Containment Zone Alerting Application

Bachelor of Engineering

In

Computer Science and Engineering

Submitted by

TEAMI ID: PNT2022TMID26804

SUKHIT KUMAR	310519104129
RISHIKESH KUMAR	310519104099
ROUSHAN RAJ	310519104103
VIKASH KUMAR RAY	310519104141

DHANALAKSHMI SRINIVASAN COLLEGE OF ENGINEERING AND TECHOLOGY

603104

ANNA UNIVERSITY: CHENNAI

TABLE OF CONTENT

Project Report

Team Id	**PNT2022TMID26804**
Project Name	**CONTAINMENT ZONE ALERTING APPLICATION**

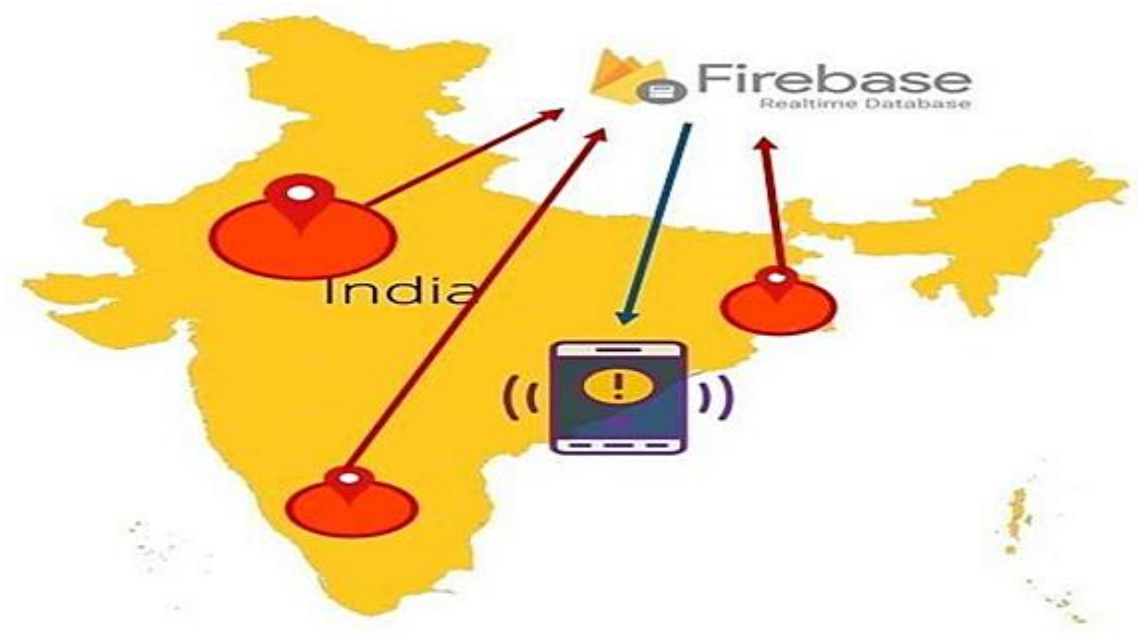
1. INTRODUCTION

1.1 Project Overview

Currently there are several research works undergoing in the country to prevent Covid-19 cases from rising. Previously our country was importing medical kits like PPE (Personal Protection Kits), mask from outside, but now it has been successful in developing these kits. Along with taking initiatives to fight this disease, our country has also taken steps to make people aware of the disease. The news and media have a great part in creating this awareness by informing the public about the preventive measures that can keep them away from infection. Awareness among the people to carry out all the preventive measures can immensely help to reduce spread of the virus. The country has created containment zones throughout the cities wherever Covid-19 cases have been reported to prevent further spread of the virus. These containment zones have been kept isolated from the outside public to ensure no contamination occurs outside. After more than 2 months of the lockdown, the government has relaxed some of the lockdown rules and has permitted reopening of government offices, bus and other road transportation facilities and shopping markets. People can move inside the city for work and other purposes. But the containment zones are still being kept isolated, and new containment zones are being formed wherever Covid-19 cases have been reported. These zones are highly contagious as droplets with virus coughed out from an unscreened asymptomatic patient can travel up to 8 m (Bahl et al. 2020). Though these containment zones are guarded by policemen, still there remains a chance that people might unknowingly step into them. In this situation where people can move in the city, these containment zones pose a risk of infection to these city dwellers. Therefore, informing people about the location of the containment zones can help them bypass and avoid these zones and thereby reduce the chance of community transmission. In this paper, we focus on developing a mobile based application to provide information regarding the Covid-19 containment zones in West Bengal. The application further tracks the user's location and provides notification alert if the user has entered a containment zone. The application also provides daily Covid-19 case statistics to the users to keep them updated. The application is developed on Android SDK and uses Firebase Cloud Fire store to store the location data. Android's geofencing client is used to create geofences around the containment zones and notification manager is used to provide notifications. The application also uses RESTful web services to show the Covid-19 cases in West Bengal. We have tested our application with different users in different locations across West Bengal and it works efficiently and is able to attain our target.

1.2 Purpose:

The Android application shows the location of the containment zones to the users. It also notifies the user when he or she trespasses the boundary of a containment zone or stays in the containment zones



2. LITERATURE SURVEY:

2.1 Existing problem:

The recent outbreak of COVID-19 has taken the world by surprise, forcing lockdowns and straining public health care systems. COVID-19 is known to be a highly infectious virus, and infected individuals do not initially exhibit symptoms, while some remain asymptomatic. Thus, a non-negligible fraction of the population can, at any given time, be a hidden source of transmissions. In response, many governments have shown great interest in smartphone contact tracing apps that help automate the difficult task of tracing all recent contacts of newly identified infected individuals. However, tracing apps have generated much discussion around their key attributes, including system architecture, data management, privacy, security, proximity estimation, and attack vulnerability. In this article, we provide the first comprehensive review of these much-discussed tracing app attributes. We also present an overview of many proposed tracing app examples, some of which have been deployed countrywide, and discuss the concerns users have reported regarding their usage. We close by outlining potential research directions for next-generation app design, which would facilitate improved tracing and security performance, as well as wide adoption by the population at large.

2.2 References

[1] . P. H. O’Neill, T. Ryan-Mosley, and B. Johnson. (2020). A Flood of Coronavirus Apps are Tracking Us. Now it’s Time to Keep Track of Them. [Online]. Available:

https://www.technologyreview.com/2020/05/ 07/1000961/launching-mittr-cov%id-tracing-tracker/
[2] . S. Vaudenay, ‘‘Centralized or decentralized? The contact tracing dilemma,’’ IACR Cryptol. ePrint Arch., vol. 2020, p. 531, May 2020. [Online]. Available: https://eprint.iacr.org/2020/531
[3] . C. Criddle and L. Kelion. (2020). Coronavirus Contact-Tracing: World Split Between Two Types of App. [Online]. Available: https://www.bbc. com/news/technology-52355028
[4] . J. Duball. (2020). Centralized vs. Decentralized: EU’s Contact Tracing Privacy Conundrum. [Online]. Available: https://iapp.org/news/a/ centralized-vs-decentralized-euscontact-tracin%g-privacy-conundrum
[5] . R. Jennings. (2020).What are the Data Privacy Considerations of Contact Tracing Apps. [Online]. Available: https://ukhumanrightsblog. com/2020/05/01/what-are-the-dataprivacy-cons%iderations-of-contacttracing-apps/
[6] . D. Palmer. (2020). Security Experts Warn: Don’t Let Contact-Tracing App Lead to Surveillance. [Online]. Available: https://www.zdnet.com/ article/security-experts-warndont-let-contact-tr%acing-app-lead-tosurveillance

2.3 Problem Statement Definition

The COVID-19 pandemic continues to affect the way of life of everyone. The contact tracing apps are likely to play a vital role in aiding health authorities quickly identify individuals that may have been exposed to the virus. The imminent interest and adoption of tracing app technology will improve the tracing capability of health authorities; however, as this article highlighted, it is not a silver bullet. These apps still face many concerns from users, data protection agencies, and researchers. The main concerns are related to the user data management, potentially non-trivial false positive and negative instances, and the security and privacy issues of these apps. Guided by these concerns, this article presented an overview of the three common tracing app architectures: centralised, decentralised, and hybrid; and an overview of popular apps within these categories. Additionally, the paper focused on the privacy and security aspects, mapping attacks that could be possibly performed in each of the three architectures. This article also elucidates some other users’ concerns regarding battery drain, compatibility, consent withdrawal, and transparency. Finally, we discussed some of the near and long term future research directions.

PAPER 1:

TITLE: Tracking the Covid zones through geo-fencing technique

AUTHOR NAME: Anto Arockia Rosaline R ,Lalitha R ,Hariharan G ,Lokesh

PUBLICATION YEAR: 2017 DESCRIPTION:

Following the tracking of a suspicious person, the geo-fenced layer is mapped out in the vicinity, and the virtual perimeter is then employed for the subsequent trapping procedure. As soon as the Covid monitoring team updates this geo-fenced layer, the public can view it. The idea of creating a virtual perimeter region is known as geo-fencing. Effective containment zone monitoring is made possible by this virtual perimeter monitoring technology. By utilising an automated system based on wireless infrastructure, it lowers operational costs. Additionally, it promptly alerts the law enforcement to find the offenders. As a result, it facilitates the inspection of containment areas and the monitoring of those who disobey governmental regulations. Users can receive updates from the Covid team on the alert zone. The Covid team has a number of modules for suspect tracking, hotspot fencing, etc. The Covid team must seek a service from the service network provider in the case of suspect tracking, and following authorization, they will offer the coordinates. According to our telecommunication legislation, it is illegal to share data; nonetheless, exchanging personal information without the individual's knowledge via any means is occasionally allowed with governmental approval for investigative purposes.

