://unpkg.com/axios/dist/axios.min.js"></script>
  <script type="text/javascript" src = "hospital.js" ></script>

</body>
</html>
```

REPORT TRAFFIC

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <meta http-equiv="X-UA-Compatible" content="ie=edge">
  <!-- <link rel="stylesheet" href="report_traffic.css"> -->
  <title>Document</title>
  <style>
     .sty{
        color: aliceblue;
     }
  </style>
</head>
<body>

  <h2 class = "sty"  style="text-align: center;">Report Traffic</h2>
  <!-- <form > -->
     <fieldset>
          <legend class = "sty">Traffic Details:</legend>
          <p class = "sty">Traffic Level:</p>

          <select id = "level">
               <option value="LOW">LOW</option>
               <option value="MEDIUM">MEDIUM</option>
               <option value="HIGH">HIGH</option>
          </select>
          <!-- <input id="level" type="text" name="level" placeholder="Medium"> -->
          <br>
          <p class = "sty">Select Your Location:</p>
          <input id="location" type="text" name="location" placeholder="Vellore">
          <br>
          <p class = "sty">or</p>
          <button id= "geoLocation">Take Current Location</button>
          <br>
```

```
<hr>
            <br>
            <input id="report" class ="button" type="submit" value="Submit">
        </fieldset>

    <!-- </form> -->
    <script type="text/javascript" src="report_traffic.js"></script>
    <script src="https://unpkg.com/axios/dist/axios.min.js"></script>

</body>
</html>
```

VIEW TRAFFIC

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <meta http-equiv="refresh" content="10"/>
    <title>GetSetTravel(Police)</title>
    <link rel="stylesheet" href="police.css">
    <link rel="icon" href="travel.png">

</head>
<body>
    <div id = "heading">
        <img src="travel.png" alt="icon" style="float: left; height:50px; width:50px; padding-left: 15px;
padding-top: 15px">
        <h1>Traffic Reported</h1>
        <!-- <button id = "logout">LOGOUT</button> -->
        <i class="fa fa-power-off" style="font-size:48px;color:red; float: right; padding: 15px;"></i>

    </div>
    <!-- <div class = "container">
        <h3>Date: </h3>
        <h3>Time: </h3>
        <h3>Place: </h3>
        <h3>Level: </h3>
    </div> -->
    <script src="https://unpkg.com/axios/dist/axios.min.js"></script>
    <script type="text/javascript" src = "police.js" ></script>
</body>
</html>
```

Report FEEDBACK

```
<!DOCTYPE html>
<html lang="en">
<head>
   <meta charset="UTF-8">
   <meta name="viewport" content="width=device-width, initial-scale=1.0">
   <meta http-equiv="X-UA-Compatible" content="ie=edge">
   <!-- <link rel="stylesheet" href="report_traffic.css"> -->
   <title>Document</title>
   <style>
      .sty{
         color: aliceblue;
      }
   </style>
</head>
<body>

   <h2 class = "sty"  style="text-align: center;">Submit Feedback</h2>
   <!-- <form > -->
      <fieldset>
            <legend class = "sty">Feedback Details:</legend>
            <p class = "sty">Choose Your Experience:</p>

            <select id = "level">
                  <option value="GOOD(cab)">GOOD(cab)</option>
                  <option value="GOOD(bus)">GOOD(bus)</option>
                  <option value="GOOD(train)">GOOD(train)</option>
                  <option value="AVERAGE(cab)">AVERAGE(cab)</option>
                  <option value="AVERAGE(bus)">AVERAGE(bus)</option>
                  <option value="AVERAGE(train)">AVERAGE(train)</option>
                  <option value="BAD(cab)">BAD(cab)</option>
                  <option value="BAD(bus)">BAD(bus)</option>
                  <option value="BAD(train)">BAD(train)</option>
            </select>
            <!-- <input id="level" type="text" name="level" placeholder="Medium"> -->
            <br>
            <p class = "sty">Enter Your Starting and Destination place:</p>
            <input id="location" type="text" name="location" placeholder="Vellore to Chennai">
            <br>
            <p class = "sty">or</p>
             <button id= "geoLocation">Take Current Location</button>
            <br>
            <hr>
            <br>
            <input id="report" class ="button" type="submit" value="Submit">
      </fieldset>
```

```
 <!-- </form> -->
   <script type="text/javascript" src="give_feed.js"></script>
   <script src="https://unpkg.com/axios/dist/axios.min.js"></script>

