

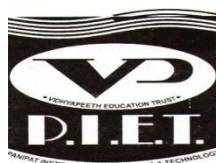
***A Project Report
on
Post Pandemic Crowd Safety***

***Submitted in partial fulfillment of the requirements
for the award of the degree of***

***Bachelor of Technology
in
Computer Science & Engineering***

***Submitted By
Pranav Taneja (2817089)
Manan Arora (2817023)
Abhay Mendiratta (2817037)***

***Under the Supervision of
Ms. Alankrita Aggarwal
(Assistant Professor)***



***Panipat Institute of Engineering & Technology, Samalkha, Panipat
Affiliated to***



***Kurukshetra University Kurukshetra, India
(2020-2021)***

DECLARATION

I hereby declare that the work presented in this project report entitled “Post Pandemic Crowd Safety “, in partial fulfilment of the requirement for the award of the degree of Bachelor of Technology in Computer Science & Engineering., submitted to Kurukshetra University, Kurukshetra, India is an authentic record of my work carried out during the period from July 2020 to Dec 2020 under the guidance of Ms Alankrita Aggarwal (Assistant Professor, C.S.E)

I hereby declare that this submission is my work and that, to the best of my knowledge and belief, it contains no material previously published or written by another person nor material which to a substantial extent has been accepted for the award of any other degree or diploma of the university or other institute of higher learning, except where due acknowledgement has been made in the text.

Signature

Name:

Roll No:

Signature

Name:

Roll No:

Signature

Name:

Roll No:

Date:

CERTIFICATE

This is to certify that the Project I report (CSE-407N) entitled “Post Pandemic Crowd Safety“ done by Pranav Taneja (2817089), Manan Arora (2817023), Abhay Mendiratta (2817037) is an authentic work carried out by them at PIET, Samalkha, Panipat under my guidance. The matter embodied in this project work has not been submitted earlier for the award of any degree or diploma to the best of my knowledge and belief.

Ms Alankrita Aggarwal

(Project Guide)

Mr Gaurav Gambhir

Mr Shekhar Singh

(Project Coordinators)

Prof. Dr S C Gupta

HOD (CSE)

PIET, Samalkha

ACKNOWLEDGEMENT

It gives us a great sense of pleasure to present the report of the B. Tech Project undertaken during B. Tech. Final Year. We owe a special debt of gratitude to Assistant Professor Ms Alankrita Aggarwal, Department of Computer Science & Engineering, PIET, Samalkha for his constant support and guidance throughout our work. Her sincerity, thoroughness and perseverance have been a constant source of inspiration for us. It is only his cognizant efforts that our endeavours have seen the light of the day.

We also take the opportunity to acknowledge the contribution of Professor Dr S C Gupta, Head, Department of Computer Science & Engineering, PIET, Samalkha for his full support and assistance during the development of the project.

We also do not like to miss the opportunity to acknowledge the contribution of all faculty members of the department for their kind assistance and cooperation during the development of our project. Last but not the least, we acknowledge our friends for their contribution to the completion of the project.

Signature:

Name :

Roll No.:

Date :

Signature:

Name :

Roll No.:

Date :

Signature:

Name :

Roll No.:

Date :

ABSTRACT

The proposed project is an innovative approach towards providing safety to the end-user i.e., **General Public** as we know we can't disrupt the economy further, we have to get back to the track and the virus is nowhere its termination hence the proposed system notifies the user about all possible harmful locations and further provides a safer route to travel via so that the user can safely or at least be at the minimum risk of contracting a virus prone zone during his/her journey.

Table of Contents

DECLARATION.....	2
CERTIFICATE.....	3
ACKNOWLEDGEMENTS.....	4
ABSTRACT.....	5
LIST OF FIGURES.....	11
LIST OF TABLES.....	14
CHAPTER 1: INTRODUCTION.....	16
1.1 The Topic of the System.....	16
1.2 Project Abstract.....	16
1.3 Purpose of the System.....	16
1.4 Target User.....	16
1.5 Topic Background.....	16
1.6 Problem Context.....	17
1.7 Rationale behind the System.....	18
1.7.1 Benifts of the System.....	18
1.8 Objectives of the System.....	20
1.8.1 Project Objectives.....	20
1.8.2 Objectives.....	20
1.9 Scope of the System.....	20
1.10 Limitation of the System.....	20
1.11 Assumptions Made.....	21
1.12 Success Criteria.....	21
1.13 Functionalities of the System.....	21
1.13.1 Core Features.....	23
1.13.2 Enhanced Features.....	23
1.14 Conclusion.....	24
CHAPTER 2: PROBLEM DESCRIPTION.....	25

2.1 Current Problem Description.....	25
2.1.1 Description of Problem Area.....	25
2.2 Problem Importance and Justification.....	25
2.3 Nature of Challenge.....	26
2.3.1 Domain Challenge.....	26
2.3.2 Technical Challenge.....	26
2.4 Feasibility Study.....	27
2.4.1 Technical Feasibility.....	28
2.4.2 Economic Feasibility.....	30
2.4.3 Schedule Feasibility.....	30
2.4.4 Operational Feasibility.....	31
2.4.5 Conclusion of Feasibility Study.....	31
2.5 Conclusion.....	31
CHAPTER 3: LITERATURE REVIEW	32
3.1 Introduction to Literature Review.....	32
3.2 Advanced Preliminary Research.....	32
3.3 Academic Research.....	32
3.3.1 Crowd Sourcing Technology	33
3.3.2 Detecting Hotspot.....	33
3.3.3 Safest Route Detection.....	33
3.3.4 Human Computer Interaction Principles.....	33
3.4 Domain Research.....	33
3.4.1 Android.....	34
3.4.2 Flutter Architecture.....	34
3.4.3 Security Permissions.....	35
3.5 Critical Evaluation of Literature Review.....	35
3.6 Conclusion.....	36

CHAPTER 4: RESEARCH METHODS.....	37
4.1.1 Primary Research.....	37
4.1.2 Questionnaire.....	37
4.1.3 Development Methodology.....	40
4.1.4 Development Plan.....	44
4.2 Conclusion.....	45
CHAPTER 5: ANALYSIS	46
5.1 Questionnaire.....	46
5.1.1 Overall Conclusion of Questionnaire.....	46
5.1.2 Overall Conclusion of Interview.....	46
CHAPTER 6: SYSTEM DESIGN.....	47
6.1 Introduction.....	47
6.2 UML Diagram.....	47
6.2.1 Use Case.....	47
CHAPTER 7: IMPLEMENTATION	51
7.1 Introduction.....	51
7.2 Tools used for Implementation.....	51
7.3 Implementation Plan.....	52
7.4 Coding.....	53
7.5 Images.....	107
7.6 Tables.....	112
CHAPTER 8: TESTING.....	114
8.1 Introduction.....	114
8.2 Need of Testing.....	114

8.3 Test Plan.....	114
8.3.1 Type of Testing.....	115
8.3.2 Pass/Fail Criteria.....	117
8.4 Criteria of Test Cases.....	117
8.5 Technical requirements for Testing.....	117
8.6 Users involved in Testing.....	118
8.7 Unit Testing.....	118
8.7.1 Black box testing.....	118
8.7.2 White box testing.....	118
8.8 Integration Testing.....	118
8.8.1 Technique used for Integration Testing.....	119
8.9 System Testing.....	119
8.10 Compatibility Testing.....	119
8.11 Usability Testing and its Justification.....	119
8.11.1 User Acceptance Testing.....	121
8.11.2 Justification for User Acceptance Testing.....	121
8.12 Documentation Testing.....	121
8.13 Conclusion.....	122
 CHAPTER 9: CRITICAL EVALUATION.....	124
9.1 Critical Evaluation.....	124
9.2 Factors of Benefit.....	124
9.3 Success Assessment.....	124
9.3.1 Degree of Success.....	124
9.3.2 Critical Appraisal.....	125

CHAPTER 10: CONCLUSION.....	126
10.1 Success Criteria.....	126
10.2 Limitations and Errors in the Developed System.....	126
10.3 Learning Experience Gathered.....	126
10.4 Conclusion.....	128
CHAPTER 11: REFERENCES.....	130
11.1 Research Papers.....	130
11.2 Books.....	130

List of Figures:

Serial No.	Fig No.	Fig Name	Page No.
1	3.1	Android Version	34
2	3.2	Flutter Architecture	35
3	4.1	V Model	41
4	6.1	Use Case Login	48
5	6.2	Use Case User	49
6	7.1	Folder Layout Overview	53
7	7.2	App Folder Layout	53
8	7.3	Data Folder Layout	54
9	7.4	Models Folder Layout	54
10	7.5	Providers Folder Layout	57
11	7.6	Repository Folder Layout	61
12	7.7	Modules Folder Layout	63
13	7.8	Login Folder Layout	63

14	7.9	Register Folder Layout	70
15	7.10	Map Folder Layout	80
16	7.11	Services Folder Layout	89
17	7.12	Server Folder Layout	91
18	7.13	Login View	107
19	7.14	Signup View	107
20	7.15	Error Detection on Fields	107
21	7.16	Final Signup View	107
22	7.17	GPS Permission Dialog	108
23	7.18	Map View with Hotspot Zones	108
24	7.19	Details of Selected Hotspot Zones	108
25	7.20	Crowded Zones	108
26	7.21	Login Web View	109
27	7.22	Signup Web View	110

28	7.23	Permission Web View	110
29	7.24	Hotspot Web View	111
30	7.25	Hotspot Detail View	111
31	7.26	Crowded Hotspot Web View	112
32	8.1	Testing	115

LIST OF TABLES:

S.No.	Table No	Table Name	Page No.
1	1.1	Tangible Benefits	18
2	1.2	Intangible Benefits	19
3	1.3	Core Features	23
4	1.4	Enhanced Features	23
5	2.1	Hardware Resources	28
7	2.2	Software Resources	28
8	2.3	Resources of Execution	29
9	4.1	Questionnaire	39
10	4.2	Development Plan	44
11	6.1	Use Cases	48
12	6.2	Use Case Description Module	49
13	7.1	Tools used for Implementation	51
14	7.2	User Table	112
15	7.3	Hotspot Table	113

16	8.1	Type of Testing	115
17	8.2	Technical requirements for Testing	117

CHAPTER 1: INTRODUCTION

1.1 Topic of the System

Title: - “A Post Pandemic Crowd Safety Provisioning Application”

1.2 Project Abstract

The proposed project is an innovative approach towards providing safety to the end user i.e. **General Public** as we know we cant disrupt the economy further, we have to get back to track and the virus is nowhere its termination hence the proposed system notifies the user about all possible harmful locations and further provides a safer route to travel via so that the cuser can safely or atleast be at the minimum risk of contacting a virus prone zone during his/her journey.

1.3 Purpose of the System

The world is going through one of the worst pandemics ever seen . After concurrent lock-downs as the government is easing out, moreand more people are heading towards the streets and are at the verge of risking their lives, hence the proposed system notifies the user about all possible harmful locations and further provides a safer route to travel via so that the cuser can safely or atleast be at the minimum risk of contacting a virus prone zone during his/her journey. The user will open the app on whichever device he/she may prefer the app will use the current location co ordintes of the user to determine where is the corona prone zone and alert the user in turn.

1.4 Target User

Any kind of system is developed with thought of a end user in mind. This project is globally usable by any person irrespective of field, any person who has to get to some place and is keen to know the current covid situation around his/her location is an ample user for this product being developed.

1.5 Topic Background

We already know Internet is the motherlord of the modern world. Everyone is interlinked with each other for last 2 decades or so. Earlier, we used to use computers and other devices but after the boom of mobile market everything has become easy to use and is within the reach of our finger tips.

In addition, with the advancement of wireless technologies, wireless networks have taken over the entire world. Nowadays, business and financial transactions can be done easily and securely, anywhere and anytime. Using the Internet, connections can be established with any devices almost anywhere in the world and can share necessary information amongst them.

With functionalities like maps, anyone can have access to facilities like knowing about a location, going to a particular place, rating and reviewing and much more.

But from past few months a serious pandemic has hit the world which has curbed everyone to their homes and even after rigorous research and hardwork, we have deduced that the pandemic is nowhere near its termination.

Hence keeping the current scenario in our mind people after all have to get to the streets for fullfilling their basic needs such as buying grocery to a necessary task such as going to your firm. This also includes any kind of delivery personnel which are quite popular these days, so keeping in mind all these situations in our mind people would love to get their jobs done in the safest ways possible when they hit the road and it would be more convenient if a device that is always in their hands could be used for exactly this.

1.6 Problem Context

In the current pandemic situation, even after rigorous research and hardwork we have not found a permanent end to this Covid19 Virus and safety of every individual is stake. Also we can not let everyone sit at home, though most of the task have shifted work from home the economy as shown us what bad impact, the current situation has laid upon us.

People after all have to get to the streets for fullfilling their basic needs such as buying grocery to a necessary task such as going to your firm. This also includes any kind of delivery personnel which are quite popular these days, so keeping in mind all these situations in our mind people would love to get their jobs done in the safest ways possible when they hit the road and it would be more convenient if a device that is always in their hands could be used for exactly this.

The “Post Pandemic Crowd Safety” app enables its users to know about the current Covid situation ie how many hotspots are there in the surrounding region, is a place safe

enough to visit or not, what could be the safest path to go from a location “A” to location “B”, where could be a potential hotspot due to overcrowding of a zone. And all of this just at the ease of access of your finger tips. But the system is not only limited to that.

With this system being developed a user can access all the above mentioned information from multiple platforms such as Android, IOS, Web by the means of simple application or simple Webapp.

The application has a minimal interface so that anyone with basic mobile operating knowledge can utilize the system to gain knowledge about the current pandemic situation.

1.7 Rationale behind the System: Why do we need this System?

As we can't disrupt the economy further, everything has to get back to the track and the virus is nowhere termination hence we need to find a suffice method that can ensure the safety of general public.

1.7.1 Benefits of the System: What are the potential benefits derived?

Upon implemented, the system could bring about significant tangible and intangible benefits:

↳ Tangible Benefits

S. No.	Tangible Benefits	Description
1.	No investment	Users doesn't have to pay anything for using this system on android mobile Everything in this system is free of cost. Even user have flexibility to use any interface there is no restriction in the system.
2.	Ease of Access	This system can be accessible anywhere whether it is Android, IOS and also through Webapp.
3.	Easy to use Interface	The application will be minimalistic yet intuitive to use. So that even a layman can use it with ease and in efficient way.

4.	Accurate Geo-Location	Upon permission agreement user is feeded with accurate Geo location of himself as well as surrounding hotspots.
----	------------------------------	---

Table 1.1: Tangible Benefits

↳ Intangible Benefits

S. No.	Intangible Benefits	Description
1.	Reduces effort	It reduces our effort to great extent. E.g. Get all the surrounding hotspots in real time.
2.	Time Saving	The user is feeded with appropriate information as soon as the app launches, he/she does not have to fiddle around with the options to get information.
3.	Friendly User Interface	The application that is developed would facilitate its end users with interactive graphical interfaces that would be easily adaptable.
4.	Customer Satisfaction	The user interface will be user-friendly and easy to use which will result in improved satisfaction of the customers.
5.	Transparency	This system will provide transparency in the hotspot determination. Every hotspot or crowded zone determined is shown irrespective of the place.
6.	Maintainability	Maintenance of the application is painless for its user. Any version updates will not bring out any haphazard changes that could disrupt the workflow of a potential user.
7.	Ease of learning and Usability.	Keeping in mind the target audience, the app will be designed for the generalized audience and hence will be

		easy to use. The application will be developed keep in mind the YAGNI principles.
--	--	---

Table 1.2: Intangible Benefits

1.8 Objectives of the System

1.8.1 Project Objectives

The aim of current system is to provide safety to general public by the means of digitization and ease of access so that an end-user can easily determine any crowded zones or corona hotspots from their device itself. In order to achieve this our team has devised the following objectives:

- ✍ Since this application is crossplatform and keeping in mind most of the users, we have prioritize the development of android application, IOS application and then to a progressive webapp.
- ✍ Since the user needs to determine nearby hotspot we have to implement Geo-locations services.
- ✍ The system also needs to notify users regarding any crowded zones.
- ✍ Their is also an utter need of providing a safest path between a source and a destination.
- ✍ Assume that every 8 people out of 10 have a smart device on which our application program will execute.

1.8.2 Objectives

The proposed system will allow the developer to attain relevant knowledge and concepts in specializing Cross-Platform software development and Automation. The objectives are disreputably emphasized on its concepts and ideas rather than trained expertise. The developer would have to undergo various sections of software development, project management, human-computer interaction, usability factors along with the mechanisms of development principles to produce highly reusable quality software on time. But the prime aim to undergo mobile computing concepts is to increase preferred output with user satisfaction within less effort and time to encounter an easy solution for checking corona

hotspots nearby him/her and reduce all the efforts that are required for knowing the current details of a hotspot zone.

The following objectives listed below are of prime importance to the developer: -

- ✚ Gaining knowledge of web-based development, web technology and JSON.
- ✚ Learning the concepts of Services such as Web Services, Location-Based Services.
- ✚ Learning how to deal with web-based software and development along with its database management.
- ✚ Able to perceive real-time application with their integration on computer/laptop.
- ✚ Implementation and integration of database services with web applications.
- ✚ Understanding BLOC- For a Robust architecture
- ✚ Understanding of data gathering methods and indexing it into the database of relevant style.
- ✚ A better understanding of project management concepts i.e. scheduling, budgeting, cost estimation, work break down structure, Gantt chart, Pert chart etc.
- ✚ Learning and Practices of Docker, Kafka and PostgreSQL.

1.9 Scope of the System

The proposed system will be beneficial in several ways. The “ **Post Pandemic Crowd Safety**” is actually about safely letting a user know about its current surroundings by providing detailed information about the current pandemic situation around him/her. So, ultimately the user will be able to know how many corona hotspots and particularly in which region exists and safely avoid them. This Application will save a lot of lives and further help in decreasing the rate of spread of coronavirus

1.10 Limitation of the System

Limitations are always a part of every project. The project scope is limited to a confined boundary as listed below:

- ✚ The user needs an internet connection.
- ✚ The user also needs to allow global positioning system services on his/her smart device.
- ✚ The user needs to be validated using email address and password.

- ⇒ Minimum Android version is lollipop (SDK 21)

1.11 Assumptions Made

This system is although easy to use in terms of its functionalities but even then, some assumptions need to be made before the development of the system:

- ⇒ The user should have internet connection.
- ⇒ The user also needs to allow global positioning system services on his/her smart device.
- ⇒ Minimum Android version is lollipop (SDK 21)

1.12 Success Criteria

Elements of Success depends on the depth of understanding and experience gained from the apps under review and how the developer effectively cracked the application-based problems that led to the creation of that app.

- ⇒ The app works well and is compatible with the specified browser. The app will be useless if it cannot work properly on a mobile device. There should be no undesirable or false performance while operating on the material.
- ⇒ The plan meets all the requirements of the proposed solution. If the app provides the desired functions and produces the expected result i.e. if you meet the user's requirements, pass the test cases above. The application must meet the system requirements and enhancements mentioned above
- ⇒ The project will be considered constructive if it is a proper performance analysis. The developer will be able to provide a successful implementation of the web application in terms of interoperability, efficiency and effectiveness.

1.13 Functionalities of the System

The proposed system having some core enhanced and some special functionality. So, let's know them one by one and use of this functionality in this system.