PAPER 2:

AUTHOR NAME: Geofencing 2.0: Taking Location-based Notifications to the Next Level

PUBLICATION

YEAR: 2016

DESCRIPTION

Sandro Rodriguez Garzon Bersant Deva The basic Android application that served as the prototype Geofencing client was used. This client is primarily responsible for carrying out the geofencing server's ongoing location update strategy. This must be accomplished with little energy consumption because the Geofencing client is located on a mobile device. We made the decision to employ the low energy Geofencing features of the Android operating system to keep an eye on the safety zone. As a result, a safety zone is considered as a single circular geofence with a required exit on the mobile device. However, they discovered that there was occasionally a significant lag time between leaving the safety zone and receiving a notification from the system about the leave. In order to address this issue, a specific amount of the safety zone's radius is decreased. While the safety zone and how it is implemented have a significant impact on overall energy consumption, it is also important to make the right choice when it comes to a placement mechanism. In order to reduce power consumption without compromising the necessary position precision, they used a device-based smart combination of various positioning mechanisms introduced by. By temporarily deactivating placement when a device is not in motion, the Geofencing client also makes use of cutting-edge mobile sensing capabilities integrated into the Android operating system's activity recognition unit. Mobile users who live close to a geo-border fence's will find this to be of particular utility. If the Geofencing server notifies the Geofencing client about a geonotice, the notification will appear right away.

PAPER 3

TITLE: Development of An Android Application for Viewing Covid19 Containment Zones Alerting.

AUTHOR NAME: India Ranajoy Mallik, Amlan Protim Hazarika, Sudarshana Ghosh Dastidar, Dilip Sing & Rajib Bandyopadhyay

PUBLICATION YEAR: 2019

DESCRIPTION:

The World Health Organization has declared the outbreak of the novel coronavirus, Covid-19 as pandemic across the world. With its alarming surge of affected cases throughout the world, lockdown, and awareness (social distancing, use of masks etc.) among people are found to be the only means for restricting the community transmission. In a densely populated country like India, it is very difficult to prevent the community transmission even during lockdown without social awareness and precautionary measures taken by the people. Recently, several containment zones had been identified throughout the country and divided into red, orange and green zones, respectively. The red zones indicate the infection hotspots, orange zones denote some infection and green zones indicate an area with no infection. This paper mainly focuses on development of an Android application which can inform people of the Covid-19 containment zones and prevent trespassing into these zones. This Android application updates the locations of the areas in a Google map which are identified to be the containment zones. The application also notifies the users if they have entered a containment zone and uploads the user's IMEI number to the online database. To achieve all these functionalities, many tools, and APIs from Google like Firebase and Geofencing API are used in this application. Therefore, this application can be used as a tool for creating further social awareness about the arising need of precautionary measures to be taken by the people of India.

PAPER 4:

TITLE: Aarogya Setu

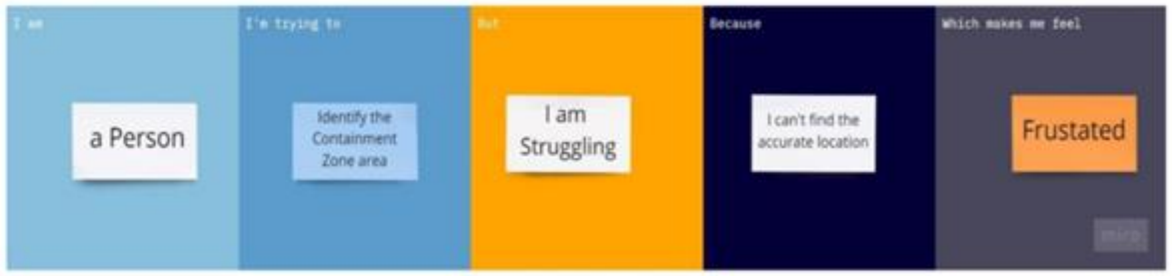
AUTHOR NAME: National Informatics Centre, Ministry of Electronics & Information Technology, Government of India

PUBLICATION YEAR: 2014

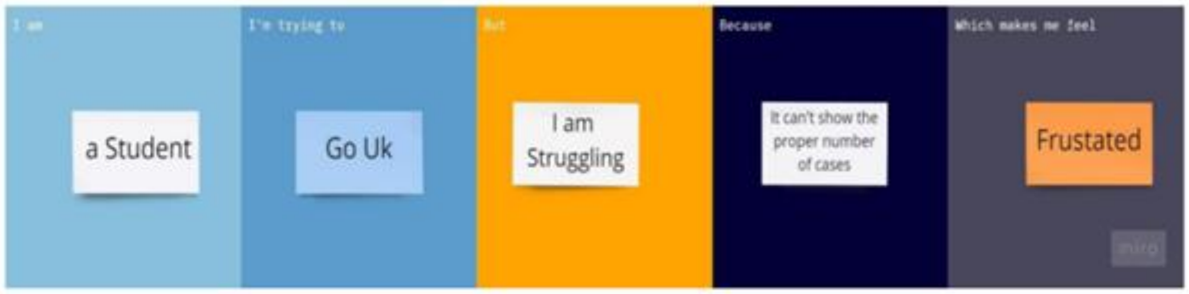
DESCRIPTION:

The most popular containment zone alert application among the options currently in use in India is called Aarogya Setu. The Indian government created a mobile application to link the public with crucial health services. Its primary features include geo-location-based COVID19 data, user risk status, automatic contact tracing using Bluetooth, and much more. The movement of an infected individual is tracked using Bluetooth and GPS technology, and the system notifies the public of the locations the infected person has visited while designating those locations as vulnerable ones. It employs cellular triangulation to determine a person's location in the absence of GPS technology. While Aarogya Setu can track down contacts and notify those who have come into touch with someone who has COVID-19, it also actively keeps track of quarantine or containment zones and alerts users who enter them. The Terms of Use and Privacy Policy must be accepted at the time of registration when installing the application on any Android or iOS mobile device, and ongoing use of the application denotes continued acceptance. Name, age, sex, occupation, phone number, overseas travel within the previous 28–45 days, and whether the user is a smoker are all pieces of information that the app gathers. This data is kept on a server that is under the jurisdiction of the Indian government. It is hashed and sent to the user's mobile application along with a special digital ID (DID). The user is recognised using the DID. In order for the user's mobile phone to exchange information with another device that has the app when it gets within range, the Bluetooth and GPS services must be turned on. Their individual IDs, along with the time and GPS location, are kept on the two phones when two users come into close proximity. The format in which this data is kept is encrypted. Only after a person tests positive is it posted to the government-controlled servers of the app.

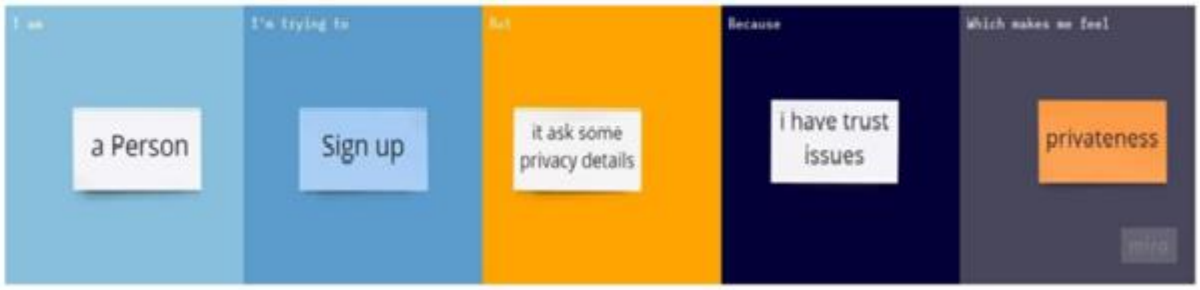
PROBLEM STATEMENT 1:



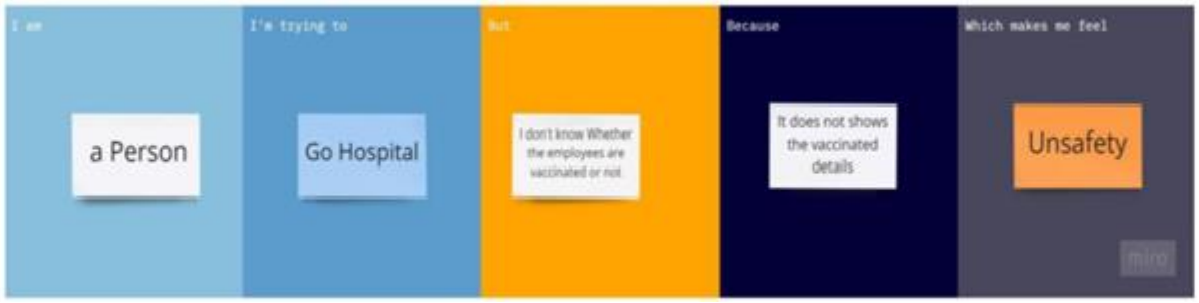
PROBLEM STATEMENT 2:



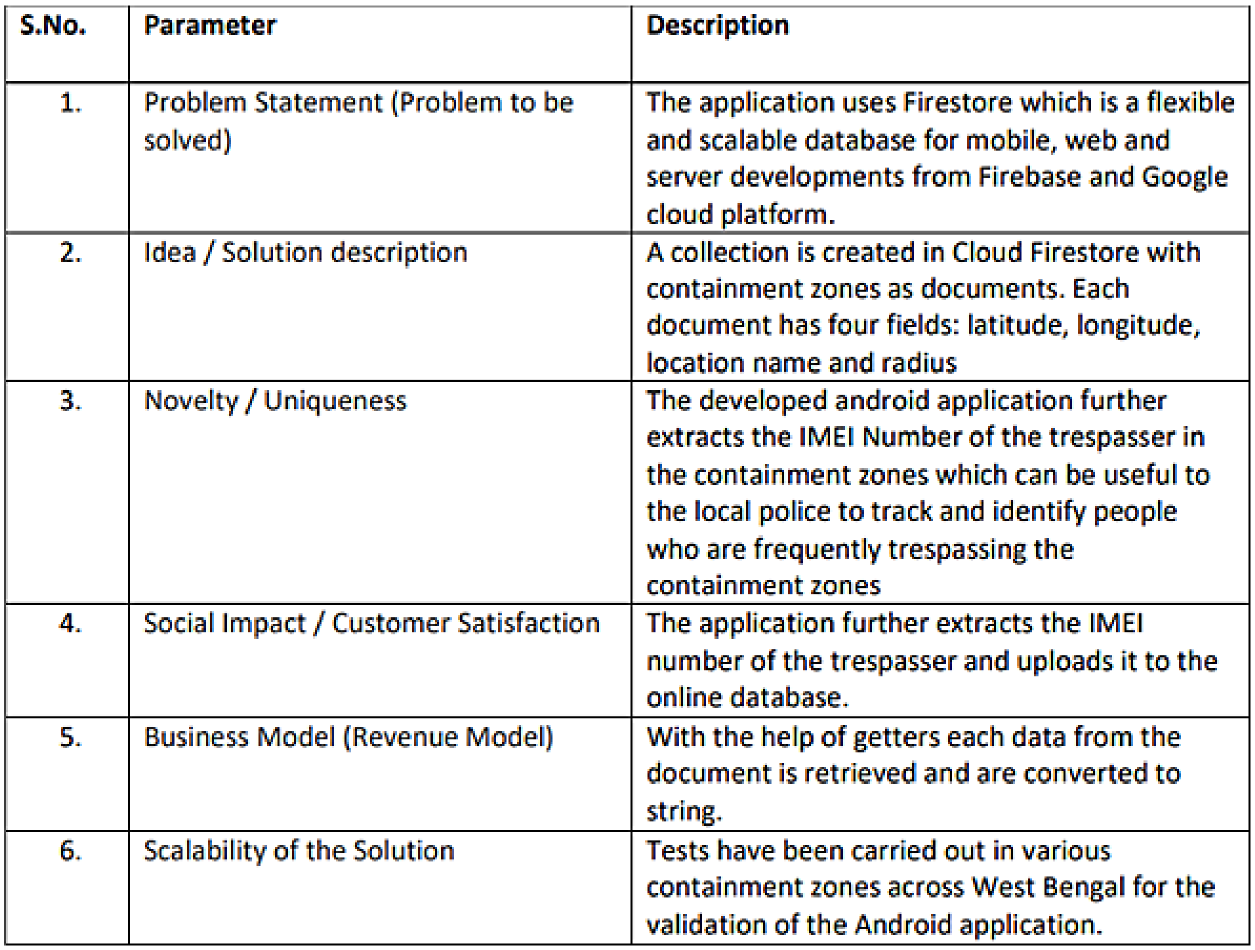
PROBLEM STATEMENT 3:



PROBLEM STATEMENT 4:


I am
a Person
I'm trying to
Go Hospital
But
I don't know Whether the employees are vaccinated or not
Because
It does not shows the vaccinated details
Which makes me feel
Unsafety
miro

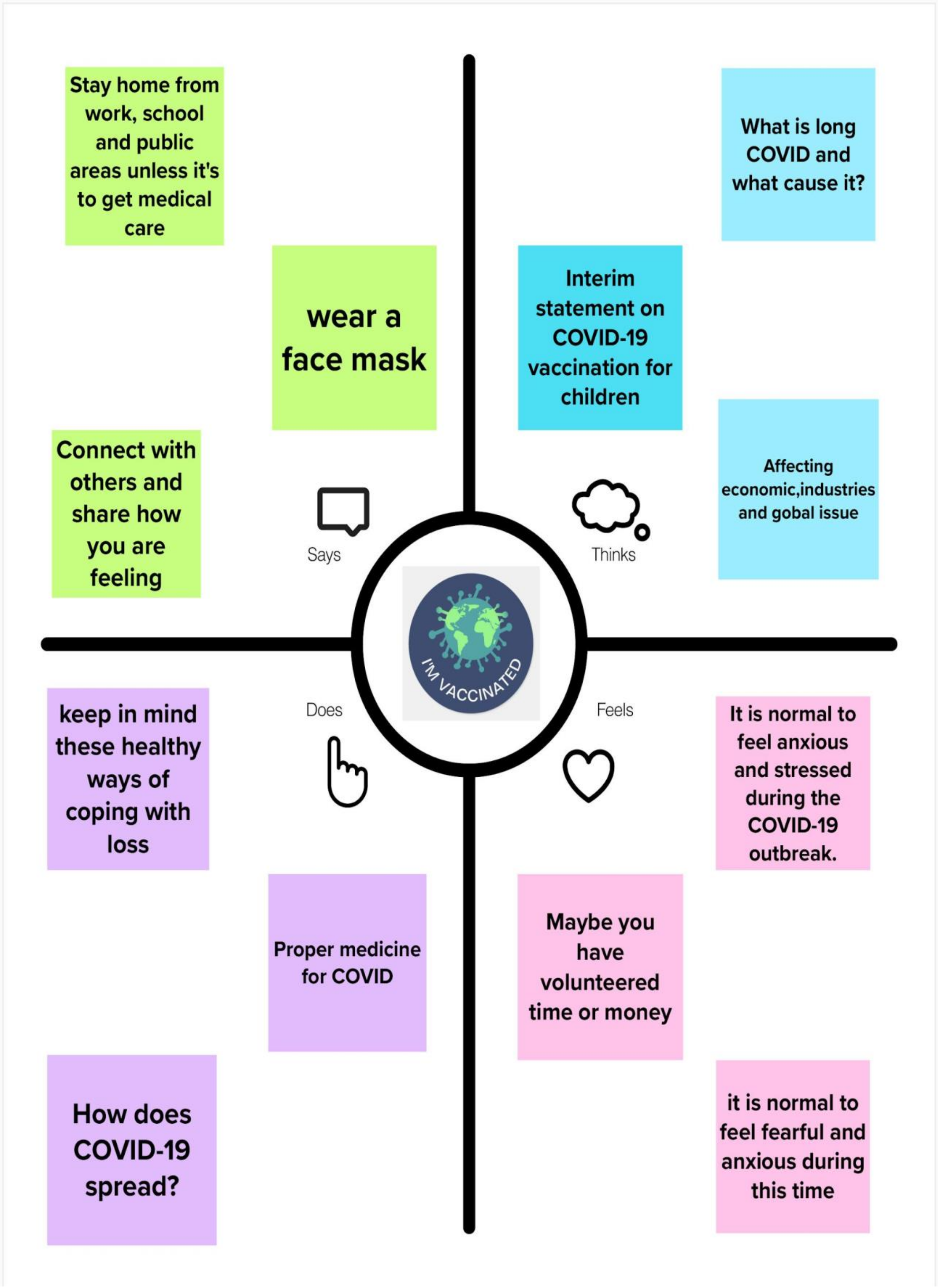
3.IDEATION & PROPOSED SOLUTION



S.No.	Parameter	Description
1.	Problem Statement (Problem to be solved)	The application uses Firestore which is a flexible and scalable database for mobile, web and server developments from Firebase and Google cloud platform.
2.	Idea / Solution description	A collection is created in Cloud Firestore with containment zones as documents. Each document has four fields: latitude, longitude, location name and radius
3.	Novelty / Uniqueness	The developed android application further extracts the IMEI Number of the trespasser in the containment zones which can be useful to the local police to track and identify people who are frequently trespassing the containment zones
4.	Social Impact / Customer Satisfaction	The application further extracts the IMEI number of the trespasser and uploads it to the online database.
5.	Business Model (Revenue Model)	With the help of getters each data from the document is retrieved and are converted to string.
6.	Scalability of the Solution	Tests have been carried out in various containment zones across West Bengal for the validation of the Android application.

3.1 Empathy Map Canvas:

An empathy map is a simple, easy-to-digest visual that captures knowledge about a user’s behaviours and attitudes. It is a useful tool to helps teams better understand their users. Creating an effective solution requires understanding the true problem and the person who is experiencing it. The exercise of creating the map helps participants consider things from the user’s perspective along with his or her goals and challenges.



Says

Stay home from work, school and public areas unless it's to get medical care

wear a face mask

Connect with others and share how you are feeling

Thinks

What is long COVID and what cause it?