</body>
</html>
```

VIEW FEEDBACK

```
<!DOCTYPE html>
<html lang="en">
<head>
   <meta charset="UTF-8">
   <meta name="viewport" content="width=device-width, initial-scale=1.0">
   <meta http-equiv="refresh" content="10"/>
   <title>View Feedback</title>
   <link rel="stylesheet" href="feedb.css">
   <link rel="icon" href="travel.png">

</head>
<body>
   <div id = "heading">
      <img src="travel.png" alt="icon" style="float: left; height:50px; width:50px; padding-left: 15px;
padding-top: 15px">
      <h1>Feedback Reported</h1>
      <!-- <button id = "logout">LOGOUT</button> -->
      <i class="fa fa-power-off" style="font-size:48px;color:red; float: right; padding: 15px;"></i>

   </div>
   <!-- <div class = "container">
      <h3>Date: </h3>
      <h3>Time: </h3>
      <h3>Place: </h3>
      <h3>Level: </h3>
   </div> -->
   <script src="https://unpkg.com/axios/dist/axios.min.js"></script>
   <script type="text/javascript" src = "feedb.js" ></script>
</body>
</html>
```

Location Based TRANSPORT

```
<!DOCTYPE html>
<html lang="en">
<head>
   <meta charset="UTF-8">
   <meta name="viewport" content="width=device-width, initial-scale=1.0">
   <meta http-equiv="X-UA-Compatible" content="ie=edge">
   <title>Document</title>
   <style>
         .sty{
            color: aliceblue;
         }
```

```
        .container{
    width: 700px;
    margin: 5%;
    padding: 2%;
    height: 20%;
    background: white;
    border-radius: 25px;
    box-shadow: #f4791f;
}
h3{
    text-align: center;
}
    </style>
    <script >
        function validateform(){

            var x = document.getElementById("start").value;
            var y= document.getElementById("end").value;
            if(x=="delhi" && y=="gurgaon"){
                document.getElementById("out").innerHTML="Transport Available: Cab / Metro / Local
Bus ";

            }
            else if(x=="BHU" && y=="lal bhadur shastri airport"){
                document.getElementById("out").innerHTML="Transport Available: Auto / Cab ";}
            else if(x=="BHU" && y=="cantt railway station"){
                document.getElementById("out").innerHTML="Transport Available: Auto / Cab /City Bus
";}
            else if(x=="vellore" && y=="vellore fort"){
                document.getElementById("out").innerHTML="Transport Available: Auto ";
            }
            else if(x=="vellore" && y=="katpadi"){
                document.getElementById("out").innerHTML="Transport Available: Auto ";
            }
            else if(x=="delhi" && y=="noida"){
                document.getElementById("out").innerHTML="Transport Available: Cab / Metro  ";
            }
            else{
                alert("Please enter Locations having distance less than 50 km")
            }
            return false;


        }
      </script>

</head>
<body>
```

```
 <br>
     <h2 class="sty" style="text-align: center;">Location Based Transport</h2>
     <h2 class="sty" style="text-align: center;">Choose places having in between distance less than 50
km</h2>
     <!-- <form > -->
        <br>
        <form name="my form" onsubmit="return validateform();" >
           <br>
              <legend class="sty">Location Based Schedule:</legend>
              <p class="sty">Start Location:</p>
              <input id="start" type="text" name="start" placeholder="Start">
              <br>
              <p class="sty">Destination Location:</p>
              <input id="end" type="text" name="end" placeholder="Destination">
              <br>
              <hr>
              <br>
              <input  type="submit" ><br>

        </form>
        <p class = "container" id="out"><center></center><p>

</body>
</html>
```

Check Weather

```
<!DOCTYPE html>
<html lang="en">
<head>
 <meta charset="UTF-8">
 <meta name="viewport" content="width=device-width, initial-scale=1.0">
 <meta http-equiv="X-UA-Compatible" content="ie=edge">
 <title>OpenWeatherMap Api</title>
 <link rel="stylesheet" href="style.css">
</head>
<body>
 <!-- <div class="main">
  <div class="header">
   <h1>OpenWeatherMap API</h1>
   <p>Enter any city name in the input box below to get the data</p>
  </div> -->

  <div class="input ">
   <input type="text" placeholder="Enter the city" class="input_text">
   <input type="submit" value="Submit" class="submit">
  </div>
 </div>

 <div class="container">
  <div class="card">
   <h1 class="name" id="name"></h1>
```

```
    <p class="temp"></p>
    <p class="clouds"></p>
    <p class="desc"></p>
   </div>
  </div>
<script src="app.js"></script>
</body>
</html>
```

ROUTE