1.13.1 Core Features

<u>Funtionality</u>	<u>Description</u>	<u>Done By</u>
Data Collection	Every user that installs the app will provide the geo-location data which will be send to the server to be analyzed.	Pranav
Organising Data for efficient processing	Entire location data will be stored in the “DATA WAREHOUSE” to make it easier for the efficient retrieval of the data.	Manan & Abhay
Detecting Crowded Zones	The organised data is further retrieved and algorithmically analyzed for detecting the hazardous zones.	Manan

Table 1.3: Core Features

1.13.2 Enhanced Features

<u>Funtionality</u>	<u>Description</u>	<u>Done By</u>
Minimal User Interface	The user should not be bogged with plethora of options but what he wnats to see.	Pranav
Scalable	The back-end server will be extremely scalable to handle humongous amount of users.	Manan & Abhay
Cross-Platform	By using Flutter Framework we can manage to develop our product on multiple platforms with single code-based.	Pranav

Table 1.4:Enhanced Features**1.14 Conclusion**

Chapter 1 includes a background study of the program. Learning the program has improved the purpose of the program and the objectives / objectives of the program. After the purpose of the program is identified, the developer then brings the benefits of the program to its intended user and how the program can benefit its users. The developer then developed the desirable features that will be used in the proposed system to solve the problems identified in the problem category. The chapter also includes submissions for the program once completed. Program limitations have also been identified which limit the scope of the proposed project.

CHAPTER 2: PROBLEM DESCRIPTION

2.1 Current Problem Description

After the analysis of the initial investigation, the researcher identified and identified some of the key application issues described briefly below:

2.1.1 Description of Problem Area

- ✚ **No Information Provided:** Currently there is no system which is deployed which can easily tell about the nearby corona hotspot zones to a user, what is the range of spread of virus etc.
- ✚ **No provision for safe travel methods:** Currently any user cannot get necessary information regarding a safe route to travel between a given source and destination.
- ✚ **No real time crowded zone detections:** As of now current systems do not have any kind of live notifying or detecting any crowded zones which can be potentially termed as unsafe zones

2.2 Problem Importance and Justification

The solution for the current system is to develop a system which will sort out all the above problems. The proposed system will work upon the current issues and will overcome the problem that it has right now:

- ✚ **Accurate Information Provisioning:** Since the current systems can only tell a user about the state level hotspot zones the considered application will be able to provide all the relevant “nearby” hotspots to the user for providing information to him/her.
- ✚ **Safest Route Detection:** User will be able to get the safest path to travel via when he/she will provide the source and destination. The path determine will try to provide the minimal/ contact free path to any hotspot zones.
- ✚ **Real Time crowded zone detection:** As the system promises, the application will provide all real time updates regarding any crowded zone which might be a risky hotspot and contaminated region based on crowd sourced data.

2.3 Nature of Challenge

2.3.1 Domain Challenge

To build this app, the researcher would have to undergo some major challenges during the development like:

- ✍ **Mobile Commerce:** Since the major deployment of the system would be in terms of application be it as a progressive web app or an android/iOS application so there is a challenge to learn about how application industry is driven
- ✍ **Synchronization:** One of the most influential features of the app would be synchronized data as per user. So, managing all the relevant data of thousands of users storing them in database and efficient fetching of it would be a task.
- ✍ **Interaction Principles:** To understand the interaction principles between the human and the mobile devices and to develop the application keeping in mind the implementation of all those principles which will justify the successful implementation of the application.
- ✍ **Location Based Services:** To understand the location-based services which contains GPS module to get the current location.
- ✍ **Easy Interface:** An application in current times becomes popular which is a necessity for crowdsourcing data only when it has easy to use interface so studying about minimalism and what are the likings of an end user is also a challenge.

2.3.2 Technical Challenge

The challenges that would come on the way of the researcher while working on the technology area are as follows:

- ✍ **Cross Platform Development:** The application being developed is based on Google's new framework Flutter. Since this is a new platform with a different language to use which is Dart. It brings its own learning curve.
- ✍ **Intermittent data provisioning to the server:** The application needs to crowdsource data for its success so a supply of stream of location data based on changes in a user's location is to be served to the server as well, which we have to figure out how to do efficiently otherwise it would be considered failure for an application which is jittered or doesn't have a good experience all in all.

- ⇒ **Determining Crowded Zones with Accuracy:** It is quite a challenge to judge how would we determine a crowded zone after a potential n number of users have supplied their location data as an algorithm needs to be applied which will judge how near and how many people are there in a zone so as to declare it a crowded zone.
- ⇒ **Integrating Various Modules in one Application:** It will be quite challenging to integrate various modules such as GPS location services, Web Services, Database Integration and many more at a single place. The developer will have to learn numerous API's to implement the system.
- ⇒ **Hardware Fragmentation:** There are thousands of different devices running on the Android platform and those various devices exist with different properties such as memory, CPU speed, and click resolutions. There are also possible variations at the operating system level. A popular example is the division of Android devices with different screen sizes and resolutions. Therefore, the developer should conduct market research to determine which mobile phones are most popular with the target audience and develop the first ones.
- ⇒ **Software Fragmentation:** As such, there are too many Android versions available that work on devices. Therefore, I as an engineer cannot focus on one or more recent versions because it is not easy for users to upgrade their app so most users should not upgrade. Therefore, an engineer will have to learn which programs are most popular and develop them with the latest widely accepted idea in mind.
- ⇒ **Web Services:** As such, the app needs to connect to a remote database to sync with the app to download relevant data, so the developer should need to have in-depth knowledge of the web service as it will often be used to download data or match data to a remote site.

2.4 Feasibility Study

A possible study of the proposed project to determine its feasibility technically, economically, timely and efficiently. A feasibility study for a proposed project to determine whether the proposed project was developed / upgraded on time, resources available, within the estimated budget. After the research is done, it is considered whether the work is continuing or not. The feasibility study can be divided into the following four categories

- ⇒ Economic feasibility
- ⇒ Schedule feasibility

↪ Operational feasibility

2.4.1 Technical Feasibility

The implementation of the technology is designed to determine whether the project is likely to be developed within the available resources or not. Possible technical implementation considers the technical requirements of the project, the technical resources required for the project to complete it successfully and efficiently. Specific technical resources that will be available to complete the project successfully are provided below:

Hardware Resources

S. No.	Hardware Resources
1.	Processor - 3Ghz Dual Core Processor
2.	RAM (Memory) - 4 GB and more
3.	Monitor – Any standard monitor
4.	Pen Drive, Mouse, Printer, Keyboard
5.	System Type- 64-bit OS
6.	Hard Drive– More than 80 GB
7.	Accessories – Internet connection.
8.	Windows USB driver for Windows: To connect an Android-powered device with the development environment over USB for USB Debugging.
9.	Android Device for testing real device integration

Table 2.1: Hardware Resources

Software Resources

S. No.	Software Resources
1.	Android Studio, Flutter Framework, Visual Studio Code
2.	Languages – Dart, Python

3.	Plug-In - JDK 6.0 or later (JRE alone is not sufficient),
4.	SDK – Android SDK with API level More than 8 and later
5.	Web Service – AWS
6.	Database Technology –PostgreSQL
7.	Operating System (PC) - Windows 10
8.	Operating System (Mobile Phone) – Android OS/iOS
9.	Case Tools - Microsoft Project, Microsoft Visio and Star UML
10.	Documentation and Presentation tool - Microsoft Office 2010
11.	Environment Sync Management - Docker

Table 2.2: Software Resources**Resources of Execution**

S. No.	Resources for Execution
1.	Operating system - Android 7.0 and later
2.	Android Smartphone - GPS enabled mobile phone, Internet services like Mobile Data.
3.	CPU: 600 MHZ
4.	RAM: 50 MB (estimated)
5.	Disk space: 100 MB (estimated)
6.	A good data/internet connection for customers

Table 2.3: Resources for Execution

2.4.2 Economic Feasibility

The main purpose of economic viability is to identify the financial benefits and costs associated with project construction. In the case of any plan if the estimated benefits are equal to or exceed the expected cost calculated by the plan then the project will be economically viable. As the proposed location of the program is based on android which is an open source platform, so there is little investment in the development component. And any software or software program needs to be screened for potential applications. Only the required investment can be a tangible tool to evaluate the system.

2.4.3 Schedule Feasibility

Project planning is done to ensure that the project can be completed in a timely manner or not as determined by the Gantt chart. Dates are set for each stage throughout the project and it must be ensured that the project is able to complete within the set dates and plan. If a project can be completed within a schedule, it is likely that planning is possible. The Gantt Chart is a project management tool that can be used to measure project performance. The proposed timeline for the application will take approximately 25 weeks and the time management of each project is best done with a Gantt chart.

2.4.4 Operational Feasibility

Acquisition measures in which the proposed plan resolves issues identified in the first phase of the project; how the project aims to meet the identified needs during the definition of scope and how the system utilizes the opportunities identified while defining the scope. The project will work inevitably as the main goal is to solve the problems of those who are facing problems in the current situation facing many people.

2.4.5 Conclusion of Feasibility Study

This project is technically feasible because the hardware, software and equipment needed to create the system are ready. After considering the costs and benefits incurred in this program, it is concluded that the project is economically viable. The project will be completed on time and will follow the deadlines as set out in the Gantt chart. The program will be publicly accepted and will meet the needs of the users.

2.5 Conclusion

Chapter 2 deals with the problems identified in the current study area i.e. problems in the application area. The developer has identified all the problems typically faced by the average user. Problems are listed along with reasons for each problem. After all the problems were identified, the next job of the engineer was to provide appropriate solutions to the problems. For the same purpose, a solution corresponding to each problem is written along with the reason. A feasibility study was conducted on the feasibility of ensuring that the proposed system is technically, operational, and economical and can be completed within a specified time. Research into the potential for success, however, paved the way for further engineering.

CHAPTER 3: LITERATURE REVIEW

3.1 Introduction to Literature Review

A literature review is a description of what work other researchers have done related to the current topic. It is done using published books, journals and other research papers and is the basis of current research. The following is the purpose of the book review:

- Perform Carrying out critical research of the research you have done to find out their ideas, strengths, weaknesses and methods.
- To revise existing suggestions related to the topic, review the findings and opinions of others.
- T Answer Answering certain questions and removing confusion related to the current topic.

3.2 Advanced Preliminary Research

The developer had a mindset in order to make the lives of the people easier and safe in a way that they can go to the desired place without any COVID-19 fear. This idea is actually really important implement in order to increase the number of Employee going to the workplace by making sure of their safety from COVID-19. Similar kind of idea is being implemented by the professors of MIT (Massachusetts Institute of Technology is a private research university in Cambridge, Massachusetts). The level/scale of the system is much higher than it appears in the first look. We are facing COVID-19 (Coronavirus Disease 2019) together but handling it without a nominal vaccine seems next to impossible. This situation is leading to downfall in economic growth and resources which are directly or indirectly affecting our lives. One of the major reasons for the same is various sectors (like corporate, education, government) not being at its pace. We can't really wait for the vaccine to be up for us in such short span. Thus, one conceivable way of making these sectors work again is to make the necessary physical appearances possible in the offices taking proportional securities. This paper validates a system that is capable of controlling the spread of Corona Virus using Safest Path Detecting Algorithm. This system receives Geo locations from end users and provided the Safest Route.

3.3 Academic Research

Prior to developing the proposed program, the engineer undertook course studies by reading research papers, journals etc. to understand the real need for **Post Pandemic Crowd Safety**. How useful it will be for users. It will also help the engineer to refine the functionality that should be included in the system.

3.3.1 Crowd Sourcing Technology

Since the system heavily depends upon crowd-source the data where it is a real challenge to synchronously process the data received .

3.3.2 Detecting Hotspot

So after receiving the Geo Locations of the users, the system detect the crowd hotspot by using K-Nearest Neighbours Machine Learning algorithm.

3.3.3 Safest Route Detection

The system also provision the safest route detection in order to successfully implement this we firstly deduced all possible routes and filter the safest route based on the minimum hotspot coverage.

3.3.4 Human Computer Interaction Principles

It requires good design, testing, usable interface etc. easy to use visually, payment power etc. The developer must be clear about the terms of communication between the person and the mobile devices so that the application can be developed keeping in mind the implementation of all those terms that will ensure the successful installation of the application.

3.4 Domain Research

Domain research is required as the project environment varies in different contexts and once this is completed a specific module, then we can work on the analysis and design of that particular module provides easy use. During the research stages, the researcher acknowledged that there are enough problems that a user experiences in their daily lives.

The domain domain application focuses on the use of the cross platform and portable computer. In order to have an in-depth knowledge of the user's needs and key features of the system, the researcher must use technology to reduce the risks faced by the user as a result of this epidemic.

3.4.1 Android

Android is a Linux based operating system and is more of a software stack for mobile phones which chiefly comprises of an operating system, middleware and key applications. Android OS is an open source operating system for mobile devices having a kernel based on Linux kernel. Its middleware, libraries and APIs are written in core C while the applications are developed for android using java with android class extension.

Here in figure different android OS versions is shown:



Fig 3.1: Android Version

3.4.2 Flutter Architecture

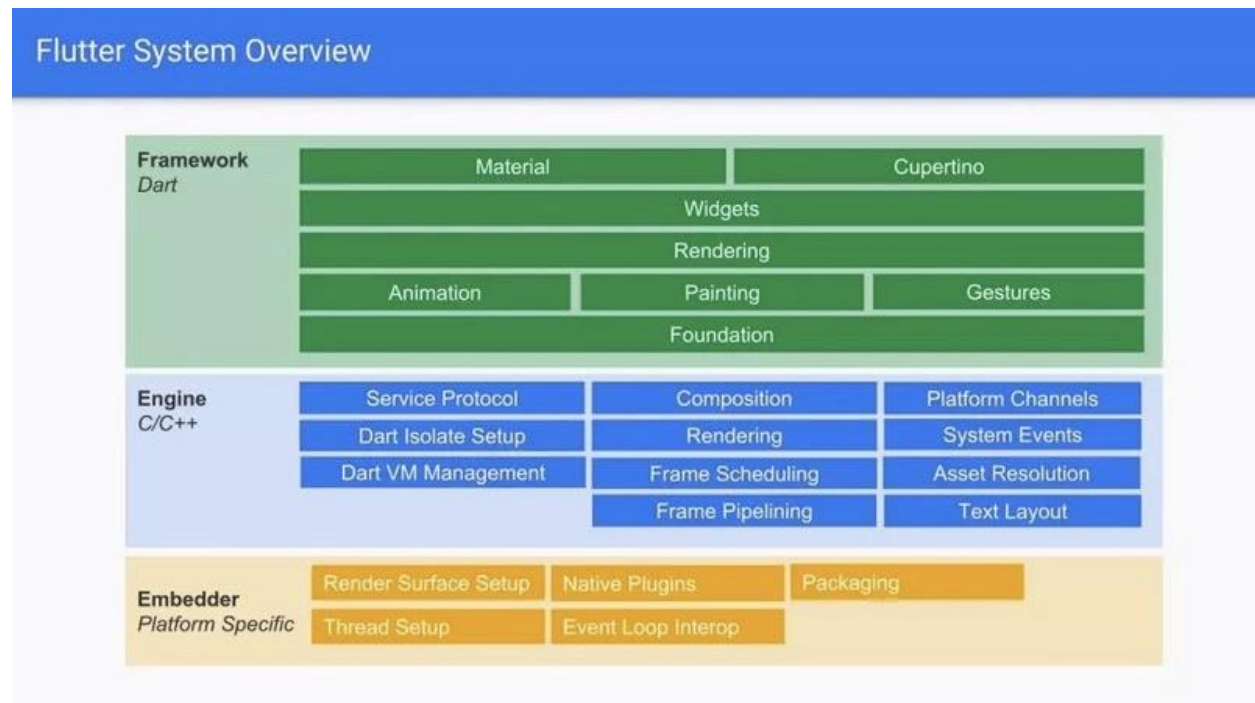


Fig 3.2: Flutter Architecture

In addition to the key concepts we discussed earlier, Flutter brings basic layouts that you can use in your app and manage its status easily. The formulation used in Flutter is called the Business Logic Component (BLOC). Basically, it is an event-based approach that allows you to launch events and manage state changes based on them. BLOC is a great way to separate your business logic from the user interface and manage key logic points by testing. The key ideas used in the construction of the BLOC are simple, fast, and experimental, and all these objectives are definitely achieved within the BLOC structures.

But this is a different topic that we can look at later.

Application Framework: Displays a variety of Android OS capabilities to app developers so that they can use them in their applications by providing an open development platform, Android gives developers the ability to build more rich and new apps. Developers are free to take advantage of the hardware of the device, access location information, use background services, set alarms, add

notifications to the status bar, and much more. This will help the developer to improve the application.

3.4.3 Security and Permissions

All Android apps use their own process and security where these apps are used with the help of Linux installation. Android has a security mechanism that prevents processes from performing various tasks. Depending on the permissions the process can access data from other applications or other applications. This concept will help the developer maintain the security of the application and gain access to all application permissions. For example, an engineer might specify access to a device's Bluetooth function through a program

3.5 Critical Evaluation of Literature Review

The development of the proposed user application is researched to come to the conclusion that all the features required by the end user to maximize office space. The literature review forms the basis for further research that provides insight into the proposed system. Systems such as the proposed plan have been researched to form the basis of the plan; what problems a builder may face, what features cannot be used at all.

During the review of the documentation, the background, technical and academic research was conducted to obtain in-depth information on the features to be used in the proposed system and to evaluate the technologies to be used to design the system. Therefore, a second study was conducted under review of the literature by studying the work of another lead engineer in developing the proposed system.

3.6 Conclusion

Chapter 3 includes reviewing past documents to find the basis for a proposed project. Under the literature review, existing programs have been studied to gather information in the research area; what new features in the market and what features a developer can incorporate into his new system. The chapter includes details and resources that form part of the literature review and which are lessons or future lessons to gather information and information about the study environment.

CHAPTER 4: RESEARCH METHODS

4.1.1 Primary Research

Primary research is defined as collecting data or information about the research domain from the real world, for the first time in a given case from the real world. It is designed to remove the confusion and ambiguity that has come to the minds of engineers after conducting a second survey by asking those questions directly to forum users and looking at their suggestions. The developer can use this research to move forward in project development. The main research methods used by the engineer contain the questions and tests described below:

4.1.2 Questionnaire

The questionnaire consists of several questions printed on a piece of paper in a clear sequence to obtain the necessary information for the targeted users in a short period of time. A questionnaire is provided for respondents who are expected to read and understand the questions and write down the answer in the space provided. As a user the types of users of the proposed system; the engineer will distribute a list of questions between them.

Justification for Conducting Questionnaire

The reason for choosing a questionnaire as a primary data gathering method is described below:

- ✍ Since the intended users of the program are quite numerous, it is impossible to reach them individually to collect their needs and suggestions.
- ✍ In the case of questionnaires, it is possible to reach as many people as possible at one time, regardless of their geographical location.
- ✍ Since mathematical methods can be used in the analysis of the questionnaire it is effective in terms of graphs and charts more accurate.
- ✍ Users are free to answer questions anonymously, so there are plenty of opportunities to find specific needs.
- ✍ It is an inexpensive method, as it can be emailed to participants or distributed as a paper-based questionnaire.

Questionnaire for Customer

<p style="text-align: center;">QUESTIONNAIRE</p> <p style="text-align: center;">A SURVEY TO FIND THE RESPONSE OF PEOPLE FOR A CROWD SAFETY SYSTEM</p>

Respected Sir/Mam,

The researcher is a finalist at the Panipat Institute of Engineering and Technology, Panipat pursues a B Tech in Computer Science broadcasting. For basic research and data collection work is done research for the final year project. The researcher kindly requests that you take a few minutes to answer the questions and help us complete this survey by completing the questions below. The questions were carefully designed to follow all the ethical guidelines of Kurukshetra University's research work. The title of the project is “**POST PANDEMIC CROWD SAFETY SYSTEM**”. The purpose of this study is to find out how people know about their surroundings and the need to travel safely to an epidemic.

This survey will help the developer to identify the various preferences of the user and modify the current system. The results of this test will be very helpful in obtaining the information needed to build the system.

Instruction

This questionnaire sections will help the researcher to estimate the project research deliverables. You must answer all applicable questions thoroughly.

Section A: - This section is optional and comprises of your personal and professional details.

Section B: - Check the most suitable options from the questions below and answer all the short and long answer questions according to your point of view.

Disclaimer

Your response to this survey or any particular question is completely confidential. Your identity will not be revealed and the data here obtained will only be used for statistical purposes.