Interim statement on COVID-19 vaccination for children

Affecting economic,industries and gobal issue

Does

keep in mind these healthy ways of coping with loss

Proper medicine for COVID

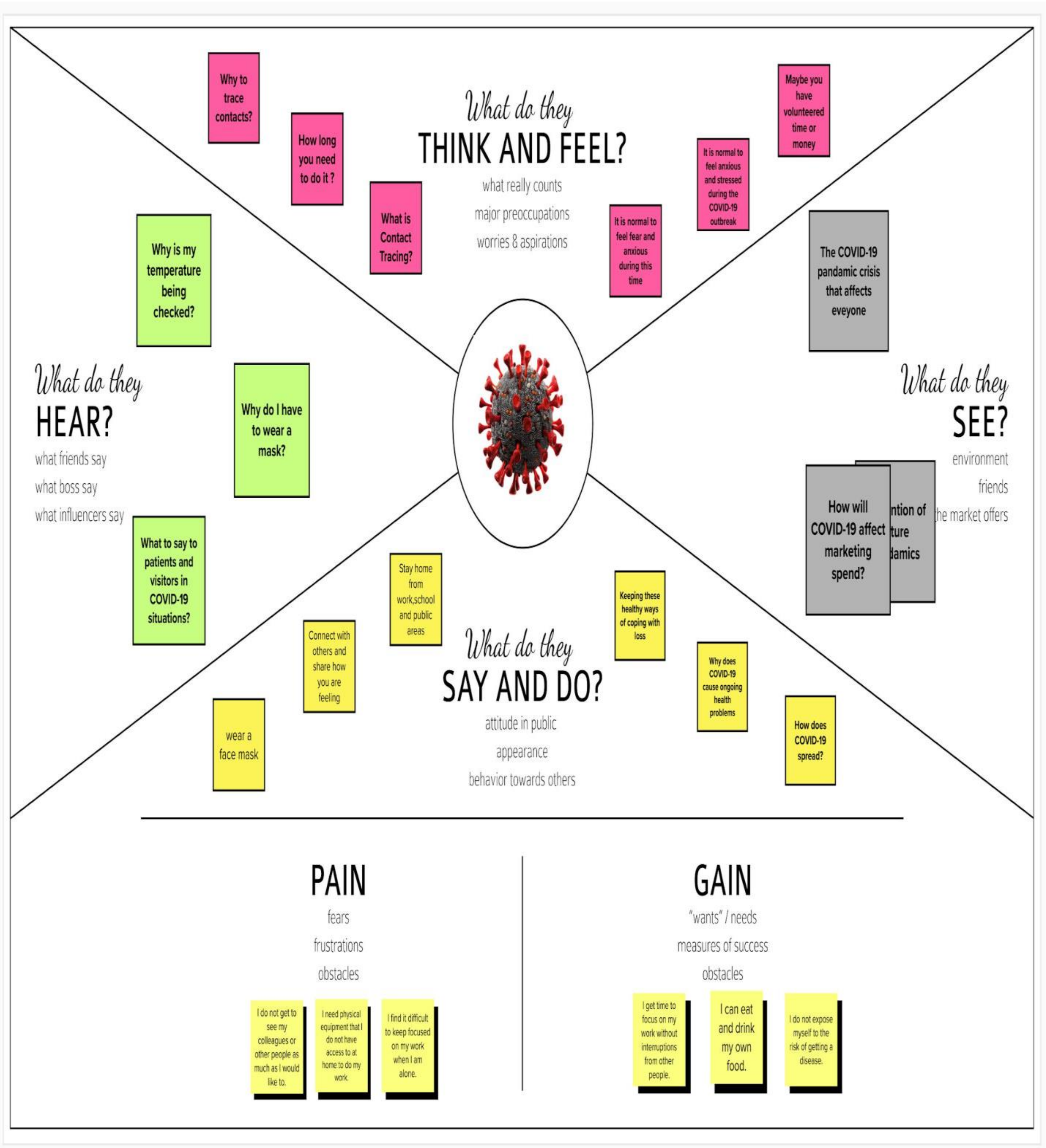
How does COVID-19 spread?

Feels

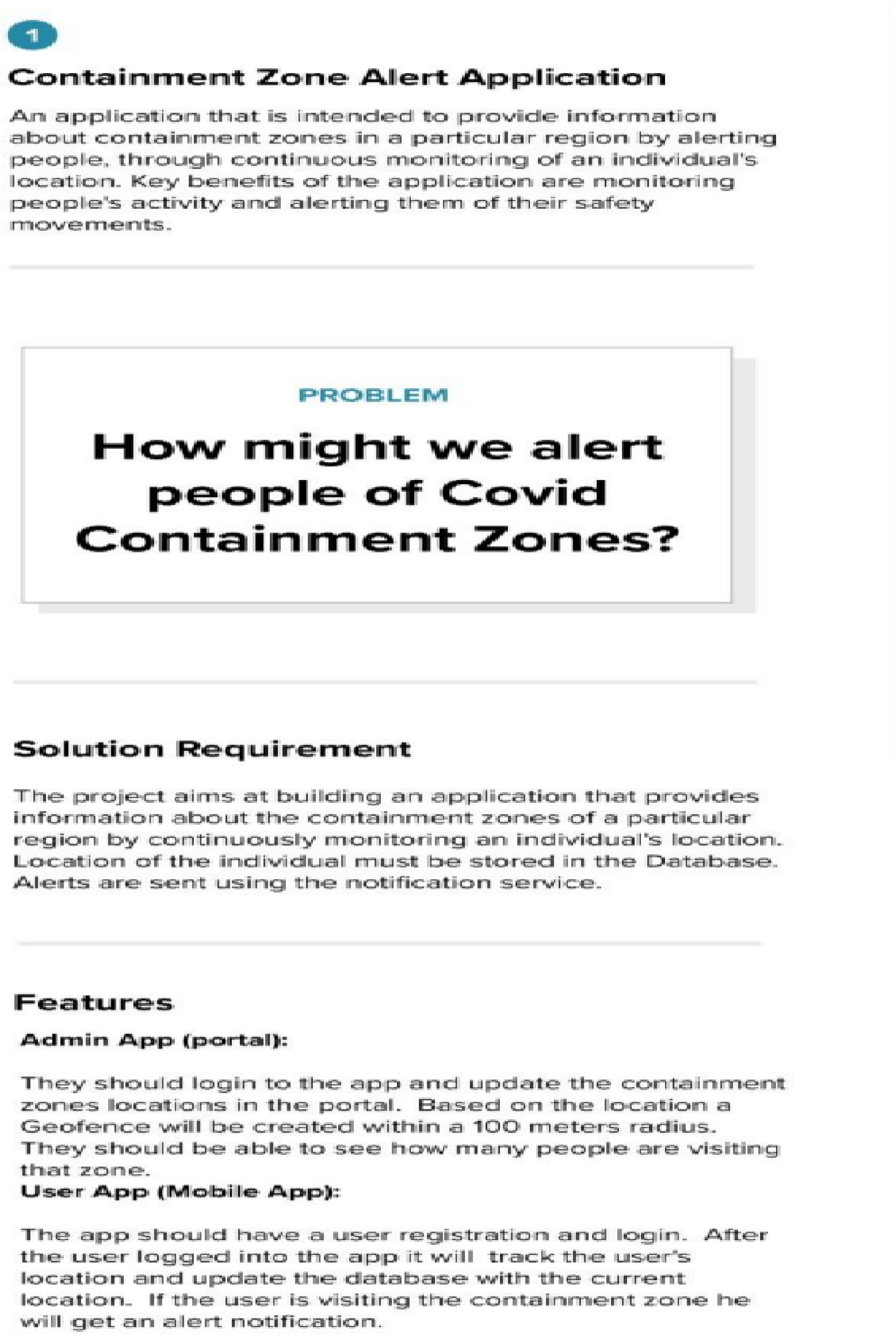
It is normal to feel anxious and stressed during the COVID-19 outbreak.

Maybe you have volunteered time or money

it is normal to feel fearful and anxious during this time


What do they
THINK AND FEEL?
what really counts
major preoccupations
worries & aspirations
Why to trace contacts?
How long you need to do it ?
What is Contact Tracing?
It is normal to feel fear and anxious during this time
It is normal to feel anxious and stressed during the COVID-19 outbreak
Maybe you have volunteered time or money
What do they
HEAR?
what friends say
what boss say
what influencers say
Why is my temperature being checked?
Why do I have to wear a mask?
What to say to patients and visitors in COVID-19 situations?
What do they
SEE?
environment
friends
the market offers
The COVID-19 pandamic crisis that affects eveyone
How will COVID-19 affect marketing spend?
ntion of ture damics
What do they
SAY AND DO?
attitude in public
appearance
behavior towards others
wear a face mask
Connect with others and share how you are feeling
Stay home from work,school and public areas
Keeping these healthy ways of coping with loss
Why does COVID-19 cause ongoing health problems
How does COVID-19 spread?
PAIN
fears
frustrations
obstacles
I do not get to see my colleagues or other people as much as I would like to.
I need physical equipment that I do not have access to at home to do my work.
I find it difficult to keep focused on my work when I am alone.
GAIN
"wants" / needs
measures of success
obstacles
I get time to focus on my work without interruptions from other people.
I can eat and drink my own food.
I do not expose myself to the risk of getting a disease.

3.2 Ideation & Brainstorming



1

Containment Zone Alert Application

An application that is intended to provide information about containment zones in a particular region by alerting people, through continuous monitoring of an individual's location. Key benefits of the application are monitoring people's activity and alerting them of their safety movements.

PROBLEM

How might we alert people of Covid Containment Zones?

Solution Requirement

The project aims at building an application that provides information about the containment zones of a particular region by continuously monitoring an individual's location. Location of the individual must be stored in the Database. Alerts are sent using the notification service.