```
<!DOCTYPE html>
<html>
  <head>
    <meta name="viewport" content="initial-scale=1.0, user-scalable=no">
    <meta charset="utf-8">
    <title>Waypoints in Directions</title>
    <style>
      #right-panel {
        font-family: 'Roboto','sans-serif';
        line-height: 30px;
        padding-left: 10px;
      }

      #right-panel select, #right-panel input {
        font-size: 15px;
      }

      #right-panel select {
        width: 100%;
      }

      #right-panel i {
        font-size: 12px;
      }
      html, body {
        height: 100%;
        margin: 0;
        padding: 0;
      }
      #map {
        height: 100%;
        float: left;
        width: 70%;
        height: 100%;
      }
      #right-panel {
        margin: 20px;
        border-width: 2px;
        width: 20%;
```

```
      height: 400px;
      float: left;
      text-align: left;
      padding-top: 0;
     }
     #directions-panel {
      margin-top: 10px;
      background-color: #FFEE77;
      padding: 10px;
      overflow: scroll;
      height: 174px;
     }
    </style>
   </head>
   <body>
    <div id="map"></div>
    <div id="right-panel">
    <div>
    <b>Start:</b>
    <!-- <select id="start">
     <option value="Halifax, NS">Halifax, NS</option>
     <option value="Boston, MA">Boston, MA</option>
     <option value="New York, NY">New York, NY</option>
     <option value="Miami, FL">Miami, FL</option>
    </select> -->
    <input type="text" id ="start">
    <br>
    <b>Waypoints:</b> <br>
    <i>(Ctrl+Click or Cmd+Click for multiple selection)</i> <br>
    <select multiple id="waypoints">
     <option value="montreal, quebec">Montreal, QBC</option>
     <option value="toronto, ont">Toronto, ONT</option>
     <option value="chicago, il">Chicago</option>
     <option value="winnipeg, mb">Winnipeg</option>
     <option value="fargo, nd">Fargo</option>
     <option value="calgary, ab">Calgary</option>
     <option value="spokane, wa">Spokane</option>
    </select>
    <br>
    <b>End:</b>
    <!-- <select id="end">
     <option value="Vancouver, BC">Vancouver, BC</option>
     <option value="Seattle, WA">Seattle, WA</option>
     <option value="San Francisco, CA">San Francisco, CA</option>
     <option value="Los Angeles, CA">Los Angeles, CA</option>
    </select> -->
    <input type="text" id = "end">
    <br>
     <input type="submit" id="submit">
    </div>
```

```
<div id="directions-panel"></div>
</div>
<script>
 function initMap() {
  var directionsService = new google.maps.DirectionsService;
  var directionsRenderer = new google.maps.DirectionsRenderer;
  var map = new google.maps.Map(document.getElementById('map'), {
   zoom: 6,
   center: {lat: 12.916517, lng: 79.132500}
  });
  directionsRenderer.setMap(map);

  document.getElementById('submit').addEventListener('click', function() {
   calculateAndDisplayRoute(directionsService, directionsRenderer);
  });
 }

 function calculateAndDisplayRoute(directionsService, directionsRenderer) {
  var waypts = [];
  var checkboxArray = document.getElementById('waypoints');
  for (var i = 0; i < checkboxArray.length; i++) {
   if (checkboxArray.options[i].selected) {
    waypts.push({
     location: checkboxArray[i].value,
     stopover: true
    });
   }
  }

  directionsService.route({
   origin: document.getElementById('start').value,
   destination: document.getElementById('end').value,
   waypoints: waypts,
   optimizeWaypoints: true,
   travelMode: 'DRIVING'
  }, function(response, status) {
   if (status === 'OK') {
    directionsRenderer.setDirections(response);
    var route = response.routes[0];
    var summaryPanel = document.getElementById('directions-panel');
    summaryPanel.innerHTML = '';
    // For each route, display summary information.
    for (var i = 0; i < route.legs.length; i++) {
     var routeSegment = i + 1;
     summaryPanel.innerHTML += '<b>Route Segment: ' + routeSegment +
       '</b><br>';
     summaryPanel.innerHTML += route.legs[i].start_address + ' to ';
     summaryPanel.innerHTML += route.legs[i].end_address + '<br>';
     summaryPanel.innerHTML += route.legs[i].distance.text + '<br><br>';
    }
```

```
} else {
          window.alert('Directions request failed due to ' + status);
        }
       });
     }
   </script>
   <script async defer

src="https://maps.googleapis.com/maps/api/js?key=AIzaSyCEF1MgrRBaktN47f3Xf_sb1POSHnDunY0
&callback=initMap">
   </script>
  </body>
</html>
```

NOTE-ONLY html files for frontend is uploaded and main server file for backend in order to avoid large document.