SECTION A

Name	_____
Email	_____
Profession	_____

SECTION B

Question 1	Are you aware of COVID-19 (Coronavirus Disease 2019)? <ul style="list-style-type: none"> • Yes • No
Question 2	Are/Were you affected by COVID-19 in any way? <ul style="list-style-type: none"> • Yes • No
Question 3	Do you take proper precautions/safety to be COVID-19 free? <ul style="list-style-type: none"> • Yes • No
Question 4	Do you encourage others to do the same? <ul style="list-style-type: none"> • Yes • No
Question 5	Do you believe in the fact that COVID-19 is not going away anyway and it will stay with us for a large period? <ul style="list-style-type: none"> • Yes • No
Question 6	Do you travel for completing any type of task during pandemic (Task could be as simple as going to a general store or travelling to your firm)? <ul style="list-style-type: none"> • Yes • No

Question 7	How probable do you find yourself to be prone to a potential COVID-19 Zone while travelling? <ul style="list-style-type: none"> • Yes • No
Question 8	Would you like a system that could easily notify you about current and potential COVID-19 Hotspot Zones? <ul style="list-style-type: none"> • Yes • No
Question 9	Do you feel an urgency for a system to be developed that can help you to travel from one location to other by avoiding or minimizing any marked or potential COVID-19 Hotspot Zones, hence minimizing the risk of catching Corona Virus? <ul style="list-style-type: none"> • Yes • No

Thank you for your cooperation and for providing valuable information. Your concern and your help are highly appreciated. Please ensure once, whether you answered every question in the questionnaire or not. After fully completion of the questionnaire kindly deposit or submit this questionnaire to the researcher.

4.1.3 Development Methodology

Development Methodology is a way of showing software in different stages and how the system and software will work well all this is down to the development process.

The V-Model contains various sections that work to show how a program or software works.

V-model Methodology

After reviewing all the software options, the developer has selected "V-MODEL" as it is compatible with project issues. Any change in the system can be made at a lower cost due to the frequency of new increases being produced. Modifications can be discussed and features can be improved or removed depending on feedback. This will effectively provide

its customer with the complete program they want or need. This approach divides its functions into smaller frames in order to achieve goals.

Phases of V-Model Methodology

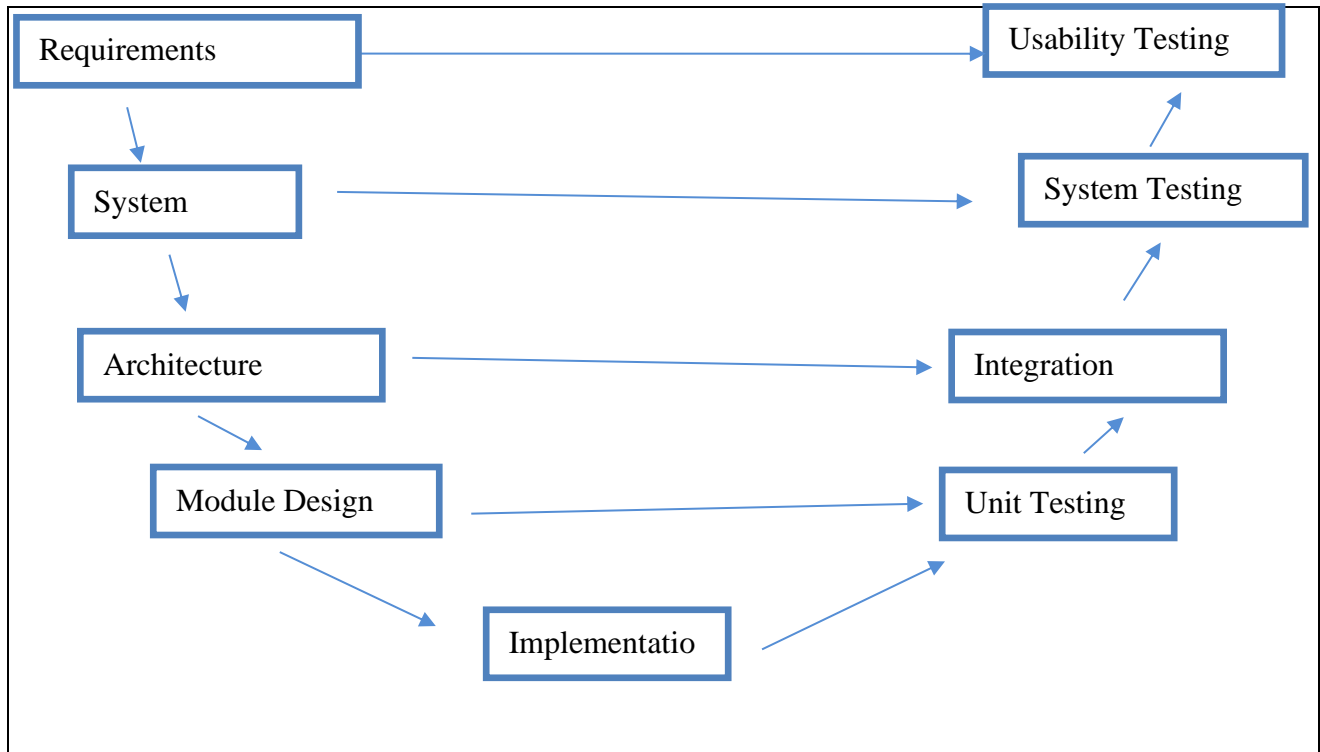


Fig4.1:Phases of V-Model Methodology

This methodology has the following phase that developer will pursue while developing the system.

- ✎ **Requirement Analysis-** It is the first step in the verification process. At this stage, the developer will not be consulting how the system will be constructed; it will be a regular dialogue and a user needs document is put forward. This document will convey information related to system performance, performance, security, data, interface etc. This document is required by business analysts to transfer system functionality to users. For now, it will be just a recommendation.
- ✎ **System Design-** At this stage the product structure is exposed. Built after keeping in mind the required summaries. In addition, while pursuing documents, if something is

inappropriate in the formulation, the developer is held accountable and changes are made as a result.

- ✚ **Architecture Design-** It is also known that computer architecture or software design should understand the modules and functionality of the modules to be integrated.
- ✚ **Module Design-** In this section, building structures are further subdivided into smaller sections that can be sorted and defined separately. Units are known as modules. It can be removed separately by the developer of the program.
- ✚ **The Validation Phases of the V model**
- ✚ **Unit Testing-** Design in the module design phase is coded during this verification phase. Unit testing is code-level testing and helps to reduce bugs early, however, not all errors can be detected by unit testing.
- ✚ **Integration Testing or Interface Testing-** Related to the construction phase of the structure. Integration testing is performed to assess the presence and connectivity of internal modules within a system. In other words, in this section, different businesses will be tested together to find errors in the interface.
- ✚ **System Testing-** Directly related to the design phase of the program. Its evaluation of the effectiveness of the entire system and the statement of the program being developed through the border plans. Also, most software and hardware compatibility issues can be identified during system testing.
- ✚ **Acceptance Testing-** Approval testing, which is related to the business needs analysis phase and includes product testing in the user area. Identifies issues related to other applications accessible to the user environment. Acceptance testing also determines ineffective concerns such as uploads and operating errors in the user's real-time environment.
- ✚ **Release Testing-** This is the time when a decision should be made if the product or software created is suitable for the end user.
- ✚ **Justification for choosing V-Model**

After some discussion, the engineer selects a Waterfall Model to improve the system. The most attractive feature in choosing a V-Model is-

- ⇒ **Stable project requirements:** After each phase, it produces documents and as a framework for our Final Year project we need to submit documents after each phase so that it will be more efficient.
- ⇒ **Progress of system is measurable:** After each phase, it produces the documentation and as the structure of our Final year project we need to submit the documentation after each phase so it will be best suitable.
- ⇒ **Strict sign-off requirements:** As developers aim to satisfy the user and to the satisfaction of the user the developer will be providing the user's preferred functions and relevant features to make the system more efficient.
 - Highlighting needs and formats before writing a single line of code ensures minimal wastage of time and effort and reduces the risk of falling schedule, or expectations of end users not achieving.
 - In the life cycle stages of converted waterfalls are acceptable to accumulate. Due to the scattered phases, most of the size is familiar with the modified waterfall model in software engineering. In the meantime, several tasks can be performed simultaneously, which ensures that the software crashes are removed from the actual development phase as well as the high cost of making changes to the software before saving the launch.

Making changes in the basic structure is possible, as there are certain healthy stages at the same time. In case, there are errors included due to changes made, fixing them is also easy (Testing can be done). This helps reduce any anxiety by mistake.

4.1.4 Development Plan

TOTAL DURATION		START DATE	20 th August 2020	END DATE	25th April 2021
PHASE AND DURATION		TASKS			
PROJECT DEFINITION		<ul style="list-style-type: none"> ✚ Idea Generation ✚ Project Title Selection ✚ Abstract Draft Project Proposal ✚ Project Proposal Form 			
PROJECT PLANNING		<ul style="list-style-type: none"> ✚ Work Breakdown Structure ✚ Schedule and Time Estimation 			
REQUIREMENT ANALYSIS <i>Duration: 7 weeks</i>		<ul style="list-style-type: none"> ✚ Define and Finalize Requirement Specification <ul style="list-style-type: none"> » Project Background » Problem Context » User Requirements » Set Objectives » Identify the Scope of Project » Features and Functionalities ✚ Organizing Project Specification Form ✚ Research and Analysis <ul style="list-style-type: none"> » Research <ul style="list-style-type: none"> ✓ Academic Research ✓ Secondary Research ✓ Human-Computer Interaction » Analysis <ul style="list-style-type: none"> ✓ Domain Analysis ✓ Existing System Analysis ✓ User Requirements ✓ User Profiling and Modelling ✓ Risk Analysis 			

Table 4.2 : Development Plan

4.2 Conclusion

Chapter 4 discusses research methods. The researcher conducted the primary and secondary research in which the technical research was conducted to determine the user needs and the technology and tools to be used. Several research methods have been used by the developer, including Questionnaire and Interviews. Research was needed to avoid wasting time over time in the development phase. Now, the researcher is sure of the features that will be included in the program and how to remove the existing problems. The developer is confident enough to use the proposed app after conducting extensive research related to background, technology, language, tools etc.

CHAPTER 5: ANALYSIS

5.1 Questionnaire

We are developing the "Post Pandemic Crowd Safety System". We compiled a checklist of about 20 questions and sent them to the people we were looking at who could be people working in the organization where they should go to the office and people who might be joining the office soon.

After collecting the answers to the questions from our respondents we can say that this program we are developing will be useful for people who cannot work at home.

5.1.1 Overall Conclusion of Questionnaire

As we asked questions about our project from our target audience the audience was satisfied from our project which included its implementation. The target audience said it was great to have this type of program in their lives able to find the "Most Safe Routes" where the spread of crowd and Corona Hotspot is limited. Also, few people have suggested upgrading a feature in the mobile app that presses notification to employees with tips to keep themselves and others safe from infection.

5.1.2 Overall Conclusion of Interview

As our target audience points out, the "Post Pandemic Crowd Safety System" will be useful in areas where homework does not work. The suggestions we received from those we looked at seemed to be helpful and very helpful in developing a better system, with some of the same changes. So, we built a mobile app and added these features to the application our requested audience.

CHAPTER 6: SYSTEM DESIGN

6.1 Introduction

The construction phase begins when the designer has finished inspecting its investigation phase. The planning phase also includes changing customer information components that configure the product interface to create a product setup category outline. Includes the design of the layout plans you are installing physical representation using design techniques and techniques. In the planning phase of the proposed framework, the building adopted needs as information and will bring the law to be used as a harvest.

6.2 UML Diagram

Use Cases: The use case will be required by the designer for the purpose of proceeding the performance referred to in the proposed framework may be evaluated for approval engineering and environmental testing by many computers successfully tested. In line with these lines, it will be helpful to check the outline by meditation and discernment. (Booch, Rumbaugh and Jacobson, 2008)

Activity Diagrams: Motion sketches will be used by the designer to illustrate continuous control of system modules from action to action. This will help us to indicate simultaneous integration as control elements in the proposed framework. (Booch, Rumbaugh and Jacobson, 2008)

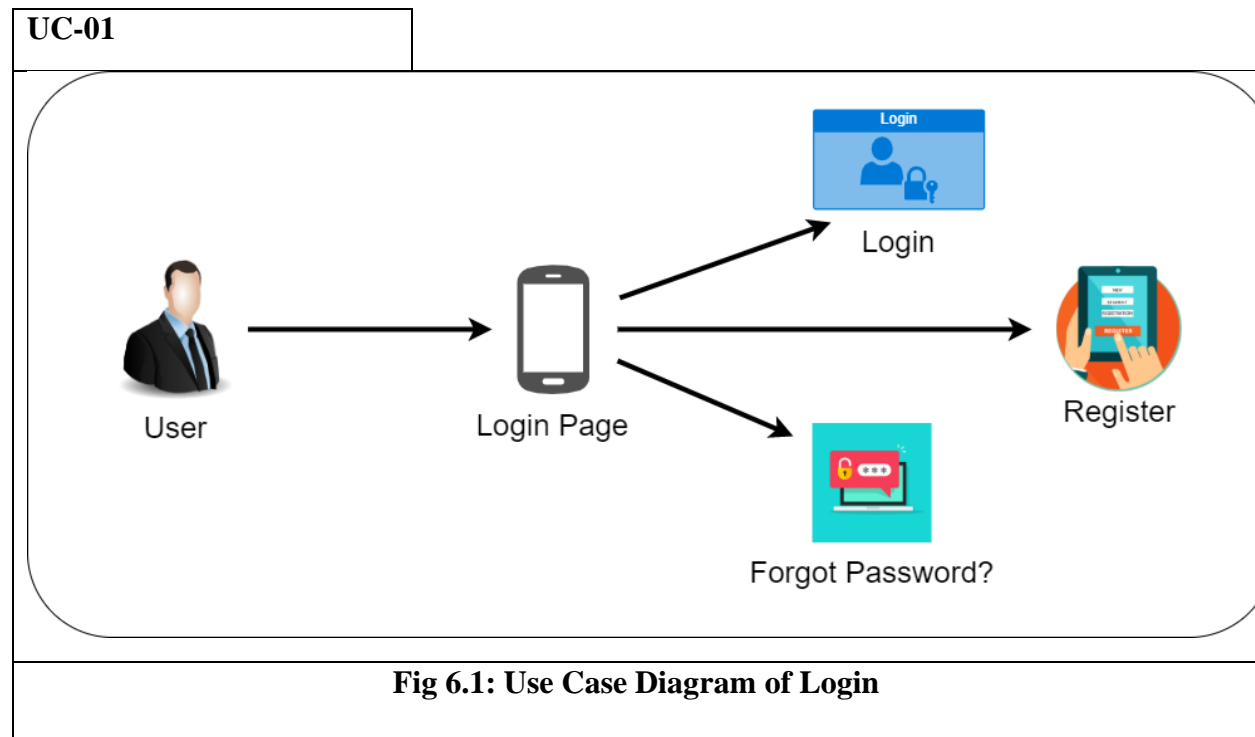
Sequence Diagrams: Structural drawings will assist the designer in displaying winning collaborative planning between clients and framework / modules. (Booch, Rumbaugh and Jacobson, 2008)

Class Diagrams: UML section charts will be required to provide an independent picture of materials to be used in the framework and which can be helpful murder phase. The structure can be used as a source of ideas for use in any the section applies to any state of development. (Booch, Rumbaugh and Jacobson, 2008)

Entity-Relationship Diagram (ERD): ER graphs the designer will need separating objects and interacting helps to place details to be converted into tables in the suspension phase. (Booch, Rumbaugh and Jacobson, 2008)

6.2.1 Use Case

Use Case Diagram of Login

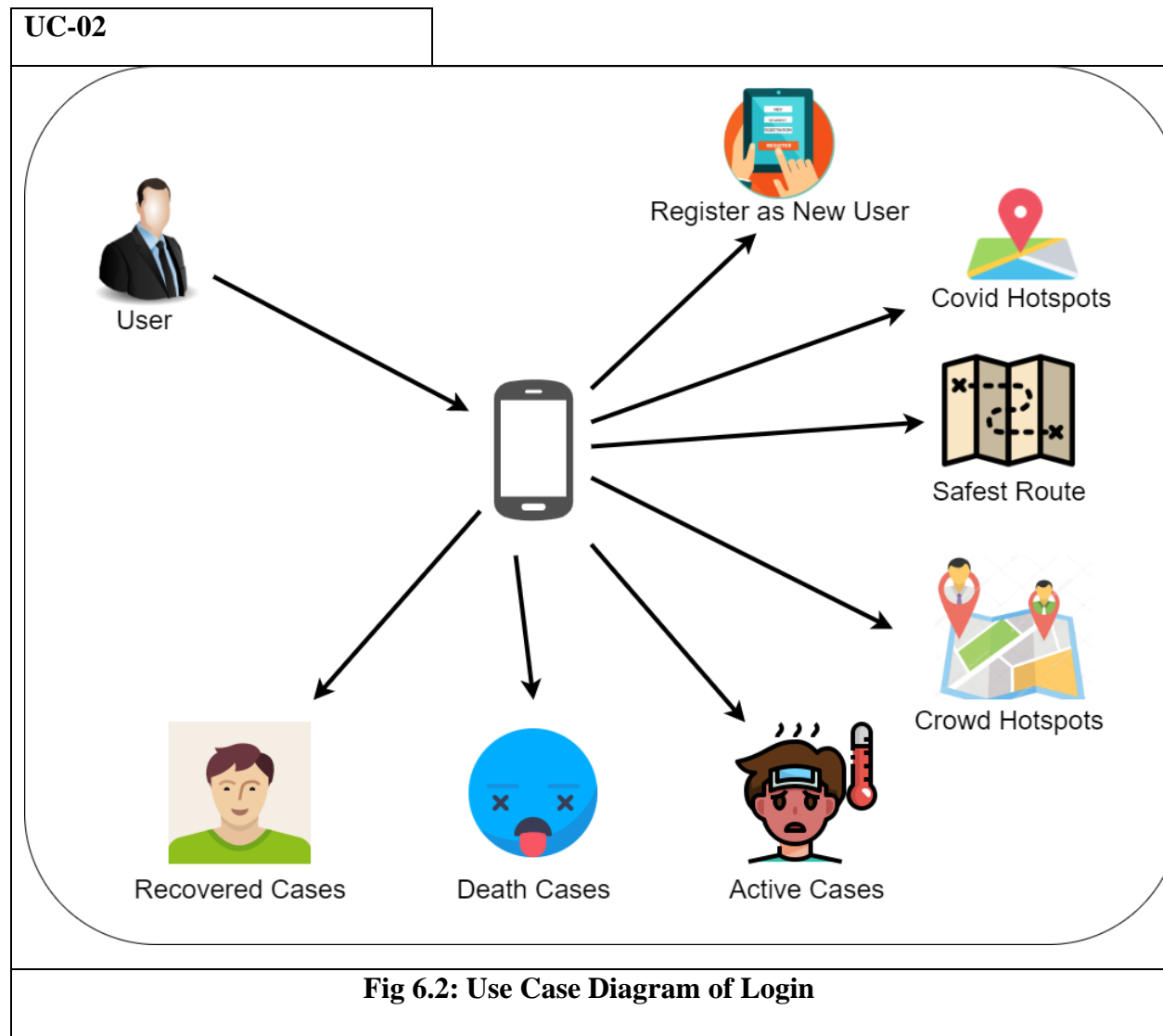


Use Case ID	UC-01
Use Case Name	Login
Description	This allows the user to sign in to his account.
Actor(s)	User
Assumption	↪ User must have internet connectivity for signing in.
Pre-Condition	↪ The actor must be a registered user.
Post-Condition	↪ Signed In successfully.
Primary Pathway	↪ Actor taps on Log In button from the app dashboard. ↪ Actor enters email and password. ↪ Actor clicks on sign in.
Alternate Pathway(s)	↪ Necessary to sign in to synchronize. ↪ Repeat steps 2 and 3 of primary pathway.