Features

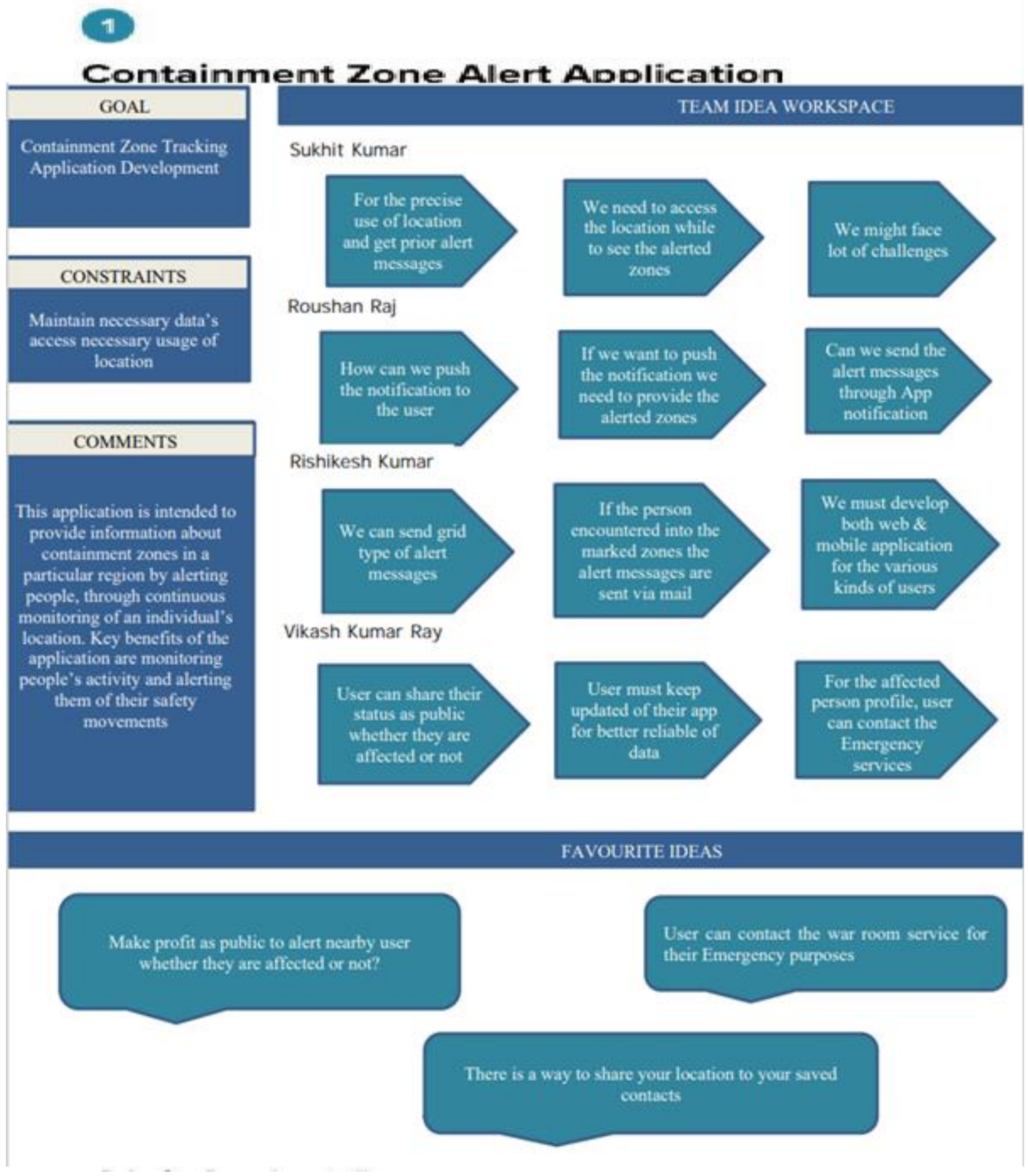
Admin App (portal):

They should login to the app and update the containment zones locations in the portal. Based on the location a Geofence will be created within a 100 meters radius. They should be able to see how many people are visiting that zone.

User App (Mobile App):

The app should have a user registration and login. After the user logged into the app it will track the user's location and update the database with the current location. If the user is visiting the containment zone he will get an alert notification.

Brainstorming, Idea Listing and Grouping



1

Containment Zone Alert Application

GOAL

Containment Zone Tracking Application Development

CONSTRAINTS

Maintain necessary data's access necessary usage of location

COMMENTS

This application is intended to provide information about containment zones in a particular region by alerting people, through continuous monitoring of an individual's location. Key benefits of the application are monitoring people's activity and alerting them of their safety movements

TEAM IDEA WORKSPACE

Sukhit Kumar

- For the precise use of location and get prior alert messages
- We need to access the location while to see the alerted zones
- We might face lot of challenges

Roushan Raj

- How can we push the notification to the user
- If we want to push the notification we need to provide the alerted zones
- Can we send the alert messages through App notification

Rishikesh Kumar

- We can send grid type of alert messages
- If the person encountered into the marked zones the alert messages are sent via mail
- We must develop both web & mobile application for the various kinds of users

Vikash Kumar Ray

- User can share their status as public whether they are affected or not
- User must keep updated of their app for better reliable of data
- For the affected person profile, user can contact the Emergency services

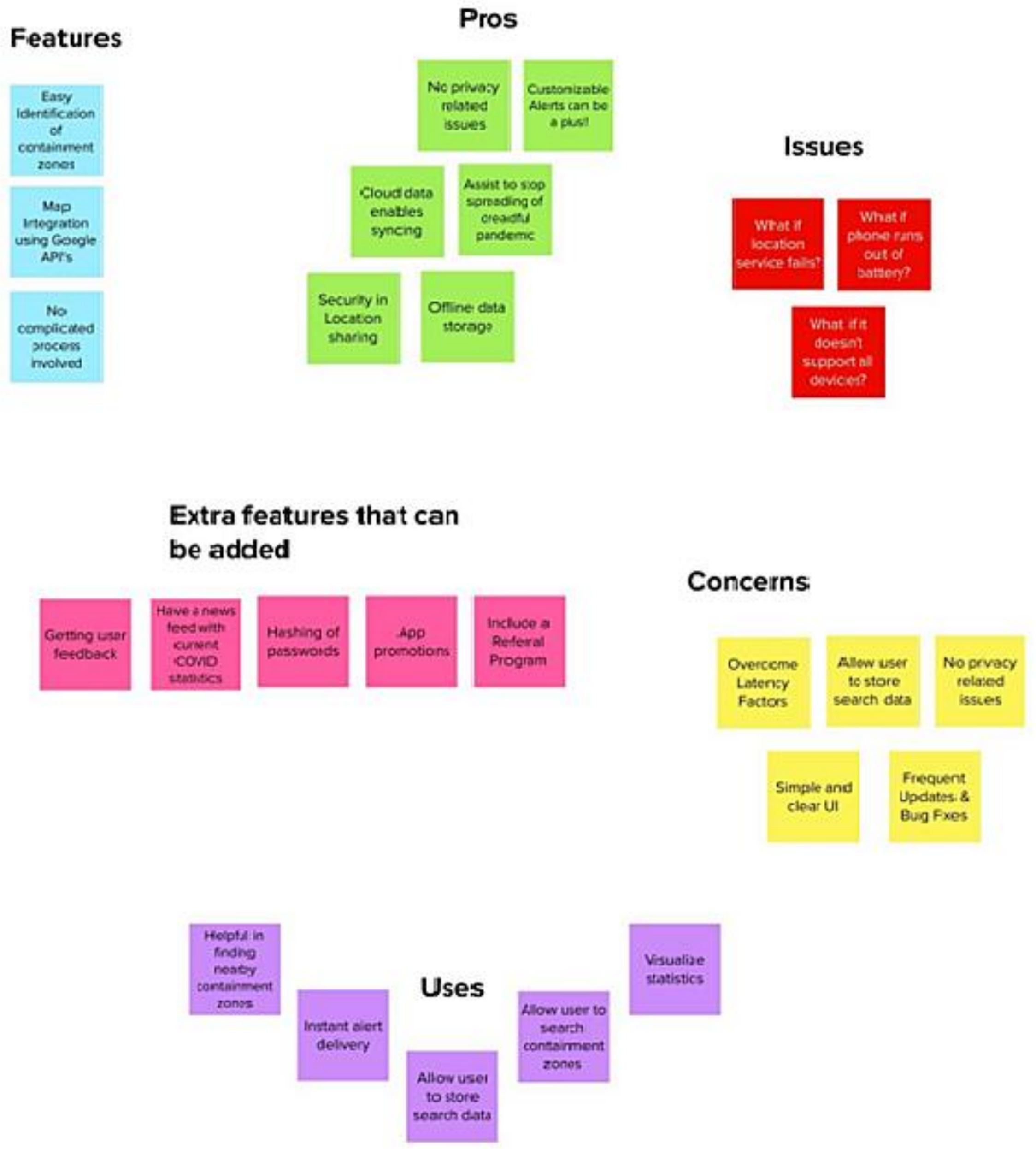
FAVOURITE IDEAS

- Make profit as public to alert nearby user whether they are affected or not?
- User can contact the war room service for their Emergency purposes
- There is a way to share your location to your saved contacts

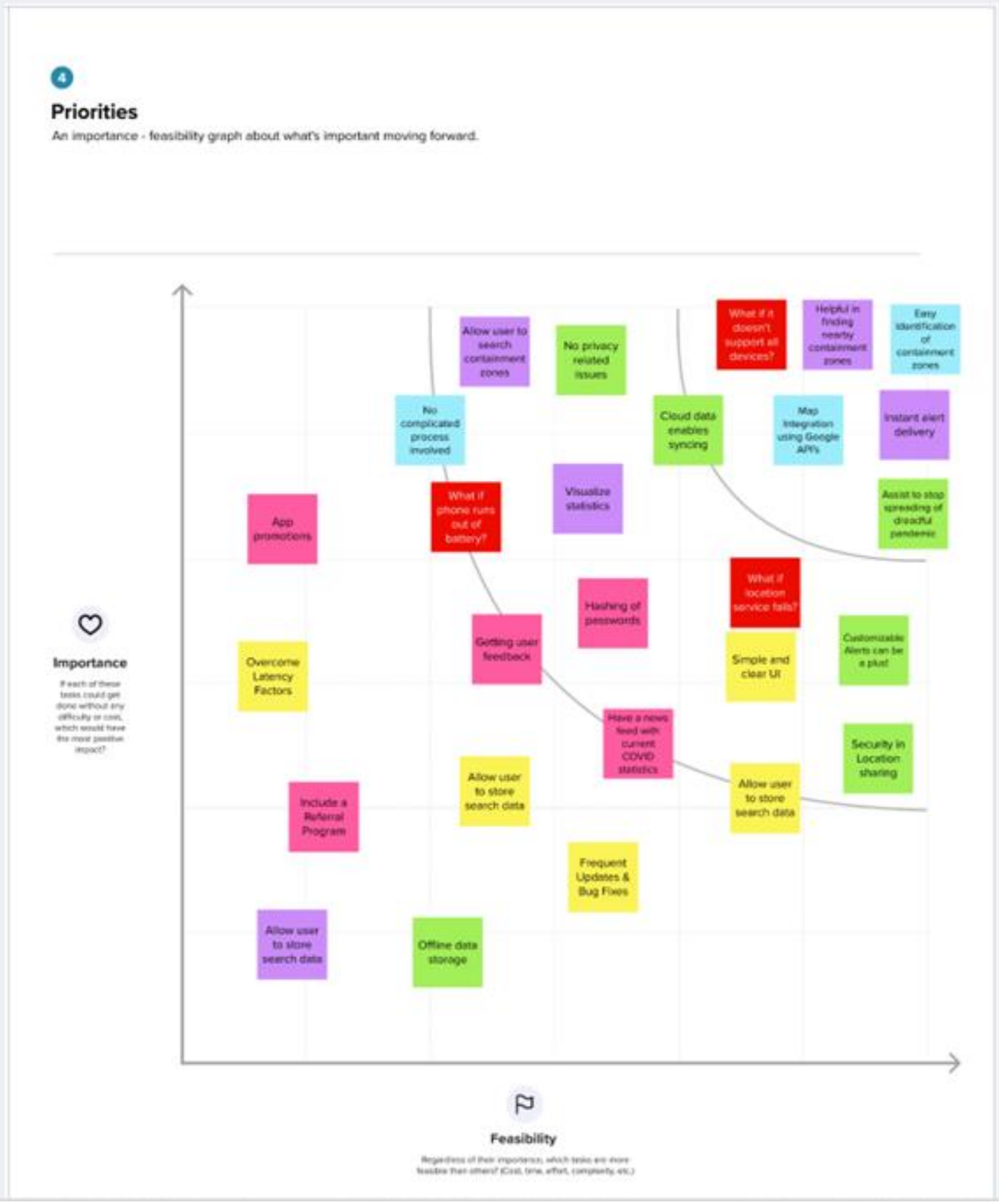
3

Grouping of ideas

Each group is a collection of similar ideas from the brainstorm that has a title which describes what the ideas have in common.


Features
Easy Identification of containment zones
Map Integration using Google API's
No complicated process involved
Pros
No privacy related issues
Customizable Alerts can be a plus!!
Cloud data enables syncing
Assist to stop spreading of dreadful pandemic
Security in Location sharing
Offline data storage
Issues
What if location service fails?
What if phone runs out of battery?
What if it doesn't support all devices?
Extra features that can be added
Getting user feedback
Have a news feed with current COVID statistics
Hashing of passwords
App promotions
Include a Referral Program
Concerns
Overcome Latency Factors
Allow user to store search data
No privacy related issues
Simple and clear UI
Frequent Updates & Bug Fixes
Uses
Helpful in finding nearby containment zones
Instant alert delivery
Allow user to store search data
Allow user to search containment zones
Visualize statistics

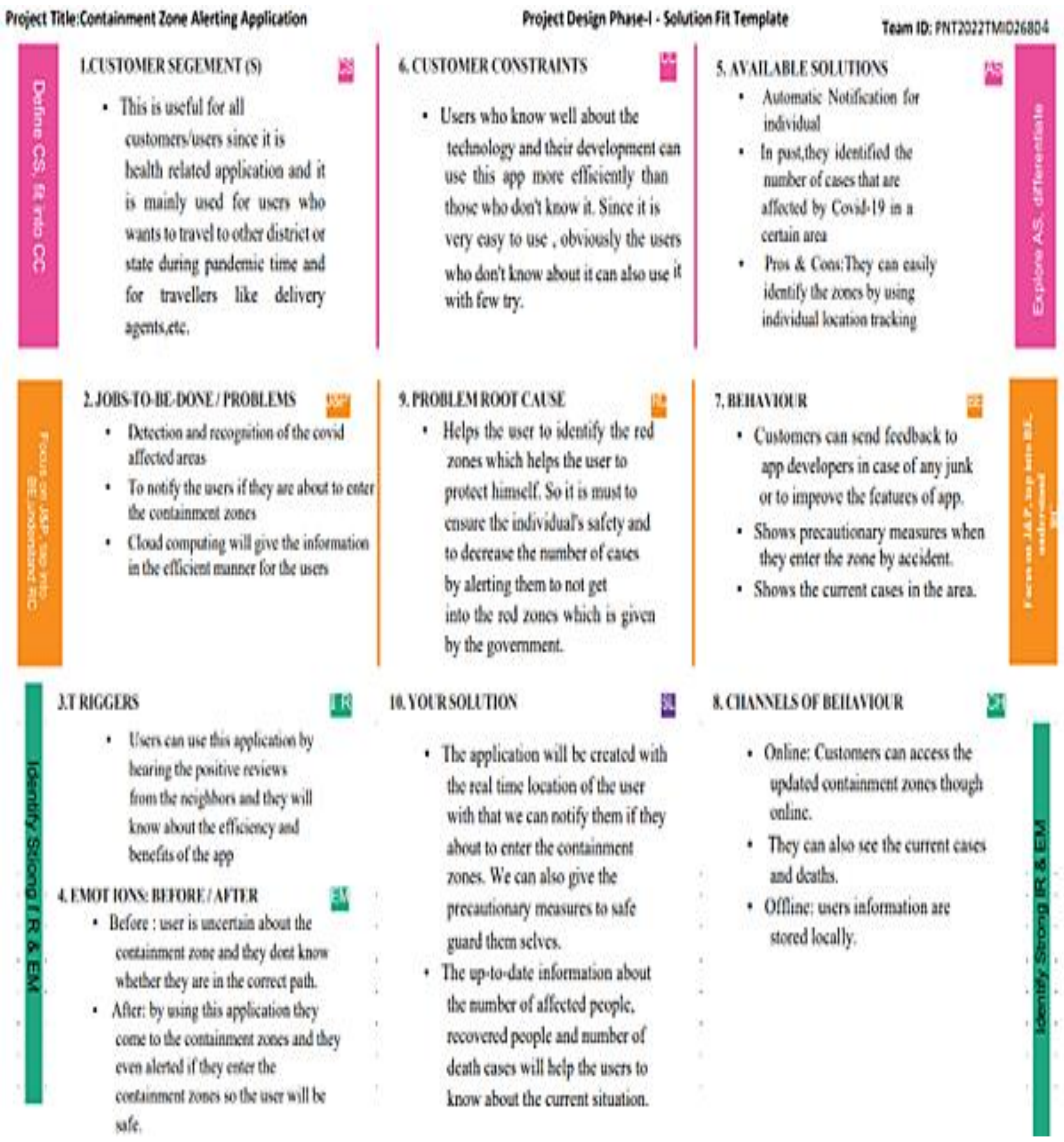
Idea Prioritization



4

Priorities

An importance - feasibility graph about what's important moving forward.



Project Title:Containment Zone Alerting Application

Project Design Phase-I - Solution Fit Template

Team ID: PNT2022TMID26804

Define CS, fit into CC

1.CUSTOMER SEGEMENT (S)

- This is useful for all customers/users since it is health related application and it is mainly used for users who wants to travel to other district or state during pandemic time and for travellers like delivery agents,etc.

6. CUSTOMER CONSTRAINTS

- Users who know well about the technology and their development can use this app more efficiently than those who don't know it. Since it is very easy to use , obviously the users who don't know about it can also use it with few try.

5. AVAILABLE SOLUTIONS

- Automatic Notification for individual
- In past,they identified the number of cases that are affected by Covid-19 in a certain area
- Pros & Cons:They can easily identify the zones by using individual location tracking

Explore AS, differentiate

Focus on J&P, tap into BE, understand RC

2. JOBS-TO-BE-DONE / PROBLEMS

- Detection and recognition of the covid affected areas
- To notify the users if they are about to enter the containment zones
- Cloud computing will give the information in the efficient manner for the users

9. PROBLEM ROOT CAUSE

- Helps the user to identify the red zones which helps the user to protect himself. So it is must to ensure the individual's safety and to decrease the number of cases by alerting them to not get into the red zones which is given by the government.