Exception Pathway	<ul style="list-style-type: none"> ↪ Actor enters email and password. ↪ The email or password is incorrect/does not exist in database. ↪ System asks the actor to re-enter the username/password. ↪ Go to step 2 of primary pathway.
--------------------------	--

Table 6.1 :Use Case Description of Login

↪ Use Case Diagrams for User Module



Use Case ID	UC-02
Use Case Name	User Module

Description	User can Register, Login, get safest route, get covid Hotspots, get crowd hotspots and also can see recovered, death and active cases in a particular corona hotspot.
Actor(s)	All the User of the system
Assumption	↪ User must know how to operate an Smart device.
Pre-Condition	↪ User must have an android device with gps enabled.
Post-Condition	↪ User get a powerful App which ensures safety in post pandemic.
Primary Pathway	↪ User can see covid hotspots from the app view ↪ User also able to view crowd hotspot (where the crowd is maximum) ↪ User can also see the number of death, active and recovered cases in covid region or hotspot. ↪ User also change the password and login as new user.
Alternate Pathway(s)	None
Exception Pathway	↪ It may happen that at the time of login user enter the incorrect username and password.

Table 6.2:Use Case Description of User Module

CHAPTER 7: IMPLEMENTATION

7.1 Introduction

After the design phase was completed, the engineer analyzed the need for that the requirements cited should meet the proposed system development schema. The implementation phase is one of the most important stages in any project as an engineer will work to make system configuration a real system. Getting Started Phase including the explicit use of tools to convert design modules into a real application. By working on a real system, an engineer requirement can be a program language and editing tool. The developer selects the flutter frame you use dart language and backend were developed in a python running docker jealousy. The developer had to go through a series of bugs that were disruptive time to start working. This chapter goes on to shed some light on the composition initiated and what are the shortcomings during the process.

In addition, a list of problems is discussed and what possible solutions are available were provided.

7.2 Tools used for Implementation

The developer has mentioned all the tools that have been used in the development of the system.

Tools	Purpose
Windows 10	Used as an operating system to run all the below-mentioned tools for the development of the system.
Ubuntu	Used as an OS for back-end development.
Draw.io	Used to draw all the UML diagrams.
Microsoft Word 2020	Used to document the project.
Microsoft PowerPoint 2020	Used to prepare the presentation slide.
Visual Studio Code	It is used for all the coding purposes.
PostgreSQL	Used as database.
Kafka	For Messaging Queue System
Docker	For OS Level Virtualization

Table 7.1: Tools used for Implementation

7.3 Implementation Plan

7.3.1 Implementation Approach

This plan would give the developer an overview of the procedures involved to develop the real system. A strategic plan is required to successfully complete a system. ‘**Agile approach**’ is used to divide the system into separate modules to detect and recover from errors easily. The plan is made to prioritize various modules on the basis of their complexity so that each and every module can be implemented on time and the complex modules should be developed first.

7.3.1 Implementation Approach Steps

1. Since the system had various app for each and every platform so first app had to be developed keeping android as its main focus along side the backend.
2. After that we can move to the progressive Webapp.
3. In the app we have to initially create the login, signup screens along with outh2 bearer API.
4. We had to simultaneously implement the api in the backend docker container which connects to a Database (PostgreSQL) for Login and Register a user.
5. Next we move on to configuring maps Api to delegate it in the app along with that a map viewing screens had to also developed.
6. After this we have design Map interface using flutter google maps Library which will show the map view of the current location when gps is enabled.
7. Now, on this map view we also provide the Covid and Crowd hotspots which is the list of latitudes and longitudes getting by the server through a Rest Api.
8. The latitudes and longitudes of Covid hotspots determined by the database which is the latest database of India so far and will automatically updated when the changes are made.
9. Crowd Hotspots are the regions where the number of users are maximum and this is determined by the K-nearest neighbour algorithm which is a Machine learning algorithm and by doing some modification in it we will get the list of latitudes and longitudes where the crowd is maximum.

7.4 Coding

Folder Structure

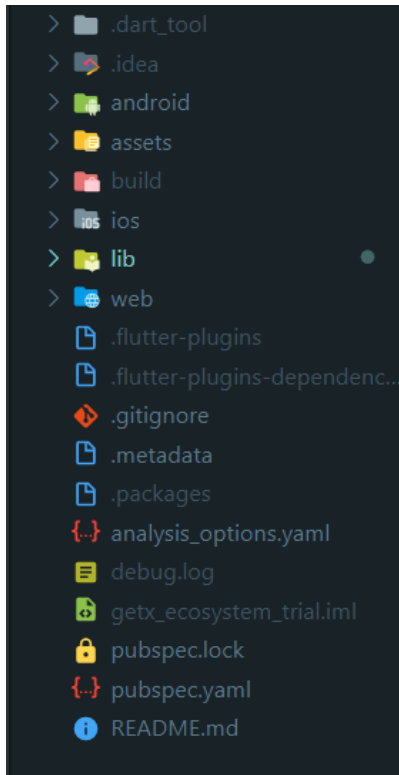


Fig 7.1: Folder Layout Overview

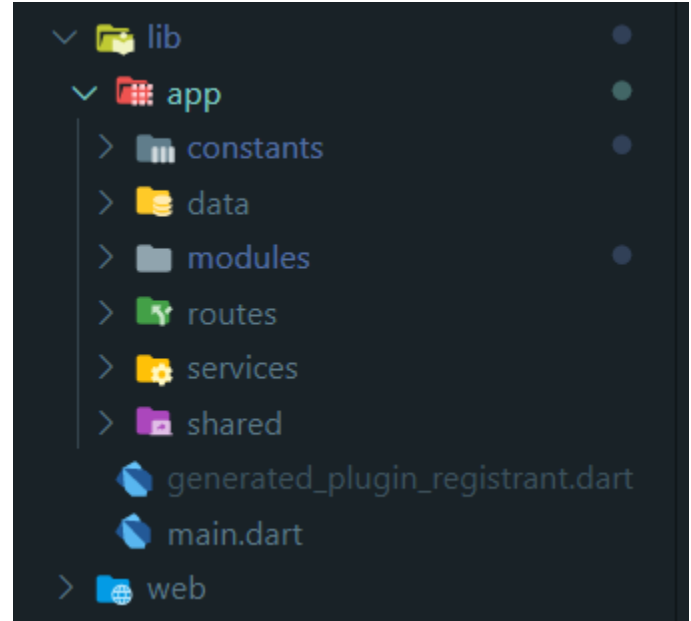


Fig 7.2: App Folder Layout

MAIN.DART

```
import 'package:flutter/material.dart';
```

```
import 'package:get/get.dart';
```

```
import 'app/routes/app_pages.dart';
```

```
import 'app/services/services.dart';
```

```
Future<void> main() async {
```

```
  await initServices();
```

```
  runApp(
```

```
    GetMaterialApp(
```

```

debugShowCheckedModeBanner: false,
title: "Crowd Safety",
initialRoute: AppPages.INITIAL,
getPages: AppPages.routes,
theme: ThemeData.dark(),
),
);
}

```

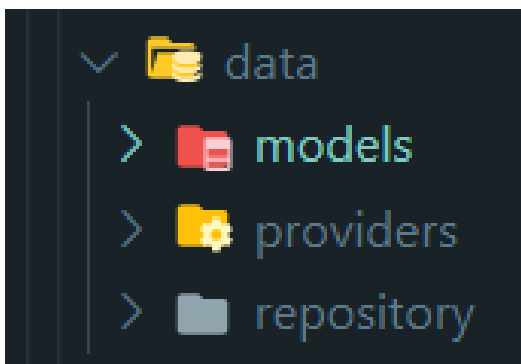


Figure 7.3: Data Folder Layout

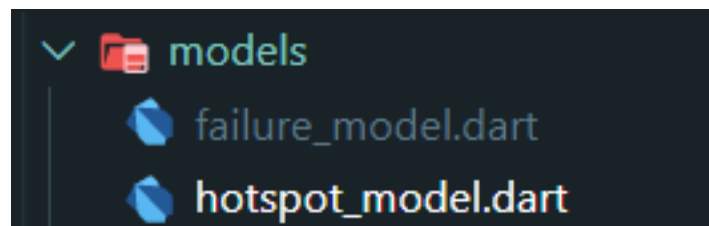


Fig 7.4: Models Folder Layout

HOTSPOT_MODEL.DART

```

class HotSpotModel {
  List<CoronaHotspot> coronaHotspot;
  List<CrowdHotspot> crowdHotspot;

  HotSpotModel({this.coronaHotspot, this.crowdHotspot});

  HotSpotModel.fromJson(Map<String, dynamic> json) {
    if (json['corona_hotspot'] != null) {
      coronaHotspot = <CoronaHotspot>[];

```

```
    json['corona_hotspot'].forEach((v) {  
        coronaHotspot.add(CoronaHotspot.fromJson(v as Map<String, dynamic>));  
    });  
}  
if (json['crowd_hotspot'] != null) {  
    crowdHotspot = <CrowdHotspot>[];  
    json['crowd_hotspot'].forEach((v) {  
        crowdHotspot.add(CrowdHotspot.fromJson(v as Map<String, dynamic>));  
    });  
}  
}
```

```
Map<String, dynamic> toJson() {  
    final Map<String, dynamic> data = <String, dynamic>{};  
    if (coronaHotspot != null) {  
        data['corona_hotspot'] = coronaHotspot.map((v) => v.toJson()).toList();  
    }  
    if (crowdHotspot != null) {  
        data['crowd_hotspot'] = crowdHotspot.map((v) => v.toJson()).toList();  
    }  
    return data;  
}  
}
```

```
class CoronaHotspot {  
    double lat;  
    double long;
```

```
int death;  
int active;  
int recovered;
```

```
CoronaHotspot({ this.lat, this.long, this.death, this.active, this.recovered });
```

```
CoronaHotspot.fromJson(Map<String, dynamic> json) {  
  lat = json['lat'] as double;  
  long = json['long'] as double;  
  death = json['death'] as int;  
  active = json['active'] as int;  
  recovered = json['recovered'] as int;  
}
```

```
Map<String, dynamic> toJson() {  
  final Map<String, dynamic> data = <String, dynamic>{ };  
  data['lat'] = lat;  
  data['long'] = long;  
  data['death'] = death;  
  data['active'] = active;  
  data['recovered'] = recovered;  
  return data;  
}
```

```
class CrowdHotspot {  
  double lat;
```



```
double long;
```

```
CrowdHotspot({ this.lat, this.long });
```

```
CrowdHotspot.fromJson(Map<String, dynamic> json) {  
  lat = json['lat'] as double;  
  long = json['long'] as double;  
}
```

```
Map<String, dynamic> toJson() {  
  final Map<String, dynamic> data = <String, dynamic>{ };  
  data['lat'] = lat;  
  data['long'] = long;  
  return data;  
}  
}
```

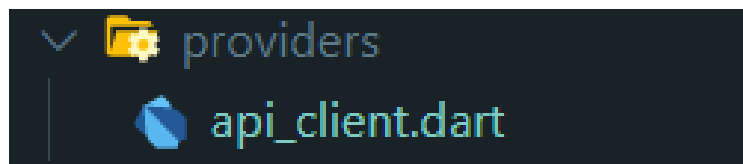


Fig 7.5: Providers Folder Layout

API_CLIENT.DART

```
import 'package:dio/dio.dart';
```

```
import 'package:flutter/foundation.dart';
```

```
import '../services/services.dart';
```

```
import '../models/failure_model.dart';
```

```
class ApiClient {
```

```
  final _api = ApiService().instance;
```

```
  Future login({ @required String email, @required String password }) async {
```

```
    return _postRequestSender(  
      path: '/login',  
      data: {  
        "email": email, //! Encrypt  
        "password": password,  
      },  
    );  
  }
```

```
  Future signUp({
```

```
    @required String username,
```

```
    @required String email,
```

```
    @required String password,
```

```
    @required double latitude,
```

```
    @required double longitude,
```

```
    @required int phonenumber,  
  }) async {
```

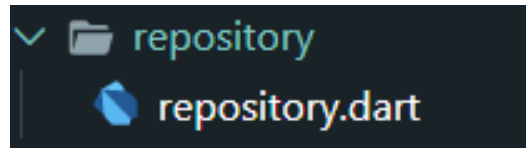
```
    return _postRequestSender(  
      path: '/signup',  
      data: {  
        "username": username,
```

```
"phone_no": phonenum,  
"email": email,  
"password": password,  
"lat": latitude,  
"longi": longitude  
},  
);  
}
```

```
Future getHotSpotZones(  
    @required double latitude,  
    @required double longitude,  
    @required String accessToken,  
    ) async {  
    return _postRequestSender(  
        path: '/covid',  
        data: {  
            "lat": latitude,  
            "longi": longitude,  
            "access_token": accessToken,  
        },  
    );  
}
```

```
Future _postRequestSender(  
    @required String path,  
    @required Map<String, dynamic> data,
```

```
}) async {  
  try {  
    final Response response = await _api.dio.post(  
      path,  
      data: data,  
    );  
  
    return response.data;  
  } on DioError catch (e) {  
    if (e.response != null) {  
      throw Failure(  
        statusCode: e.response.statusCode,  
        message: e.response.statusMessage,  
      );  
    } else {  
      throw Failure(message: e.message);  
    }  
  }  
}
```

**Fig 7.6: Repository Folder Layout****REPOSITORY.DART**

```
import 'package:flutter/foundation.dart';
```

```
import '../providers/api_client.dart';
```

```
class Repository {
```

```
  final ApiClient apiClient;
```

```
  Repository({ @required this.apiClient }) : assert(apiClient != null);
```

```
  Future login({
```

```
    @required String email,
```

```
    @required String password,
```

```
  }) async =>
```

```
    apiClient.login(email: email, password: password);
```

```
  Future signUp({
```

```
    @required String username,
```

```
    @required String email,
```

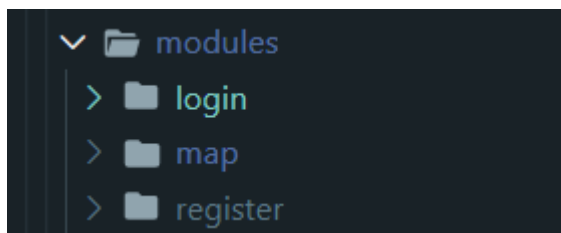
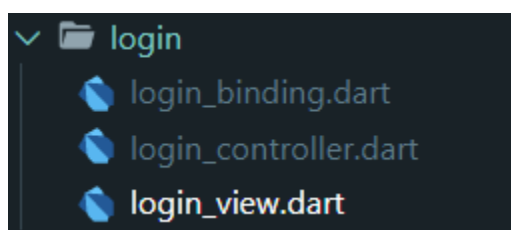
```
    @required String password,
```

```
    @required double latitude,
```

```
    @required double longitude,
```

```
@required int phonenum,  
}) async =>  
  apiClient.signUp(  
    username: username,  
    email: email,  
    password: password,  
    latitude: latitude,  
    longitude: longitude,  
    phonenum: phonenum,  
  );
```

```
Future getHotSpotZones(  
  @required double latitude,  
  @required double longitude,  
  @required String accessToken,  
}) async {  
  return apiClient.getHotSpotZones(  
    latitude: latitude,  
    longitude: longitude,  
    accessToken: accessToken,  
  );  
}  
}
```

**Fig 7.7: Models Folder Layout****Fig 7.8: Login Folder Layout**

LOGIN_VIEW.DART

```
import 'package:flutter/material.dart';
```

```
import 'package:get/get.dart';
```

```
import '../constants/constants.dart';
```

```
import '../constants/style_constants.dart';
```

```
import '../routes/app_pages.dart';
```

```
import '../shared/button.dart';
```

```
import 'login_controller.dart';
```

```
class LoginView extends GetView<LoginController> {
```

```
  final _formKey = GlobalKey<FormState>();
```

Page 64


```
Padding(  
  padding: const EdgeInsets.all(8.0),  
  child: TextFormField(  
    textInputAction: TextInputAction.next,  
    validator: (value) {  
      if (!GetUtils.isEmail(value)) {  
        return 'Please enter a valid email-address';  
      }  
      _email = value;  
      return null;  
    },  
    autovalidateMode: AutovalidateMode.onUserInteraction,  
    keyboardType: TextInputType.emailAddress,  
    textAlign: TextAlign.center,  
    style: const TextStyle(fontSize: 20),  
    decoration: style.kInputDecoration,  
  ),  
,  
Obx(  
  () => Padding(  
    padding: const EdgeInsets.all(8.0),  
    child: TextFormField(  
      textInputAction: TextInputAction.send,  
      autovalidateMode: AutovalidateMode.onUserInteraction,  
      validator: (value) {  
        if (!GetUtils.isLengthBetween(value, 8, 12)) {  
          return 'Please enter a valid password';  
        }  
      }  
    )  
  )  
)
```

Page 66

```
        Duration.zero,
        () {
          Get.offAllNamed(Routes.MAP);
        },
      );
      return const SizedBox.shrink();
    } else {
      return Text(controller.data);
    }
  )),
  Button(
    isTextOnly: false,
    text: 'Log In',
    icon: Icons.lock_open,
    onPressed: () async {
      if (_formKey.currentState.validate()) {
        controller.login(email: _email, password: _password);
      } else {}
    },
  ),
  Button(
    isTextOnly: false,
    text: 'Sign Up',
    icon: Icons.vpn_key,
    onPressed: () => Get.toNamed(Routes.REGISTER),
  ),
],
```

```

    ),
    ),
    ),
);
}
}

```

LOGIN_CONTROLLER.DART

```

import 'package:flutter/foundation.dart';
import 'package:get/get.dart';

```

```

import '../constants/constants.dart';
import '../data/models/failure_model.dart';
import '../data/repository/repository.dart';
import '../services/services.dart';

```

```

class LoginController extends GetxController {
  final Repository repository;
  LoginController({ @required this.repository });

```

```

  final currentState = AppState.initial.obs;

```

```

  String data = 'Press the button ';

```

```

Future<void> login(
  { @required String email, @required String password }) async {
  try {
    currentState.value = AppState.loading;
    final _storage = StorageService().instance;

```

```
final body = await repository.login(  
  email: email,  
  password: password,  
);  
_storage.box.write(storageKey, body["access_token"]);  
currentState.value = AppState.loaded;  
} on Failure catch (f) {  
  data = f.toString();  
  currentState.value = AppState.failure;  
}  
}  
}
```

LOGIN_BINDING.DART

```
import 'package:get/get.dart';  
  
import '../data/providers/api_client.dart';  
import '../data/repository/repository.dart';  
import 'login_controller.dart';  
  
class LoginBinding extends Bindings {  
  @override  
  void dependencies() {  
    Get.lazyPut<LoginController>(  
      () => LoginController(  
        repository: Repository(  

```

```
    apiClient: ApiClient(),  
  ),  
),  
);  
}  
}
```

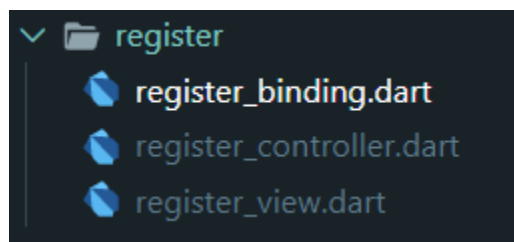


Fig 7.9: Register Folder Layout

REGISTER VIEW.DART

```
import 'package:flutter/material.dart';  
import 'package:get/get.dart';  
import '../shared/location_data_sender.dart';  
import 'package:location/location.dart';  
  
import '../constants/constants.dart';  
import '../constants/style_constants.dart';  
import '../routes/app_pages.dart';  
import '../shared/button.dart';  
import 'register_controller.dart';
```

```
class RegisterView extends GetView<RegisterController> {
```

```
final _formKey = GlobalKey<FormState>();  
  
@override  
Widget build(BuildContext context) {  
  String _email;  
  String _password;  
  String _fullName;  
  String _phoneNum;  
  final RxBool _showPassword = false.obs;  
  return Scaffold(  
    body: Padding(  
      padding: EdgeInsets.symmetric(  
        horizontal: Get.width * .05,  
        vertical: Get.height * 0.07,  
      ),  
      child: SingleChildScrollView(  
        child: Form(  
          key: _formKey,  
          child: Column(  
            children: <Widget>[  
              Hero(  
                tag: 'logo',  
                child: SizedBox(  
                  height: Get.height * .3,  
                  child: Image.asset('assets/icons/icon.png'),  
                ),  
              ),  
              SizedBox(  

```

```
height: Get.height * .05,
),
Padding(
padding: const EdgeInsets.all(8.0),
child: Row(
children: [
Flexible(
child: TextFormField(
textInputAction: TextInputAction.next,
autovalidateMode: AutovalidateMode.onUserInteraction,
validator: (value) {
if (!GetUtils.isAlphabetOnly(value) ||
value.length > 20) {
return 'Not a valid First Name';
}
_fullName = value;
return null;
},
textAlign: TextAlign.center,
keyboardType: TextInputType.text,
decoration: style.kInputDecoration.copyWith(
hintText: "",
labelText: 'First Name',
),
),
),
),
const SizedBox(width: 20),
```



```
Flexible(  
  child: TextFormField(  
    textInputAction: TextInputAction.next,  
    autovalidateMode:  
      AutovalidateMode.onUserInteraction,  
    validator: (value) {  
      if (!GetUtils.isAlphabetOnly(value) ||  
        value.length > 20) {  
        return 'Not a valid Last Name';  
      }  
      _fullName += " $value";  
      return null;  
    },  
    textAlign: TextAlign.center,  
    keyboardType: TextInputType.text,  
    decoration: style.kInputDecoration.copyWith(  
      hintText: "", labelText: 'Last Name')),  
  ),  
],  
,  
),  
),  
Padding(  
  padding: const EdgeInsets.all(8.0),  
  child: TextFormField(  
    textInputAction: TextInputAction.next,  
    validator: (value) {  
      if (!GetUtils.isEmail(value)) {
```

```
        return 'Please enter a valid email-address';
    }
    _email = value;
    return null;
},
autovalidateMode: AutovalidateMode.onUserInteraction,
keyboardType: TextInputType.emailAddress,
textAlign: TextAlign.center,
style: const TextStyle(fontSize: 20),
decoration: style.kInputDecoration,
),
),
Obx(
  () => Padding(
    padding: const EdgeInsets.all(8.0),
    child: TextFormField(
      textInputAction: TextInputAction.next,
      autovalidateMode: AutovalidateMode.onUserInteraction,
      validator: (value) {
        if (!GetUtils.isLengthBetween(value, 8, 12)) {
          return 'Please enter a valid password';
        }
        _password = value;
        return null;
      },
      obscureText: !_showPassword.value,
      textAlign: TextAlign.center,
```

Page 75

```
    },  
    keyboardType: TextInputType.phone,  
    textAlign: TextAlign.center,  
    style: const TextStyle(fontSize: 20),  
    decoration: style.kInputDecoration.copyWith(  
      hintText: 'Enter your Phone Number',  
      labelText: 'Phone Number',  
    ),  
  ),  
),  
Obx(() {  
  if (controller.currentState.value == AppState.loading) {  
    return const Center(child: CircularProgressIndicator());  
  } else if (controller.currentState.value == AppState.loaded) {  
    Future.delayed(  
      Duration.zero,  
      () {  
        Get.offAllNamed(Routes.MAP);  
      },  
    );  
    return const SizedBox.shrink();  
  } else {  
    return Text(controller.data);  
  }  
}),  
Button(  
  isTextOnly: false,
```


```
text: 'Sign Up',
icon: Icons.lock_open,
onPressed: () async {
  if (_formKey.currentState.validate()) {
    final LocationData locationData =
      await sendLocationData();
    controller.signUp(
      username: _fullName,
      phonenum: int.parse(_phoneNum),
      email: _email,
      password: _password,
      longitude: locationData.longitude,
      latitude: locationData.latitude,
    );
  } else {}
},
),
],
),
),
),
),
);
}
```

REGISTER_CONTROLLER.DART

```
import 'package:flutter/foundation.dart';
import 'package:get/get.dart';

import '../constants/constants.dart';
import '../data/models/failure_model.dart';
import '../data/repository/repository.dart';
import '../services/services.dart';

class RegisterController extends GetxController {
  Repository authRepository;
  RegisterController({@required this.authRepository});

  final currentState = AppState.initial.obs;
  String data = 'Press the button 
```



```
username: username,
phonenum: phonenum,
longitude: longitude,
latitude: latitude,
email: email,
password: password,
);
_storage.box.write(storageKey, body["access_token"]);
currentState.value = AppState.loaded;
} on Failure catch (f) {
  data = f.toString();
  currentState.value = AppState.failure;
}
}
}
```

REGISTER_BINDING.DART

```
import 'package:get/get.dart';

import '../data/providers/api_client.dart';
import '../data/repository/repository.dart';
import 'register_controller.dart';

class RegisterBinding extends Bindings {
  @override
  void dependencies() {
    Get.lazyPut<RegisterController>(
```

```
() => RegisterController(  
  authRepository: Repository(  
    apiClient: ApiClient(),  
  ),  
),  
);  
}
```

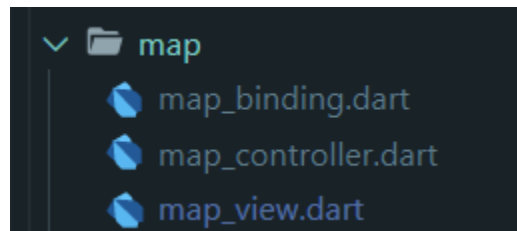


Fig 7.10: Map Folder Layout

MAP_VIEW.DART

```
import 'dart:async';  
  
import 'package:flutter/material.dart';  
import 'package:flutter/services.dart' show rootBundle;  
import 'package:get/get.dart';  
import 'package:get_storage/get_storage.dart';  
import 'package:google_maps_flutter/google_maps_flutter.dart';  
  
import '../constants/constants.dart';  
import '../routes/app_pages.dart';
```



```
import 'map_controller.dart';

class MapView extends GetView<MapController> {
  final Completer<GoogleMapController> _controller = Completer();
  @override
  Widget build(BuildContext context) {
    GoogleMapController mapController;
    String _mapStyle;
    rootBundle.loadString('assets/map_style.txt').then((string) {
      _mapStyle = string;
    });
    return Scaffold(
      appBar: AppBar(
        title: const Text("Hotspots Near You ☺"),
        centerTitle: true,
        actions: [
          IconButton(
            icon: const Icon(Icons.login),
            onPressed: () async {
              final GetStorage box = GetStorage();
              await box.remove("_accessToken");
              Get.offAllNamed(Routes.LOGIN);
            },
          ),
        ],
      ),
      body: SafeArea(
```

```
// child: Obx(
//   () => !controller.isLoading.value
//   ? const Center(child: CircularProgressIndicator())
//   : GoogleMap(
//     initialCameraPosition: CameraPosition(
//       target: LatLng(
//         controller.locationData.latitude,
//         controller.locationData.longitude,
//       ),
//       zoom: 17,
//     ),
//     onMapCreated: (GoogleMapController controller) {
//       _controller.complete(controller);
//       mapController = controller;
//       mapController.setMapStyle(_mapStyle);
//     },
//     circles: Set<Circle>.of(controller.circleList.values),
//     myLocationEnabled: true,
//     onLongPress: (argument) {
//       controller.isLoading.value = false;
//       controller.getHotspotList();
//     },
//   ),
// ),
child: Obx(() {
  if (controller.currentState.value == AppState.initial) {
    return Text(controller.data);
```

```
} else if (controller.currentState.value == AppState.loading) {
    return const CircularProgressIndicator();
} else if (controller.currentState.value == AppState.loaded) {
    return GoogleMap(
        initialCameraPosition: CameraPosition(
            target: LatLng(
                controller.locationData.latitude,
                controller.locationData.longitude,
            ),
            zoom: 17,
        ),
        onMapCreated: (GoogleMapController controller) {
            _controller.complete(controller);
            mapController = controller;
            mapController.setMapStyle(_mapStyle);
        },
        circles: Set<Circle>.of(controller.circleList.values),
        myLocationEnabled: true,
        // onLongPress: (argument) {
        //   controller.isLoading.value = false;
        //   controller.getHotspotList();
        // },
    );
} else {
    return Text(controller.data);
}
}),
```

```
    ),  
  );  
}  
}
```

MAP_CONTROLLER.DART

```
import 'dart:async';  
import 'dart:collection';  
  
import 'package:flutter/material.dart';  
import 'package:get/get.dart';  
import 'package:google_maps_flutter/google_maps_flutter.dart';  
import 'package:location/location.dart';  
  
import '../constants/constants.dart';  
import '../data/models/failure_model.dart';  
import '../data/models/hotspot_model.dart';  
import '../data/repository/repository.dart';  
import '../services/services.dart';  
import '../shared/info_dialog.dart';  
import '../shared/location_data_sender.dart';  
  
class MapController extends GetxController {  
  final Repository repository;  
  MapController({ @required this.repository });  
  final currentState = AppState.initial.obs;  
  String data = 'Initial';
```

```
LocationData locationData;

final circleList = HashMap<CircleId, Circle>();

HotSpotModel hotspotList;

@Override

Future<void> onInit() async {
    await getHotspotList();
    super.onInit();
}

Future<void> getHotspotList() async {
    currentState.value = AppState.loading;

    locationData = await sendLocationData();

    if (!locationData.isNullOrBlank) {
        try {
            final _storage = StorageService().instance;
            final body = await repository.getHotSpotZones(
                latitude: locationData.latitude,
                longitude: locationData.longitude,
                accessToken: await _storage.box.read(storageKey),
            );
            hotspotList = HotSpotModel.fromJson(body as Map<String, dynamic>);
            _storage.box.write(storageKey, body["access_token"]);
        } on Failure catch (f) {
            data = f.toString();
        }
    }
}
```

```
currentState.value = AppState.failure;
}

for (final CoronaHotspot element in hotspotList.coronaHotspot) {
    final CircleId circleId =
        CircleId(hotspotList.coronaHotspot.indexOf(element).toString());
    circleList[circleId] = Circle(
        circleId: circleId,
        center: LatLng(
            element.lat,
            element.long,
        ),
        radius: element.active / 5,
        fillColor: Colors.redAccent.withOpacity(element.death / 100),
        strokeColor: Colors.redAccent.withOpacity(0.2),
        // strokeWidth: 20,
        consumeTapEvents: true,
        onTap: () async {
            Get.defaultDialog(
                title: '${element.lat},${element.long}',
                content: Column(
                    children: [
                        InfoDialog(
                            title: "Active",
                            imagePath: "031-medical-mask.svg",
                            hotspotInfo: element.active,
                        ),
                    ],
                ),
            );
        },
    );
}
```

```
        InfoDialog(  
            title: "Dead",  
            imagePath: "025-No.svg",  
            hotspotInfo: element.death,  
        ),  
        InfoDialog(  
            title: "Recovered",  
            imagePath: "044-immunity.svg",  
            hotspotInfo: element.recovered,  
        ),  
    ],  
),  
);  
},  
);  
}  
  
for (final CrowdHotspot element in hotspotList.crowdHotspot) {  
    final CircleId circleId =  
        CircleId(hotspotList.crowdHotspot.indexOf(element).toString());  
    circleList[circleId] = Circle(  
        circleId: circleId,  
        center: LatLng(  
            element.lat,  
            element.long,  
        ),  
        radius: 40,  
    );  
}
```

```
        fillColor: Colors.black.withOpacity(0.5),
        strokeColor: Colors.black45.withOpacity(0.2),
      );
    }
    currentState.value = AppState.loaded;
  }
}
```

MAP_BINDING.DART

```
import 'package:get/get.dart';
import '../data/providers/api_client.dart';
import '../data/repository/repository.dart';
```

```
import 'map_controller.dart';
```

```
class MapBinding extends Bindings {
  @override
  void dependencies() {
    Get.lazyPut<MapController>(
      () => MapController(
        repository: Repository(
          apiClient: ApiClient(),
        ),
      ),
    );
  }
};
```



```
}  
}
```

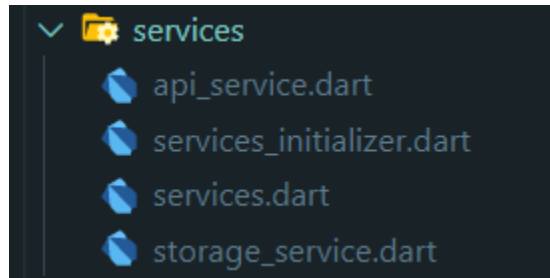


Fig 7.11: Services Folder Layout

API_SERVICE.DART

```
import 'package:dio/dio.dart';  
import 'package:get/get.dart';  
  
import '../constants/api_constants.dart';  
  
class ApiService extends GetxService {  
  ApiService get instance => Get.find();  
  
  final dio = Dio(  
    BaseOptions(  
      baseUrl: baseUrl,  
      connectTimeout: 5000,  
      receiveTimeout: 3000,  
    ),  
  );  
}
```

```
}
```

Service_initializer.dart

```
import 'package:get/get.dart';
```

```
import 'services.dart';
```

```
Future<void> initServices() async {  
  await Get.putAsync(() => StorageService().initialize());  
  Get.put(ApiService());  
}
```

STORAGE_SERVICE.DART

```
import 'package:get/get.dart';
```

```
import 'package:get_storage/get_storage.dart';
```

```
class StorageService extends GetxService {  
  StorageService get instance => Get.find();  
  GetStorage box;
```

```
Future<StorageService> initialize() async {  
  await GetStorage.init();  
  box = GetStorage();  
  return this;  
}
```

```
// GetStorage storageBox() {  
//   final GetStorage box = GetStorage();
```

```
// return box;  
// }  
}
```

SERVER SIDE CODE

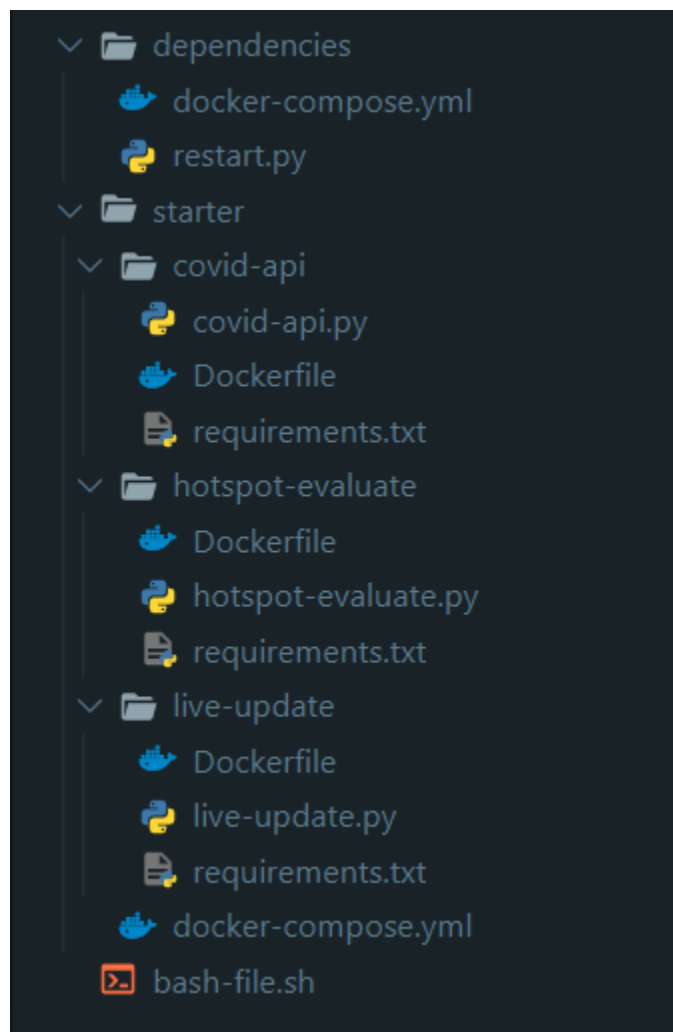


Fig 7.12: Server Folder Layout

DOCKER-COMPOSE.YML

version: '3'

services:

zookeeper:

image: wurstmeister/zookeeper

container_name: zookeeper

ports:

- "2181:2181"

kafka:

image: wurstmeister/kafka

container_name: kafka

ports:

- "9092:9092"

environment:

KAFKA_ADVERTISED_HOST_NAME: localhost

KAFKA_ZOOKEEPER_CONNECT: zookeeper:2181

postgresql:

image: postgres

container_name: post-pandemic-db

ports:

- "5432:5432"

volumes:

- postgres_data:/var/lib/postgresql/data/

environment:

PGDATA: /var/postgres_data

POSTGRES_USER: postgres

POSTGRES_PASSWORD: 7878

POSTGRES_DB: post-pandemic-db

volumes:

postgres_data:

external: true

RESTART.PY

```
import psycopg2
```

```
import random
```

```
import string
```

```
from kafka.admin import KafkaAdminClient, NewTopic
```

```
connection = psycopg2.connect(host="localhost", port=5432,  
                             database="post-pandemic-db", user="postgres", password="7878")
```

```
cursor = connection.cursor()
```

```
admin_client = KafkaAdminClient(bootstrap_servers="localhost:9092")
```

```
topic_list = []
```

```
topic_list.append(NewTopic(name='get-hotspot-in', num_partitions=1, replication_factor=1))
```

```
topic_list.append(NewTopic(name='get-hotspot-out', num_partitions=1, replication_factor=1))
```

```
admin_client.create_topics(new_topics=topic_list, validate_only=False)
```

```
class UserModel:
    def __init__(self):
        pass
    def create_user_table(self):
        cursor.execute("DROP TABLE IF EXISTS User_Data")
        sql = "CREATE TABLE User_Data(
            id SERIAL PRIMARY KEY,
            username VARCHAR(30),
            phone_no BIGSERIAL,
            password TEXT NOT NULL,
            email TEXT NOT NULL,
            lat FLOAT,
            long FLOAT
        )"
        cursor.execute(sql)
        connection.commit()
        print("Table Created ..")
    def create_dummy_data(self):
        for i in range(10000):
            username = "
            for j in range(random.choice([3,7,9,5])):
                username+=random.choice(list(string.ascii_lowercase))
            phone_no = random.randint(9000000000,9999999999)
            password = "
            for j in range(random.choice([13,7,9,10])):
                password+=random.choice(list(string.ascii_lowercase))
            email = "
```

```
for j in range(random.choice([3,7,9,5])):
    email+=random.choice(list(string.ascii_lowercase))
    email+='@gmail.com'
    lat = round(random.uniform(28.4567,28.8902),4)
    long = round(random.uniform(77.0012,77.3456),4)
    sql = "INSERT INTO User_Data (username,phone_no,password,email,lat,long)
VALUES(%s,%s,%s,%s,%s,%s)"
    cursor.execute(sql,(username,phone_no,password,email,lat,long))
    connection.commit()
for i in range(10000):
    lat = round(random.uniform(28.4567,28.8902),4)
    long = round(random.uniform(77.0012,77.3456),4)
    death = random.randint(0,100)
    active = random.randint(0,100)
    recovered = random.randint(0,100)
    sql = "INSERT INTO Hotspot (lat,long,death,active,recovered)
VALUES(%s,%s,%s,%s,%s)"
    cursor.execute(sql,(lat,long,death,active,recovered))
    connection.commit()
print('Dummy Data Inserted ..')

def create_hotspot_table(self):
    cursor.execute("DROP TABLE IF EXISTS Hotspot")
    sql = "CREATE TABLE Hotspot(
        id SERIAL PRIMARY KEY,
        lat FLOAT,
        long FLOAT,
        death SERIAL,
```

```
        active SERIAL,
        recovered SERIAL
    )"

    cursor.execute(sql)
    connection.commit()
    print('Table Created ..')

def create_data():
    user = UserModel()
    user.create_user_table()
    user.create_hotspot_table()
    user.create_dummy_data()

# inp = input("Do you want create data? (y/n): ")
# if inp == 'y':
#     create_data()
# else:
#     print('kafka-topic created without data')
create_data()
```