7. BEHAVIOUR

- Customers can send feedback to app developers in case of any junk or to improve the features of app.
- Shows precautionary measures when they enter the zone by accident.
- Shows the current cases in the area.

Focus on J&P, tap into BE, understand RC

Identify Strong TR & EM

3.TRIGGERS

- Users can use this application by hearing the positive reviews from the neighbors and they will know about the efficiency and benefits of the app

4. EMOTIONS: BEFORE / AFTER

- Before : user is uncertain about the containment zone and they dont know whether they are in the correct path.
- After: by using this application they come to the containment zones and they even alerted if they enter the containment zones so the user will be safe.

10. YOUR SOLUTION

- The application will be created with the real time location of the user with that we can notify them if they about to enter the containment zones. We can also give the precautionary measures to safe guard them selves.
- The up-to-date information about the number of affected people, recovered people and number of death cases will help the users to know about the current situation.

8. CHANNELS OF BEHAVIOUR

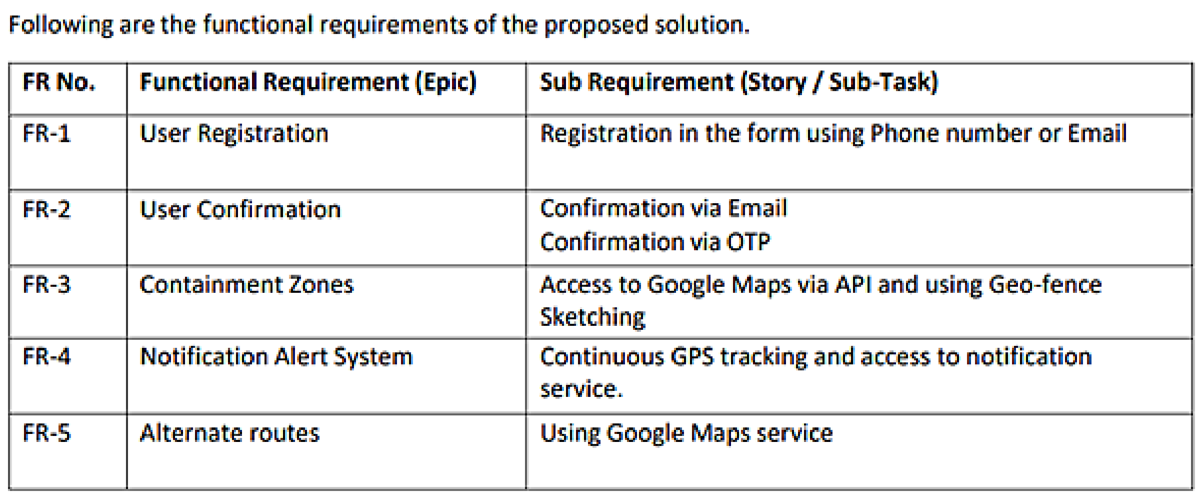
- Online: Customers can access the updated containment zones though online.
- They can also see the current cases and deaths.
- Offline: users information are stored locally.

Identify Strong TR & EM

3.4 Problem Solution fit

4.REQUIREMENT ANALYSIS

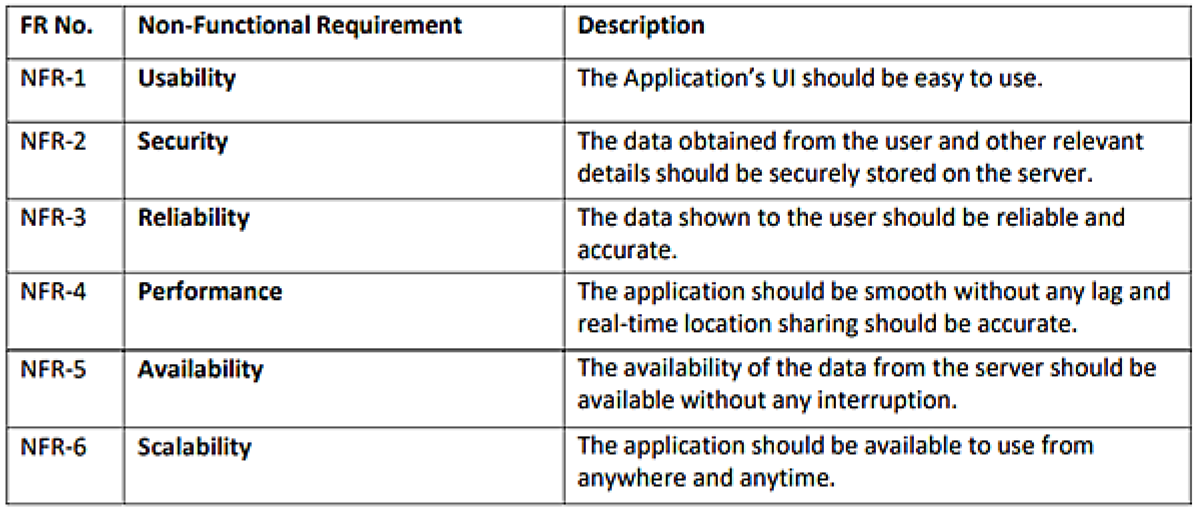
4.1 Functional requirement


Following are the functional requirements of the proposed solution.

FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	User Registration	Registration in the form using Phone number or Email
FR-2	User Confirmation	Confirmation via Email Confirmation via OTP
FR-3	Containment Zones	Access to Google Maps via API and using Geo-fence Sketching
FR-4	Notification Alert System	Continuous GPS tracking and access to notification service.
FR-5	Alternate routes	Using Google Maps service

4.2 Non-Functional requirements

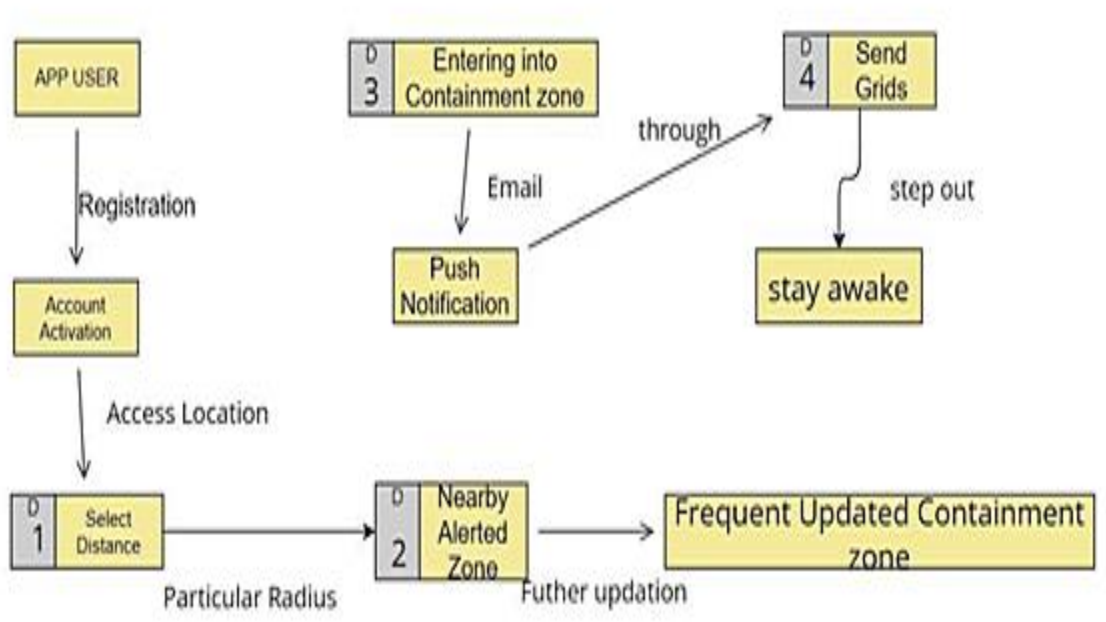
Following are the non-functional requirements of the proposed solution.


FR No.	Non-Functional Requirement	Description
NFR-1	**Usability**	The Application's UI should be easy to use.
NFR-2	**Security**	The data obtained from the user and other relevant details should be securely stored on the server.
NFR-3	**Reliability**	The data shown to the user should be reliable and accurate.
NFR-4	**Performance**	The application should be smooth without any lag and real-time location sharing should be accurate.
NFR-5	**Availability**	The availability of the data from the server should be available without any interruption.
NFR-6	**Scalability**	The application should be available to use from anywhere and anytime.