STARTER/DOCKER-COMPOSE.YML

```
version: '3'

services:
    api:
        build: covid-api/.
        container_name: covid-api
```



```
network_mode: "host"
```

```
live-update:
```

```
  build: live-update/.
```

```
  container_name: live-update
```

```
  network_mode: "host"
```

```
hotspot-evaluate:
```

```
  build: hotspot-evaluate/.
```

```
  container_name: hotspot-evaluate
```

```
  network_mode: "host"
```

COVID-APL.PY

```
from datetime import datetime, timedelta
from typing import Optional
import uvicorn
from fastapi import Depends, FastAPI, HTTPException, status
from fastapi.security import OAuth2PasswordBearer
from jose import jwt
from passlib.context import CryptContext
from pydantic import BaseModel
from fastapi.middleware.cors import CORSMiddleware
import pycpg2
from kafka import KafkaProducer
from kafka import KafkaConsumer
```

```
consumer = KafkaConsumer("get-hotspot-out")
producer = KafkaProducer(bootstrap_servers='localhost:9092')
# import time

# time.sleep(20)
print("Running Api")

connection = psycopg2.connect(host="localhost", port=5432,
                              database="post-pandemic-db", user="postgres", password="7878")
cursor = connection.cursor()

oauth2_scheme = OAuth2PasswordBearer(tokenUrl= 'token')

app = FastAPI()

app.add_middleware(
    CORSMiddleware,
    allow_origins=["*"],
    allow_credentials=True,
    allow_methods=["*"],
    allow_headers=["*"],
)

SECRET_KEY = "09d25e094faa6ca2556c818166b7a9563b93f7099f6f0f4caa6cf63b88e8d3e7"
ALGORITHM = "HS256"
```

```
ACCESS_TOKEN_EXPIRE_HOURS = 100
```

```
pwd_context = CryptContext(schemes=["bcrypt"], deprecated="auto")
```

```
class Token(BaseModel):
```

```
    access_token: str
```

```
    token_type: str
```

```
class User(BaseModel):
```

```
    username: str
```

```
    phone_no: int
```

```
    password: str
```

```
    email: Optional[str] = None
```

```
    lat: float
```

```
    longi : float
```

```
    disabled: Optional[bool] = None
```

```
class Login_user(BaseModel):
```

```
    email : str
```

```
    password : str
```

```
class Action(BaseModel):
```

```
    lat : float
```

```
    longi : float
```

```
    access_token: str
```

```
def add_user(user_tuple):
```

```
sql = "INSERT INTO User_Data (username,phone_no,password,email,lat,long)
VALUES(%s,%s,%s,%s,%s,%s)"
```

```
cursor.execute(sql,user_tuple)
```

```
connection.commit()
```

```
def verify_password(plain_password, hashed_password):
```

```
    return pwd_context.verify(plain_password, hashed_password)
```

```
def get_password_hash(password):
```

```
    return pwd_context.hash(password)
```

```
def create_access_token(data: dict, expires_delta: Optional[timedelta] = None):
```

```
    to_encode = data.copy()
```

```
    if expires_delta:
```

```
        expire = datetime.utcnow() + expires_delta
```

```
    else:
```

```
        expire = datetime.utcnow() + timedelta(minutes=15)
```

```
    to_encode.update({"exp": expire})
```

```
    encoded_jwt = jwt.encode(to_encode, SECRET_KEY, algorithm=ALGORITHM)
```

```
    return encoded_jwt
```

```
def update_user(lat,longi,email):
```

```
    sql_update = f"UPDATE User_Data SET lat = {round(lat,4)},long = {round(longi,4)}
WHERE email = '{email}'"
```

```
    cursor.execute(sql_update)
```

```
    connection.commit()
```

```
@app.post('/signup',response_model = Token)
```

```
async def signup_get_token(user:User):
    cursor.execute(f"SELECT email FROM User_Data WHERE email = '{user.email}'")
    user_db = cursor.fetchall()
    if user_db != []:
        raise HTTPException(
            status_code=status.HTTP_306_RESERVED,
            detail="Email is already registered !!",
            headers={"WWW-Authenticate": "Bearer"},
        )
    user_tuple =
    (user.username,user.phone_no,get_password_hash(user.password),user.email,user.lat,user.longi)
    add_user(user_tuple)
    access_token_expires = timedelta(hours=ACCESS_TOKEN_EXPIRE_HOURS)
    access_token = create_access_token(
        data={"sub": user.email}, expires_delta=access_token_expires
    )
    return {"access_token": access_token, "token_type": "Bearer"}
```

```
@app.post("/login", response_model=Token)
```

```
async def login_for_access_token(form_data: Login_user):
```

```
    cursor.execute(f"SELECT email,password FROM User_Data WHERE email = '{form_data.email}'")
```

```
    user_db = cursor.fetchall()
```

```
    if user_db == []:
```

```
        raise HTTPException(
```

```
            status_code=status.HTTP_401_UNAUTHORIZED,
```

```
            detail="Incorrect email",
```

```
            headers={"WWW-Authenticate": "Bearer"},
```

```

    )
    if verify_password(form_data.password,user_db[0][1]) == False:
        raise HTTPException(
            status_code=status.HTTP_401_UNAUTHORIZED,
            detail="Incorrect password",
            headers={"WWW-Authenticate": "Bearer"},
        )
    access_token_expires = timedelta(hours=ACCESS_TOKEN_EXPIRE_HOURS)
    access_token = create_access_token(
        data={"sub": form_data.email}, expires_delta=access_token_expires
    )
    return {"access_token": access_token, "token_type": "Bearer"}

@app.post("/covid")
async def get_covid_hotspot(action : Action):
    user = jwt.decode(action.access_token,key=SECRET_KEY,algorithms=ALGORITHM)
    cursor.execute(f"SELECT email FROM User_Data WHERE email = '{user['sub']}'")
    user_db = cursor.fetchall()
    if user_db == []:
        raise HTTPException(
            status_code=status.HTTP_401_UNAUTHORIZED,
            detail="You are not Authorized",
            headers={"WWW-Authenticate": "Bearer"},
        )
    update_user(action.lat,action.longi,user['sub'])
    producer.send("get-hotspot-in",str(str(action.lat)+","+str(action.longi)).encode("utf-8"))
    for message in consumer:

```

```
hotspots=message.value.decode("utf-8").split("-")
return_data = list(eval(hotspots[0]))
crowd_data = list(eval(hotspots[1]))
access_token_expires = timedelta(hours=ACCESS_TOKEN_EXPIRE_HOURS)
access_token = create_access_token(
    data={"sub": user['sub']}, expires_delta=access_token_expires
)
return
{"corona_hotspot":return_data,"crowd_hotspot":crowd_data,"access_token":access_token}

if __name__ == "__main__":
    uvicorn.run(app,host = '0.0.0.0', port = 8000)
```

DOCKERFILE

```
FROM python:3.7.6-stretch
WORKDIR /covid-api
ADD . /covid-api
RUN pip install --upgrade pip
RUN pip install -r requirements.txt
EXPOSE 8000
CMD ["python3", "covid-api.py"]
```

HOTSPOT-EVALUATE.PY

```
import psycpg2
from sklearn.cluster import KMeans
from kafka import KafkaProducer
from kafka import KafkaConsumer
```

```
consumer = KafkaConsumer("get-hotspot-in")
producer = KafkaProducer(bootstrap_servers='localhost:9092')
connection = psycopg2.connect(host="localhost", port=5432,
                              database="post-pandemic-db", user="postgres", password="7878")
cursor = connection.cursor()

def calculate_covid_hotspot(lat,longi):
    query = f"SELECT lat,long,death,active,recovered FROM Hotspot WHERE (lat BETWEEN {lat-0.05} AND {lat+0.05}) AND (long BETWEEN {longi-0.05} AND {longi+0.05})"
    cursor.execute(query)
    loc_data = cursor.fetchall()
    return_data = []
    if loc_data!= []:
        for i in loc_data:
            return_dict = { }
            return_dict["lat"] = i[0]
            return_dict["long"] = i[1]
            return_dict["death"] = i[2]
            return_dict["active"] = i[3]
            return_dict["recovered"] = i[4]
            return_data.append(return_dict)
    return return_data

def calculate_crowd_hotspot(lat,longi):
    sql = f"SELECT lat,long FROM User_Data WHERE (lat BETWEEN {lat-0.05} AND {lat+0.05}) AND (long BETWEEN {longi-0.05} AND {longi+0.05})"
    cursor.execute(sql)
    crowd_data = cursor.fetchall()
```



```
return_crowd_data = []
kmean=KMeans(n_clusters=10)
if len(crowd_data)>10:
    kmean.fit(crowd_data)
    data = kmean.cluster_centers_.tolist()
    for i in data:
        data_dict = { }
        sql1 = f"SELECT lat,long FROM User_Data WHERE (lat BETWEEN {i[0]-0.01} AND {i[0]+0.01}) AND (long BETWEEN {i[1]-0.01} AND {i[1]+0.01})"
        cursor.execute(sql1)
        crowd_data_medians = cursor.fetchall()
        if len(crowd_data_medians)>25:
            data_dict["lat"] = i[0]
            data_dict["long"] = i[1]
            return_crowd_data.append(data_dict)
return return_crowd_data
```

for message in consumer:

```
location=message.value.decode("utf-8").split(",")
lat=float(location[0])
longi=float(location[1])
covid_hotspot=calculate_covid_hotspot(lat,longi)
crowd_hotspot=calculate_crowd_hotspot(lat,longi)
producer.send("get-hotspot-out",str(str(covid_hotspot)+"-"+str(crowd_hotspot)).encode("utf-8"))
```

LIVE_UPDATE.PY

```
import psycopg2
import random
import time

connection = psycopg2.connect(host="localhost", port=5432,
                             database="post-pandemic-db", user="postgres", password="7878")
cursor = connection.cursor()

while True:
    time.sleep(5)
    sql="Select * from User_Data order by random() limit 100"
    cursor.execute(sql)
    user_records = cursor.fetchall()
    for i in user_records:
        random_users=list(i)
        random_users[5]=round(random.uniform(28.4567,28.8902),4)
        random_users[6]=round(random.uniform(77.0012,77.3456),4)
        cursor.execute("UPDATE User_Data set lat = %s,long=%s where id = %s",
        (random_users[5],random_users[6],random_users[0]))
        connection.commit()

    cursor.close()
    connection.close()
```

7.5 Project User Interface

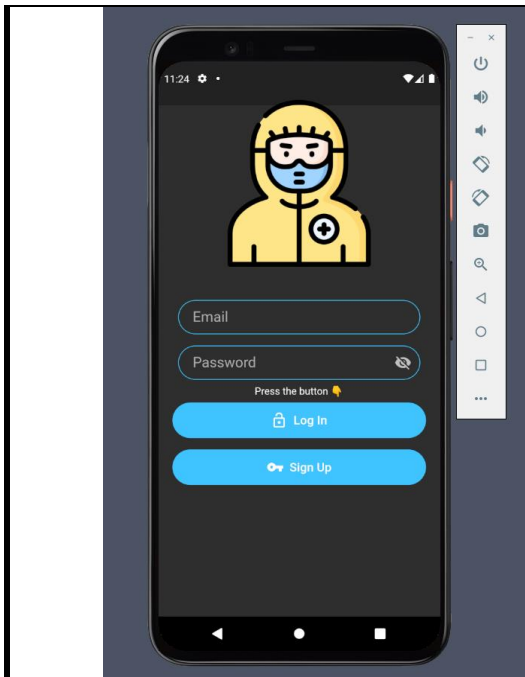


Fig 7.13: Login View

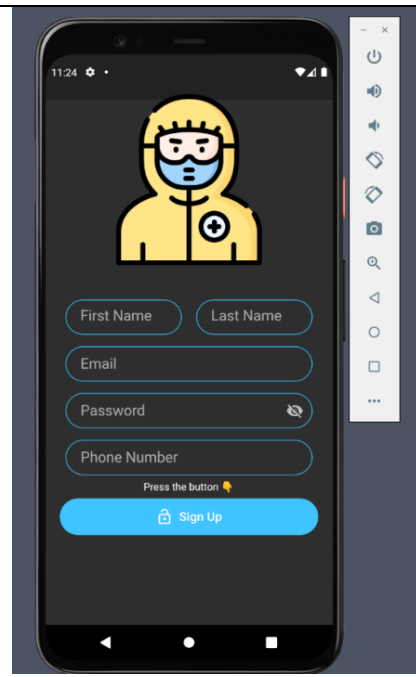


Fig 7.14: Signup View

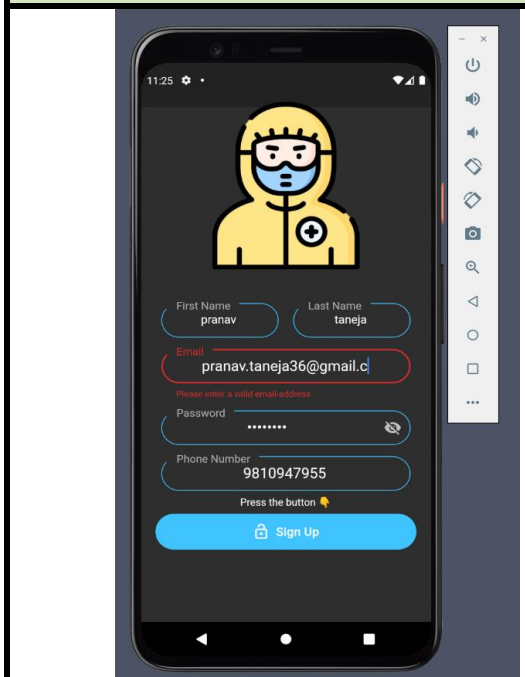


Fig 7.15: Error Detection on Fields

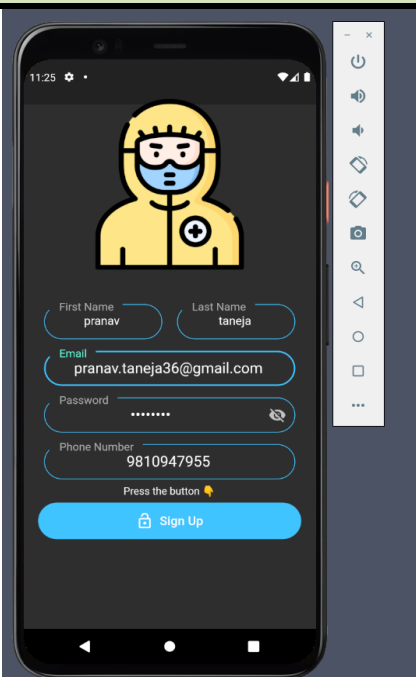


Fig 7.16: Final Signup View

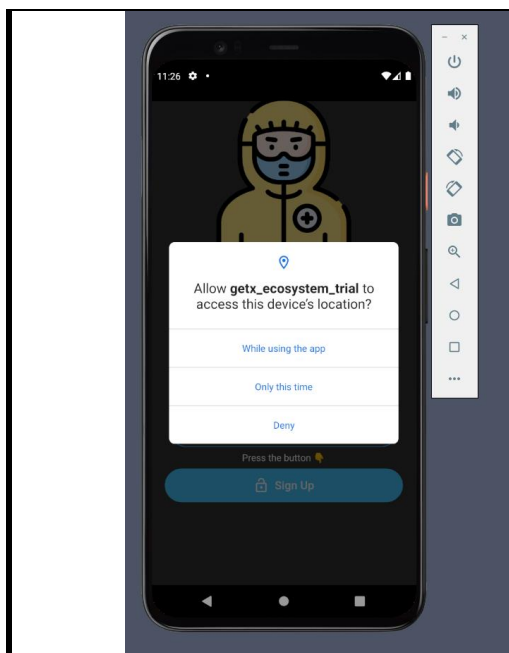


Fig 7.17: GPS Permission Dialog

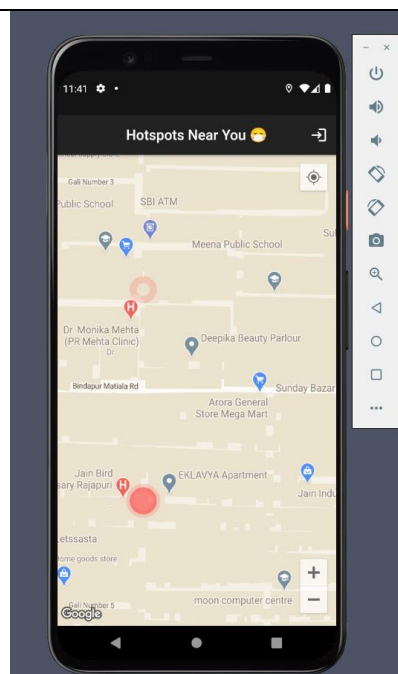


Fig 7.18: Map View with Hotspot Zones

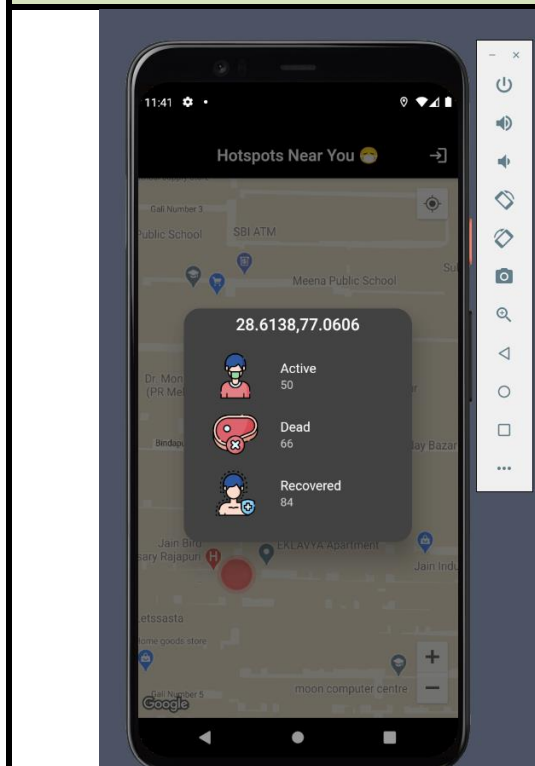


Fig 7.19: Details of Selected Hotspot Zone

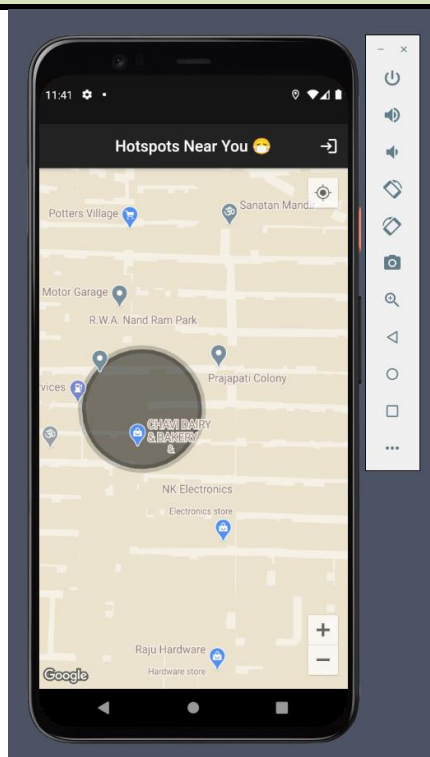
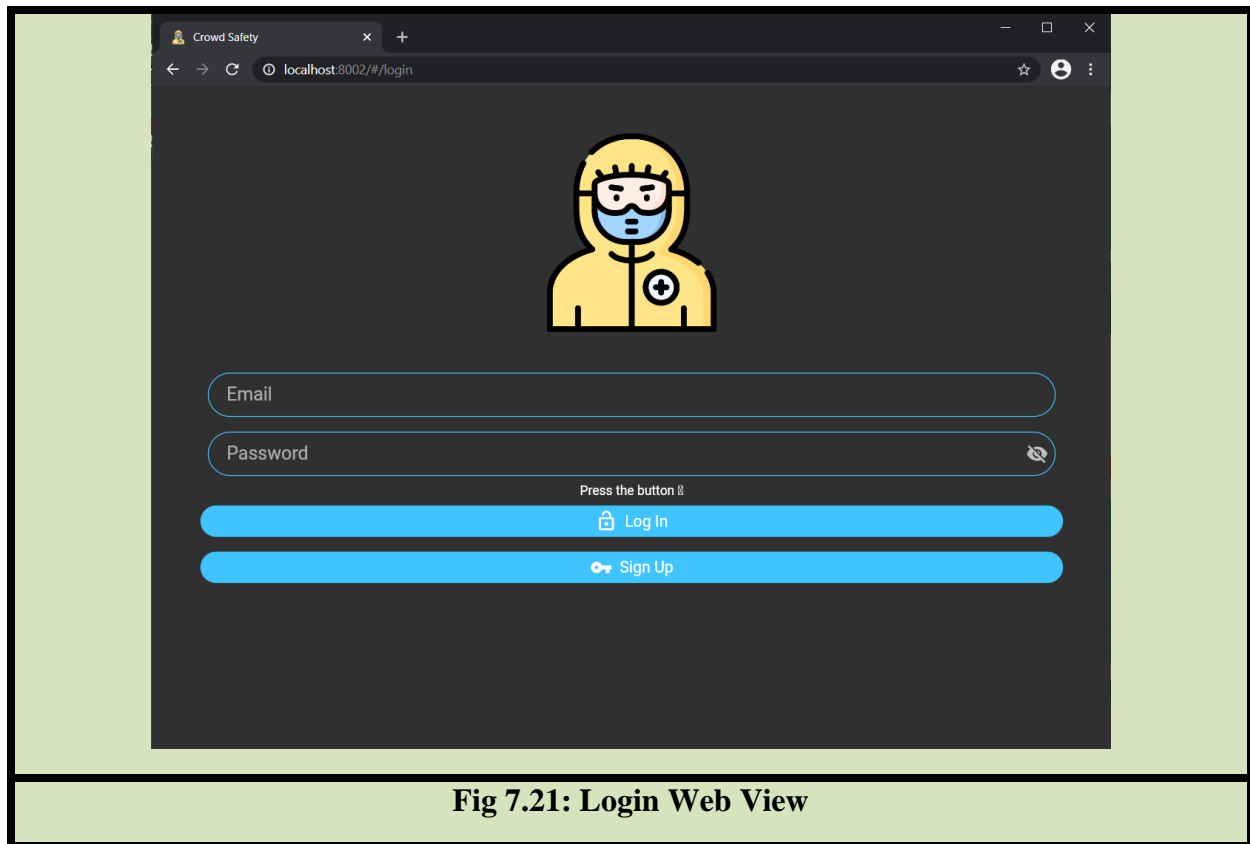


Fig 7.20: Crowded Zones



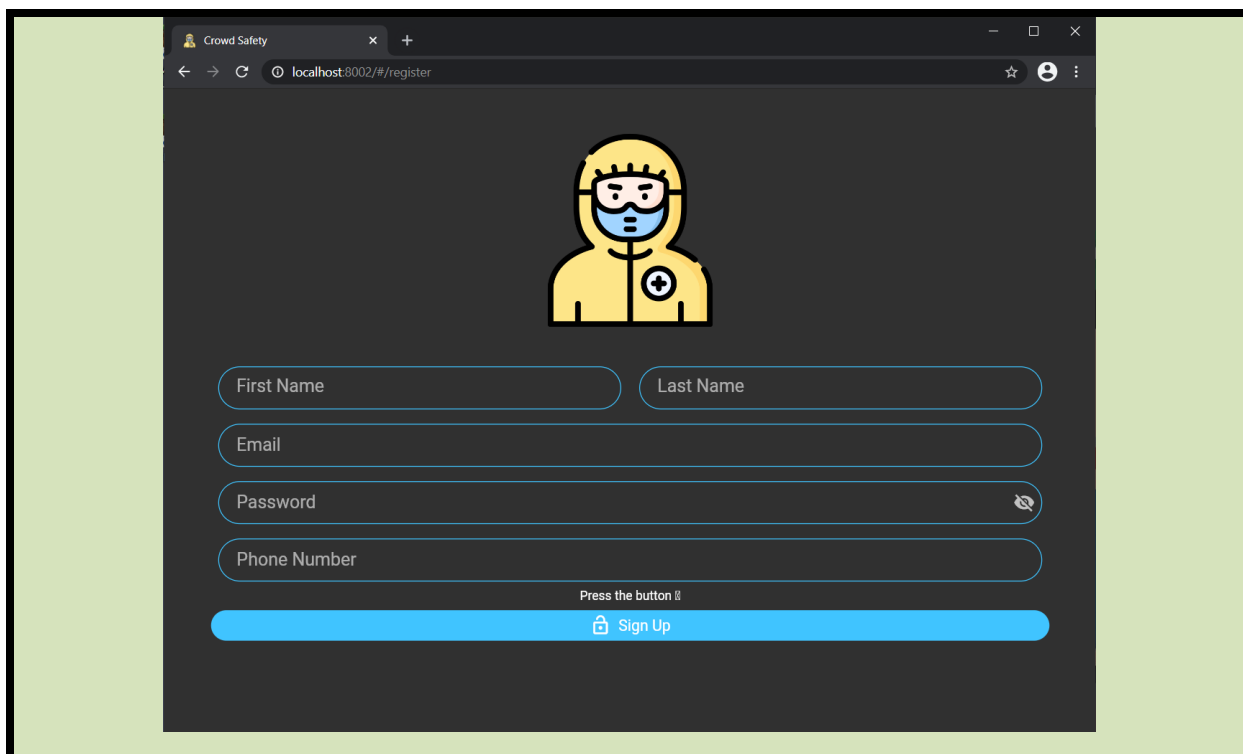


Fig 7.22: Signup Web View

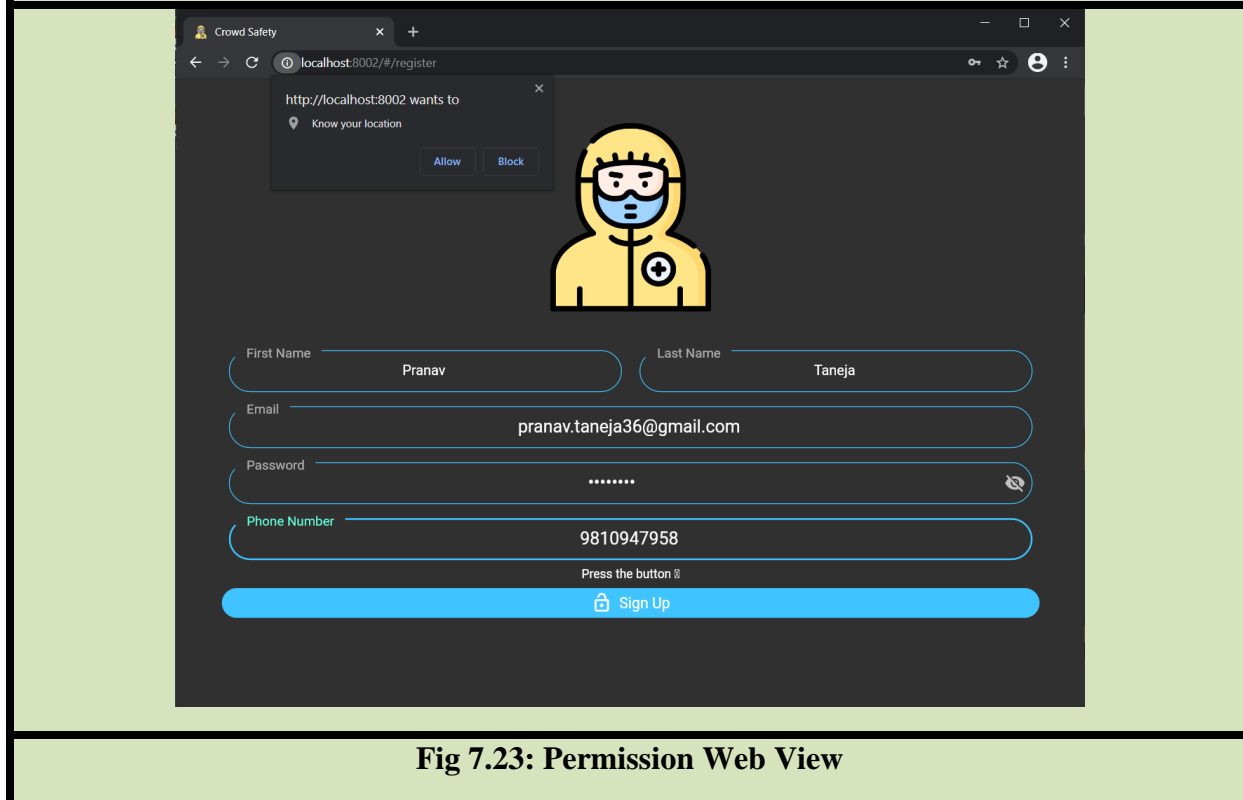


Fig 7.23: Permission Web View

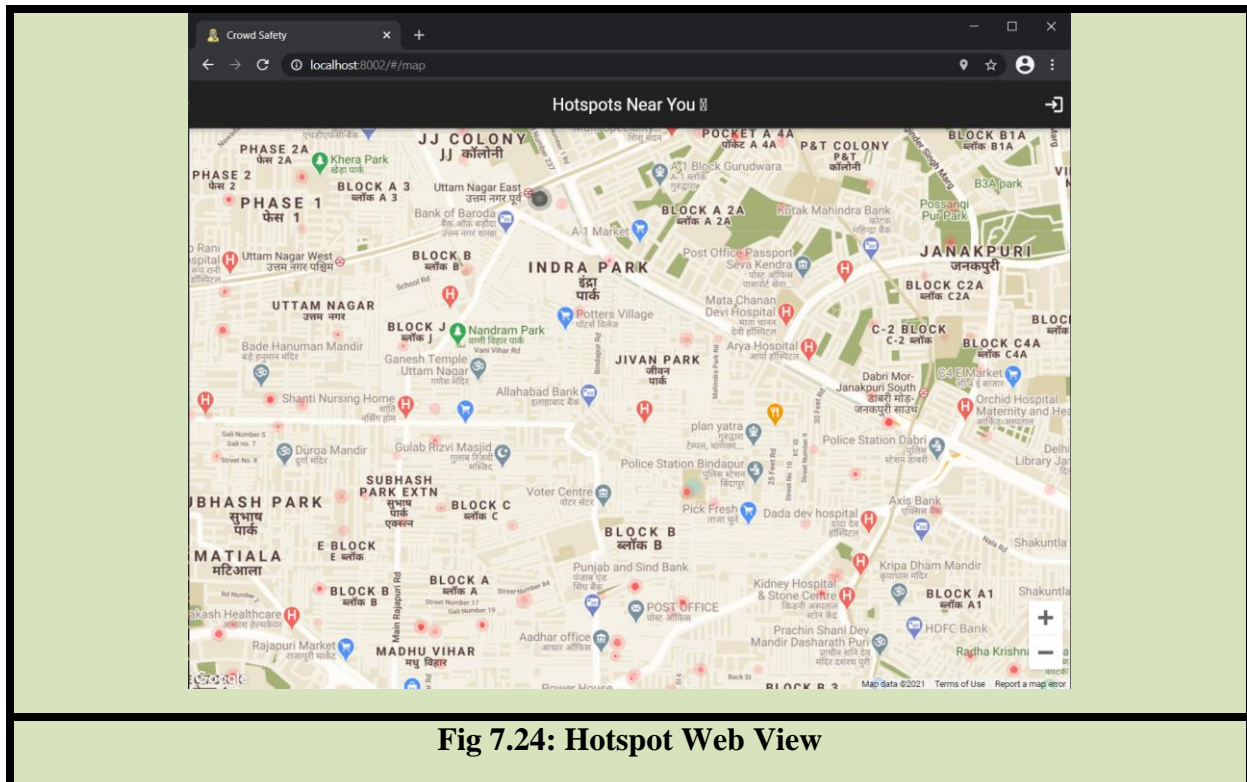


Fig 7.24: Hotspot Web View

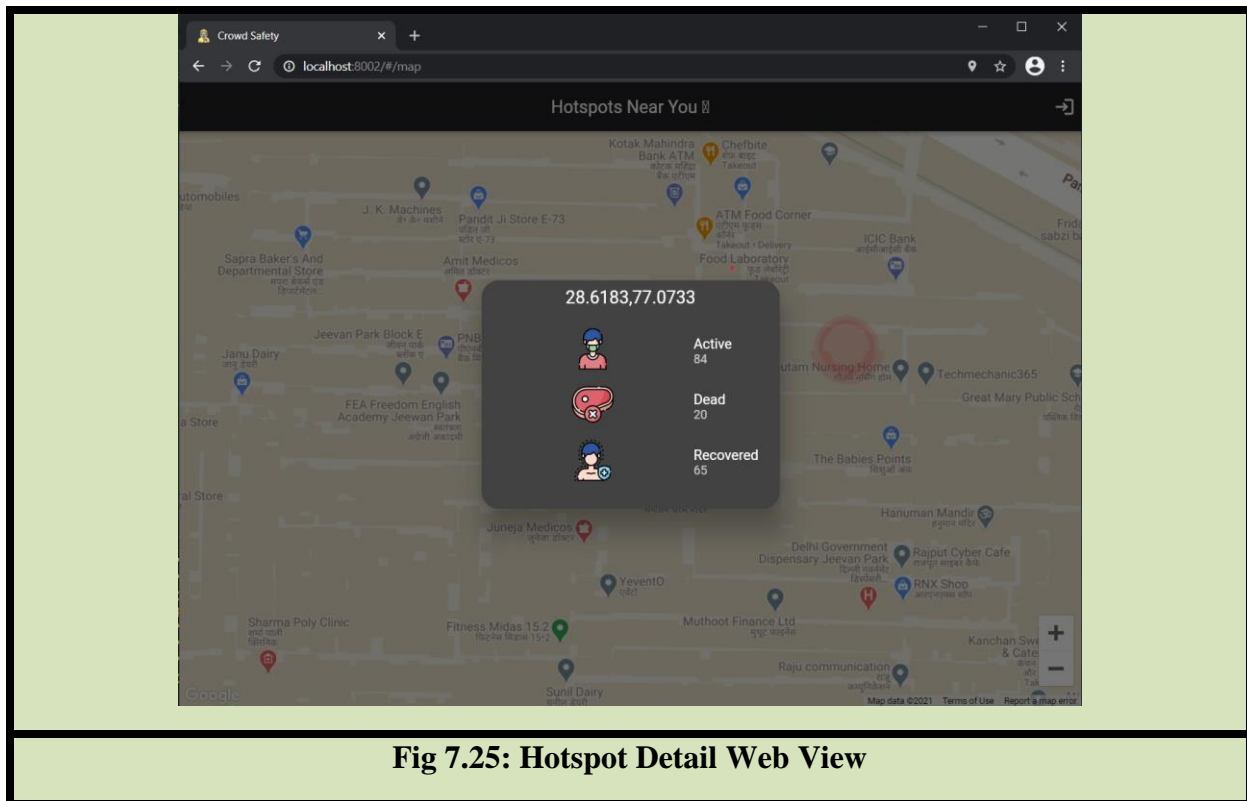
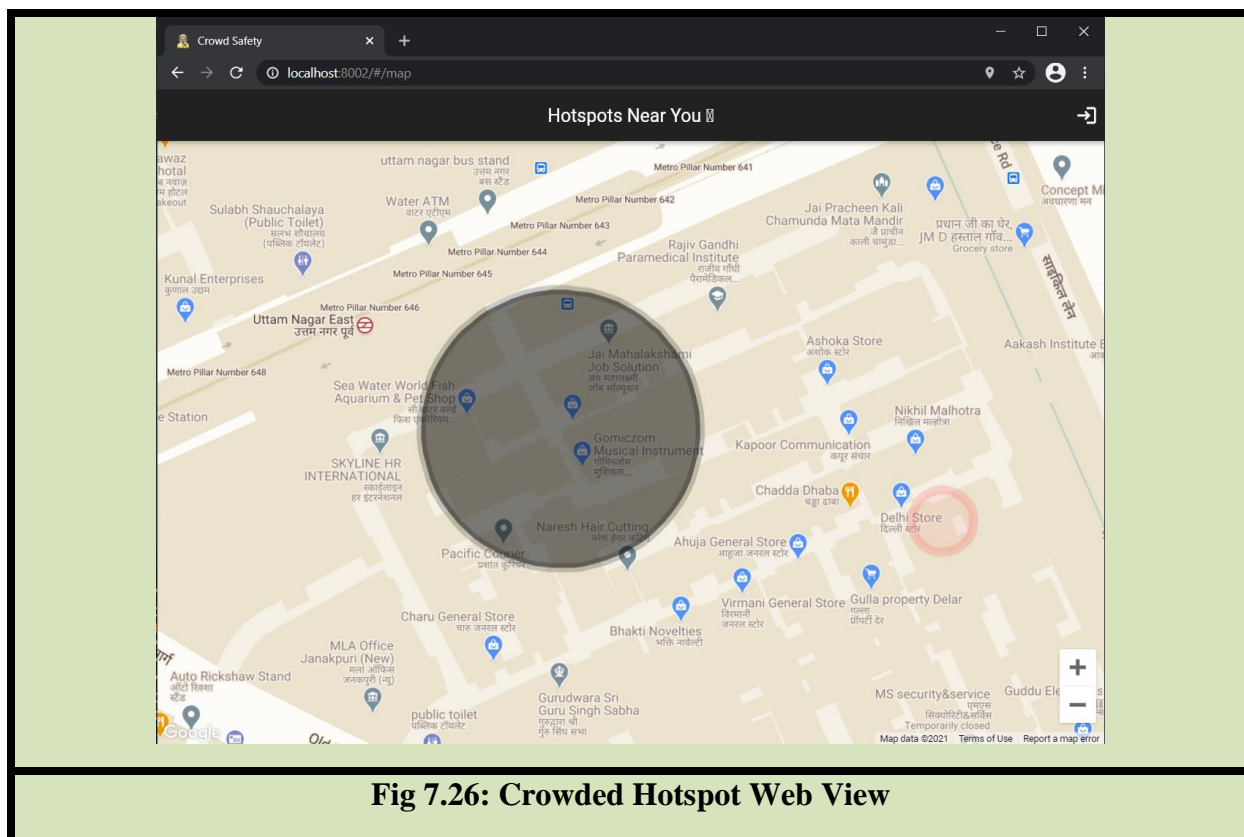


Fig 7.25: Hotspot Detail Web View



7.6 Tables:

Name	Type
id	SERIAL
username	VARCHAR(30)
phone_no	BIGSERIAL
password	TEXT
email	TEXT
lat	FLOAT
long	FLOAT

Table 7.2 User Table.

Name	Type
id	SERIAL
lat	FLOAT
long	FLOAT
death	SERIAL
active	SERIAL
recovered	SERIAL

Table 7.3 Hotspot Table.

CHAPTER 8: TESTING

8.1 Introduction

Testing is the way toward practising programming to find and redress blunders. The target of the testing is to reveal various classes of mistakes and to do as such with a base measure of time and exertion. To give an exceptionally worthy and mistake-free framework, the framework ought to need to confront the testing technique and assessment of every single module and usefulness. (Mustafa and Khan, 2007)

8.2 Need for Testing

Most of the testing phase is done to test and approve the finished frame. On purpose that it would pull out and show how normal it was and all the functionality worked as it was is reflected in the business information structure. Testing does not require the designer to require it test each part of the code to verify bugs and return to existing bugs and bugs found in it but its purpose is to identify conditions that may affect the framework or the size or which could lead to fruitless work. In line with these lines, experiments were performed to improve the environment The design of the product satisfies the client's expectations.

8.3 Test Plan

System upgrade forms usually focus on keeping the strategic distance from errors, identifying and correcting possible and expected product product suspicions unstable quality after development. Testing process, methods and materials important supporters of effective and productive testing and quality assurance. (Mustafa no Khan, 2007). Various assessment methods to be used for the proposed framework are provided the next place.

8.3.1 Type of Testing

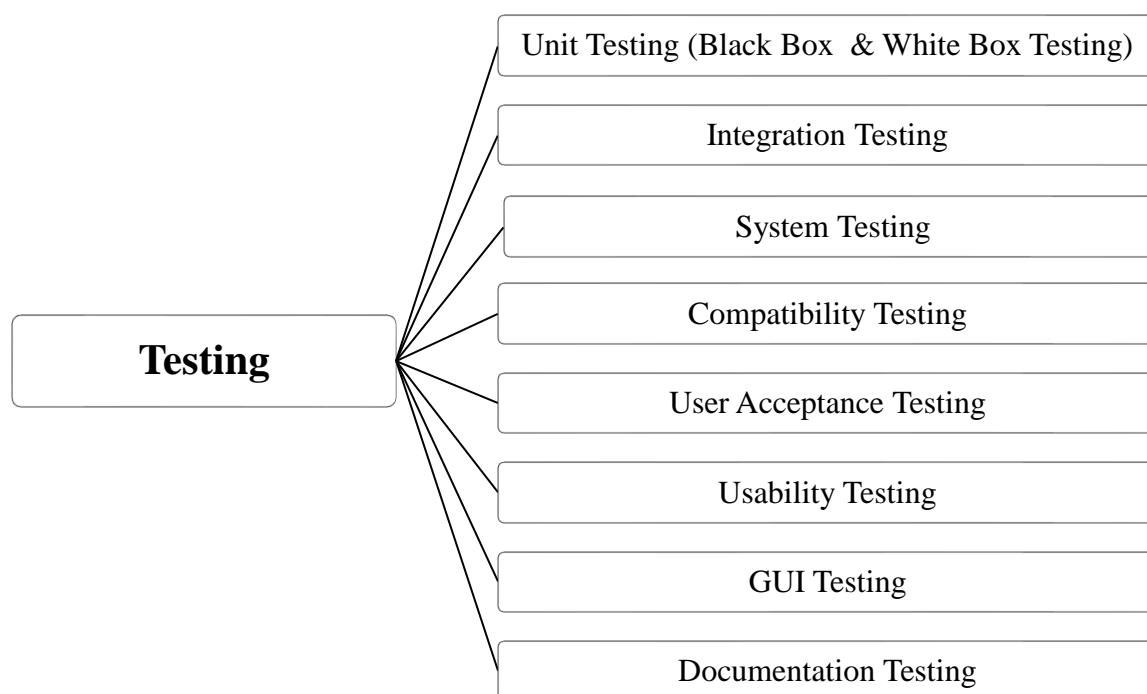


Fig 8.1: Types of Testing

S. No	Testing Technique	Requirement
1.	Unit Testing	Testing of individual programming segment or modules of the framework. This kind of testing is performed by the engineers simply because it requires point by point information on the interior program plan and code. It incorporates performing black box and white box testing.

2.	Integration Testing	Testing of reconciliation modules to check joined usefulness after incorporation. Different modules are incorporated together and afterwards tried by the engineer of the framework to check whether the modules work appropriately after joining.
3.	System Testing	The whole framework is tried by the engineer according to the necessities. Every one of the modules of the framework is joined and check whether they work flawlessly in the wake of being coordinated into one system as a whole.
4.	Compatibility Testing	Similarity testing is utilized to decide whether the framework causes any issues identified with how it works in worry with the working framework and diverse kind of framework equipment and programming. The engineer plays out this kind of testing on different equipment/programming stage to test the similarity of the framework on every stage.
5.	GUI Testing	The testing is to be done to check the plan of the framework whether it fulfils the six predefined HCIU structure guideline and whether the framework is easy to understand.
6.	Usability Testing	The reason for doing ease of use test is to decide if an item or record works with its expected clients or perusers. This is to be finished by assessing an item by watching the genuine individuals taking a shot at or utilizing it to improve or upgrade the framework. The end-clients execute the framework to test it
7.	User Acceptance Testing	Client Acceptance Testing tests that the framework fulfils all the fundamental client prerequisites and is performing to the satisfactory level true to form by the client. This is finished by characterizing a lot of acknowledgement criteria which the framework must fulfil before the client will acknowledge it.

8.	Documentation Testing	Documentation Testing is to be done to confirm the framework documentation whether it is significant and easy to use or not.
----	------------------------------	--

Table 8.1: Testing Techniques**8.3.2 Pass/Fail Criteria**

- ✍ The produced system meets all the specifications mentioned in the project specification form and all the requirements of the user gathered during the research phase.
- ✍ All the modules/ functionalities are implemented in a way that the desired system produced runs without any bugs and quickly.
- ✍ The system is to be verified for the usability that depends upon proven Human-Computer Interaction (HCI) principles and aims to achieve in the application ease of use, Navigation, time for completion and errors removal.

8.4 Criteria of Test Cases

The system must comply with the success criteria mentioned in the specification. Some of the requirements are:

- ✍ Product Management Module should work efficiently.
- ✍ Transaction Management Module should work efficiently.
- ✍ Synchronization must work efficiently.
- ✍ The system should satisfy all user requirements.
- ✍ The Website should be easy to use.

8.5 Technical requirements for Testing

S. No.	Technical Requirements
1.	Operating system - Windows 7 and later
2.	CPU: 1 GHz
3.	RAM: 1 GB (estimated)
4.	Disk space: 20 GB (estimated)
5.	A good data/internet connection for customers

Table 8.2 Technical Requirements**8.6 Users Involved in Testing**

The developer will play the role of a tester.

8.7 Unit Testing

Examining all the pieces of the frame creates a single test being created. Unit The test was useful because it helped the designer test individual units of the source code. This done before further testing because it is not difficult to find bugs below a lower level than that obtained by a combination or a frame level. Thumbnail done on two levels; Acquisition tests are performed and after that the white box test is opened it's over. Two levels of unit testing were basically led, especially the black box (additionally the so-called testable test) and the white box (also called the accompanying test).

8.7.1 Black box testing (Functional Testing)

Acquisition Detection, also called Functional Testing, is the process of testing a product just revealed working with the framework plan; internal concealment frame formation. The internal theme or structure is unknown to the client who will use the framework

8.7.2 White Box Testing (Conceptual Testing) White Box Testing knew Code-Based Testing or Logical Testing is a product testing method that uncovered the interior execution or the inner structure of the framework or the code. The testing is to be performed by somebody who is a hard coder or the software engineer. It is something in front of the UI as it has nothing to do with the outside structure or the interface however is worried about the interior plan/structure or the execution. The testing is utilized to consequently test the inward operations of the proposed framework or the application.

8.8 Integration Testing Mix Testing is a trying strategy that consolidates the individual units or the modules of the framework and coordinated them together to be tried whether the modules work appropriately true to form after joining or produce any bugs. The reason for this testing to accomplish the accompanying: Every module ought to incorporate with one another with no information misfortune. Every module ought to be free of one another while running. To lead steady testing beginning from a lower level in the unit testing.

8.8.1 Technique used for Integration Testing

Base up combination testing is utilized which ensure that the module at the least level is grown first and different modules which go towards the 'primary' program are incorporated and tried each in turn.

8.9 System Testing

Framework Testing is a trying strategy that tests or checks whether the framework performs well and executed fine at a level when every one of the modules is coordinated into one framework in general. The testing additionally incorporates mapping of the prerequisites indicated with the yield of the framework or the application created.

8.10 Compatibility Testing

Matching test is a type of test that is not used for performance testing of Frame presented with equipment or product categories. Tests confirm that the finished application or framework works best when tested at various stages considering the operating framework and applications and tools. Motive after this test to see if the proposed framework works in a specific domain that contains equipment, editing, editing, framework operation and more.

8.11 Usability Testing and its Justification

The frame interface can be used and how easy the frame is understanding that it will be used. It looks at how much the frame is used or how easy it is to use with various categories of clients. In line with these lines, it is done from the perspective of the client. Ease of use Testing from now on proves that framework clients can use Effective, professional and effective framework. Performance testing is guided final clients evaluate the framework in five categories; readability, efficiency, customer memory, errors and fulfillment.

Learnability: Measure how easy it is for the end client to gain a framework performance or how complex the interface of the framework can be.

Efficiency: Means how fast the performance of the framework and assignments can accelerate he was killed. For example, the framework is very fast, directing the client from one movement to another some well.

Memorability: Measure how much the last client can remember the outline and efficiency when it comes to the framework after a critical time. For example, intuitive and easy-to-use interface makes it easy for a client to save a file outline items. Presentations were used to improve memory highlighting of the framework.

↳ **Errors:** Measures how the outline can work out in order to correct errors. Proposed outline is prone to errors; the toast message is given anywhere any error or release is normal.

- ↳ **User Satisfaction:** Measure how much the client likes the frame or is happy with outline. For example, the implementation of the aforementioned components contributes in terms of customer fulfillment, achieving readability, memory, productivity and dismissing mistakes achieves customer fulfillment.

↳ **8.11.1 User Acceptance Testing (UAT)**

Customer Acceptance Testing is a product testing process done in memory of final client opinion regarding the proposed framework or application there the engineer needs to check the framework to make sure it reaches the final client and is adequate by end client.

8.11.2 Justification for User Acceptance Testing

Reasons for User Acceptance Testing The purpose of the UAT is to monitor and direct the proposed requirements for action A building with the highlights of a completed business to ensure it is adequate for consumers. Customer Acceptance Testing Performed when the designer led another separate test, for for example, Unit Testing, Integration Testing, System Testing, Compliance Testing and more. The user consent test leads to the conclusion asking if the engineer has it achieve customer fulfillment. Not many people selected as final customers (analysts) of the framework and their applications were collected as a result of the User Acceptance Test.

8.12 Examination of documents This test is performed to check the accuracy and completeness of the text. All important conditions should be satisfied and a well-presented text should be published to meet all the requirements the standard required by the project manager. The completeness of the Scriptures was confirmed by details of the Final Year Project documents provided by the project manager. MS Word spelling tests also helped to correct spelling and grammar.

8.12 Documentation testing

This testing is done to check the accuracy and completeness of the document. All important criteria must be satisfied and a well-presented document should be published so that it meets the entire standard required by the project manager. The completeness of Documentation was verified with the Final Year Project documentation specification provided by the project manager. MS Word spell check also helped to correct spellings as well as write grammar.

8.13 Conclusion

Testing empowers designers to deliver that product that meets the needs of customers, to maintain unintended consequences, and improved long-term application support. In the beginning testing is very important if the designer is trying to reduce costs, unfortunately time again revitalizing, and trying to increase quality. System updates assisted engineer detect bugs or errors in the framework ensures efficient operation of the type drafts, tests and code in various cases. Depending on the motivation to test and product requirements, strategies are appropriate is used, for example, the implementation strategy to test the installation. Various test methods used at different levels to test the framework equally. Testing which is still being drafted, the adoption test was led and the draft test was officially conducted, white Box check was directed.

Unit testing is the first phase of testing that tests each framework module, both almost the same way. The design is designed to test the help of modules such as the distribution of code modules to check for bugs and errors where they occur and take and other tasks to get them out.

Integration testing, the corresponding promotion of unit testing, leads to the availability of individual modules it has been tried and should be combined continuously to test them to see if they cause any problems either errors after joining. Nothing but hard to do a merger test after unit testing targeted because it helps identify issues where units are integrated. Through an inspection system that requires the designer to inspect every unit and verify the performance of each item in advance assembling units, the developer knows that any errors found when joining units are possible identified by the contact area between the units. After the installation of the installation, the test of the framework led to the detection of problems found there regularly one of the units or modules connected together in a frame all in comparison tests were performed and recorded. The similarity test was directed at the framework to be considered even if the frame is well built in the resource base, editing, editing, database, framework or other program / equipment or some metal.

Performance testing and user acceptance testing are carried out late by end customers of outline to get their ideas on the framework, how to use the framework and if the framework

has a final purpose that provides complete customer fulfillment. The framework was seen as usable; in terms of readability, reproduction, recall, without errors the client is attractive. Customer service achieved to ensure that the framework is ready at the moment transferred.

CHAPTER 9: CRITICAL EVALUATION

9.1 Critical Evaluation

After the completion of the work on the designer side, a great opportunity to basically assess a job by mapping the task and its requirements with decisions given in a particular performance structure. Since the engineer did comprehensive research on the effectiveness of the framework, has led to a comprehensive study of equivalent. After research and investigation, the engineer designed the draft to set more targeted goals and then try to recover any errors when found in the framework of building a frame without distractions.

Now the opportunity has come to test the value-added framework framework assessment. Basic testing is not done module by module however the whole framework is evaluated to determine the quality, quality and value of outline.

9.2. Factors of Benefit (Usefulness of System to Target Audience)

A structured framework offers a lot of help to customers who are purposeful because the framework may deal with the problems specified in the section showing the release CHAPTER 2. The value of the framework was unquestionably and improbable local benefits.

9.3 Success Assessment

The success rate consists of a distribution of whether the yield has received a similarity was proposed. Contains comparisons of performance and results expectations. The current obligation meets all the requirements mentioned in PSF. Satisfy quality and performance.

9.3.1 Degree of Success

The speed of accomplishing any task is determined by the performance of the requirements archived of the proposed framework in the decision-making structure. The The framework must meet quality measures to improve business and efficient use to make it happen.

9.3.2 Critical Appraisal

The developer has attempted to incorporate all of the proposed functionality into the advanced application. The proposed requirements are successfully set up in the real world system. Now, after the end of the first and test phase, the safe application is ready posts. There were many obstacles during the process. The the developer included part of the application he has made in the implementation phase and is now working on the same topic with his research work. Without this, performance requirements, non-functional requirements such as quality, usability, and usability of the user, etc. reached by the engineer. Apart from this, the project is tested in a variety of real-time testing situations to achieve real results, tests, and responses. The customized system works well as proposed and ready to access.

CHAPTER 10: CONCLUSION

10.1 Success Criteria

The proposed framework called the "Post Pandemic Crowd Safety System" has been shown to be primarily useful to the end customer as it meets all the details of client requirement just as it satisfies every model that judges quality against implementation of the framework. The implementation of the proposed framework is achieved as it has met the process of success and created the right one. The framework has resolved the issues assessed in section 2 relating to issues suggested and arrangements provided. Proposed framework after testing which promises to provide security for its users by providing compliant details about the current epidemic for him.

10.2 Limitations and Errors in the Developed System:

- ✚ The customer will need the internet connection for getting information simultaneously about the current situation.
- ✚ The entire project is based on crowdsourcing of data.
- ✚ The user should have GPS enabled in his/her device.

10.3 Learning Experience Gathered

- ✚ **Learnt a lot from research work:** In the first phase of the work, the engineer he was very confused about what was to be remembered for the framework. In this way, the designer completed the optional study by considering the experimental activities of Various creators and watch to pay attention to all the mistakes they have made. It helped to find unusual information about the story and comparative distribution structures related to their prominence and decline. Finally, it is led by a designer important research to integrate ongoing customer needs.
- ✚ **Learned to manage stress and time:** While working at various times of SDLC, the engineer has taken on different functions such as the following: analyzer, analyst, database and portable app designer and analyzer. This gave the designer the power to manage pressure and time successfully. The engineer had a chance to finish the job within the time

frame we overcome all the burdens and needs of the opportunities that come during development.

- ✍ **Gained technical expertise:** The usage of the framework required not exclusively to apply learning results accomplished during scholastics yet additionally a reasonable comprehension of different versatile innovations. It required a lot of webs searching to grasp the user interface design as well as the structure of the system. This lead to a huge learning curve for the developer.
- ✍ **Had an experience of project management:** The undertaking gave a great encounter and sentiment of filling in as though in some corporate world. The designer likewise assumed an imperative job of an undertaking supervisor by getting the finished on schedule, conveying a quality framework, satisfying client necessities, utilizing reality discovering strategies to gather prerequisites. Separation and vanquish approach was utilized to partition the whole venture into little assignments and finishing them as indicated by the calendar arranged in the Gantt graph.
- ✍ **How to resolve errors and learn from them:** When using an outline, The designer has encountered many errors and special cases reviewed in the background using different techniques. This provided a glimpse of how to solve a problem and fix errors using the top-down method. Similarly, an engineer saw a way to find out errors and correct them which is appropriate which is expected to pass error free outline.
- ✍ **Gained lot of confidence:** Before making the outline, the engineer was in a certain position how to use complex framework modules. Be as realistic as possible, continuously things became easier after doing the research work in each development period itself removing the concept-building approach and spreading the whole framework. Still it required a steady and real effort, yet the designer proved to be the idea that ‘where there is a will, there is a way’. This study inspired the composer that embraces the outstanding needs by unquestionably accepting that in each issue there is the answer.
- ✍ Last yet not the least, this last year venture gave an extraordinary learning and charming experience of chipping away at ongoing ventures and made a profound enthusiasm for the portable application advancement field which will help in receiving rewards for having a splendid profession in future.

10.4 Conclusion

The documentation shows the report of the proposed framework, how the framework informs various stages as we progress. The engineer began with a basic investigation to determine whether there is a need for a security system that can provide its users with information about it Covid-19's worst areas. The basic reading phase is followed by a literary study of the type research to integrate data into the area of mitigation and focus flow to build functional framework.

The designer led important and helpful research. Assuming it is a good combination data about the proposed framework, the designer needed to hear a different development. Survey and Discussions and data collection methods used to identify the need of willing people, workers within the organization. Research completed by staff / staff while interviews are led by management and in-depth receptions people. Poll testing and meetings and institutional circles helped the developer to master it full details and understanding of the organization's security efforts as clients were taken from various parts of the machine. In addition, special details were taken by taking a meeting with product engineers and planners. Their suggestions check for internal and external information on developments and details. In accordance with these lines, the test portion of the proposed work has been completed.

The engineer then proceeded with the part of the draft frame where the draft adjusted configurations that include the intellectual and physical systems of the framework that advances on UML lists, framework body plans, relationship graphs and more forward. At that point it comes to the point of use which includes a method of dealing with compliance with designer to create different modules for a framework that starts with a lower module need and going towards the highest needs. Otherwise, the client manual as A special manual is provided that assists the client with the continuation of the framework and how using the framework and direction of the establishment separately.



Finally, the final task is assessed to determine the decision, how the framework can benefit its end clients and how the work benefits the client's perspective. Finally, the last task is assessed to arrive at a resolution, how the framework can profit its end-clients and how much the undertaking is fruitful from the client's perspective.

CHAPTER 11: REFERENCES

11.1 Research Papers

- ✎ R. Lovreglio, E. Ronchi, and M. J. Kinsey. "An Online Survey of Pedestrian Evacuation Model Usage and Users," Fire Technology, November 2019, DOI: 10.1007/s10694-019-00923-8.
- ✎ I. T. Yu et al. "Evidence of airborne transmission of the severe acute respiratory syndrome virus," New England Journal of Medicine, Vol. 350, No. 17, pp. 1731–1739, 2004. [3] D. Lewis. "Is the coronavirus airborne? Experts can't agree," Nature, Vol. 580, No. 7802, pp. 175–175, April 2020, DOI: 10.1038/d41586-020-00974-w.

11.2 Books

- ✎ Marco L. Napoli (Author) 2020 Beginning Flutter: A Hands-On Guide to App Development 1st Edition