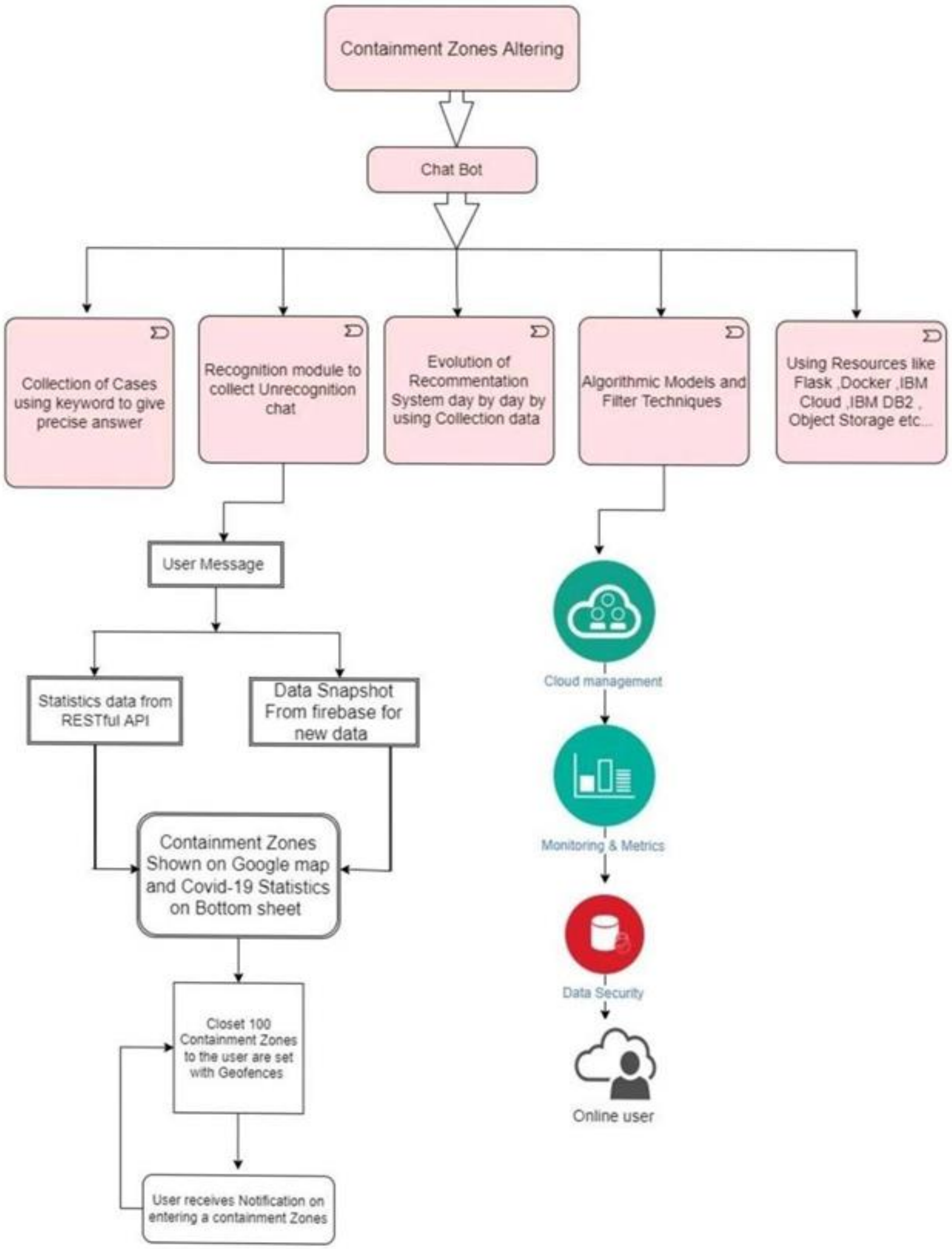
5.PROJECT DESIGN

5.1 Data Flow Diagrams:

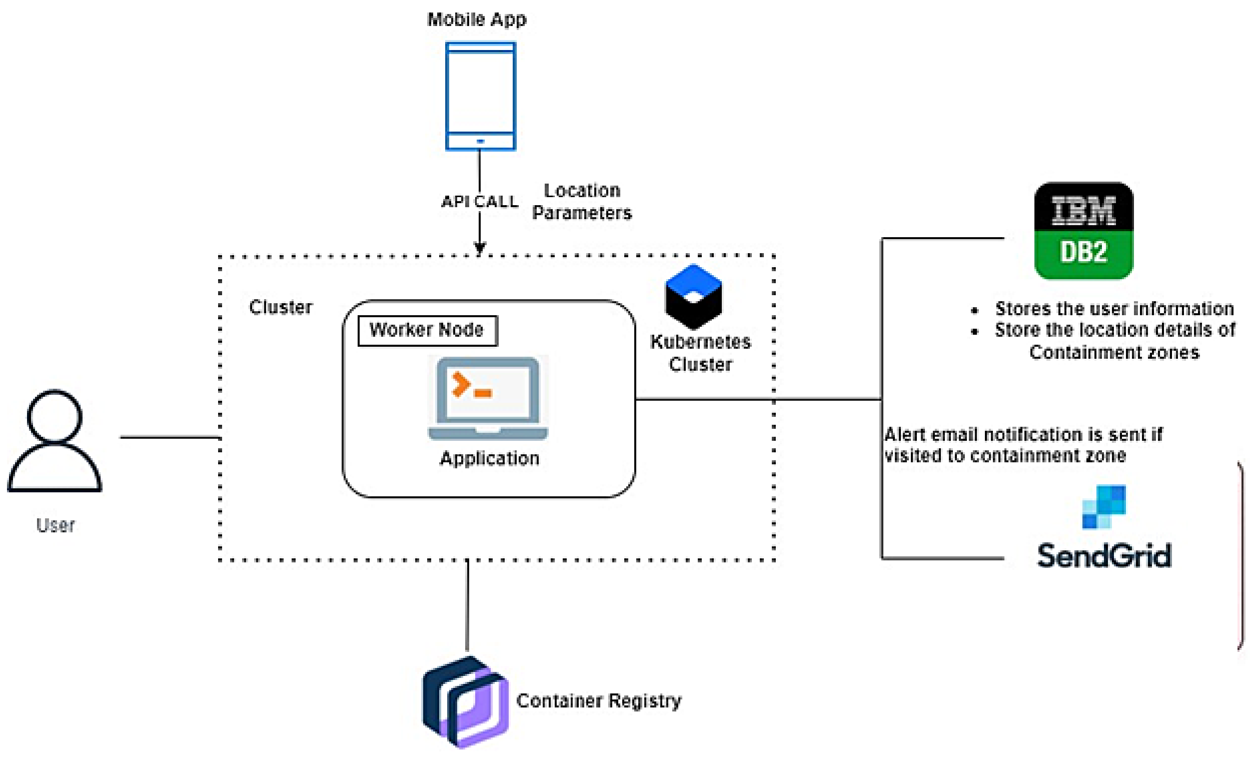
A Data Flow Diagram (DFD) is a traditional visual representation of the information flows within a system. A neat and clear DFD can depict the right amount of the system requirement graphically. It shows how data enters and leaves the system, what changes the information, and where data is stored.



5.2 SOLUTION ARCHITECURE & TECHNICAL ARCHITECTURE:



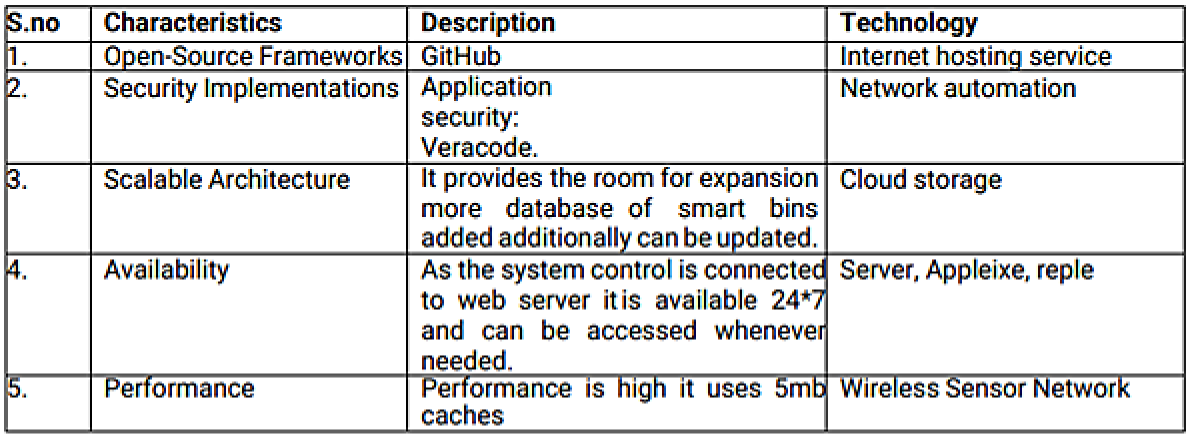
Containment Zones Altering
Chat Bot
Collection of Cases using keyword to give precise answer
Recognition module to collect Unrecognition chat
Evolution of Recommentation System day by day by using Collection data
Algorithmic Models and Filter Techniques
Using Resources like Flask ,Docker ,IBM Cloud ,IBM DB2 , Object Storage etc...
User Message
Statistics data from RESTful API
Data Snapshot From firebase for new data
Containment Zones Shown on Google map and Covid-19 Statistics on Bottom sheet
Closet 100 Containment Zones to the user are set with Geofences
User receives Notification on entering a containment Zones
Cloud management
Monitoring & Metrics
Data Security
Online user


Mobile App
API CALL
Location Parameters
Cluster
Worker Node
Application
Kubernetes Cluster
User
IBM DB2
Stores the user information
Store the location details of Containment zones
Alert email notification is sent if visited to containment zone
SendGrid
Container Registry

5.2 Table-1: Components & Technologies:

S.no	Component	Description	Technology
1.	User Interface	Mobile Application	HTML, CSS, JavaScript.
2.	Application Logic	Logic for a process in the application	Javascript
3.	Database	Data Type, Configurations etc.	Firebase, ibm cloud
4.	Cloud Database	Database Service on Cloud	IBM Cloud
5.	File Storage	File storage requirements	Local Filesystem and IBM cloud
6.	Infrastructure (Server / Cloud)	Application Deployment on Cloud Local Server Configuration	Local and Cloud Foundry

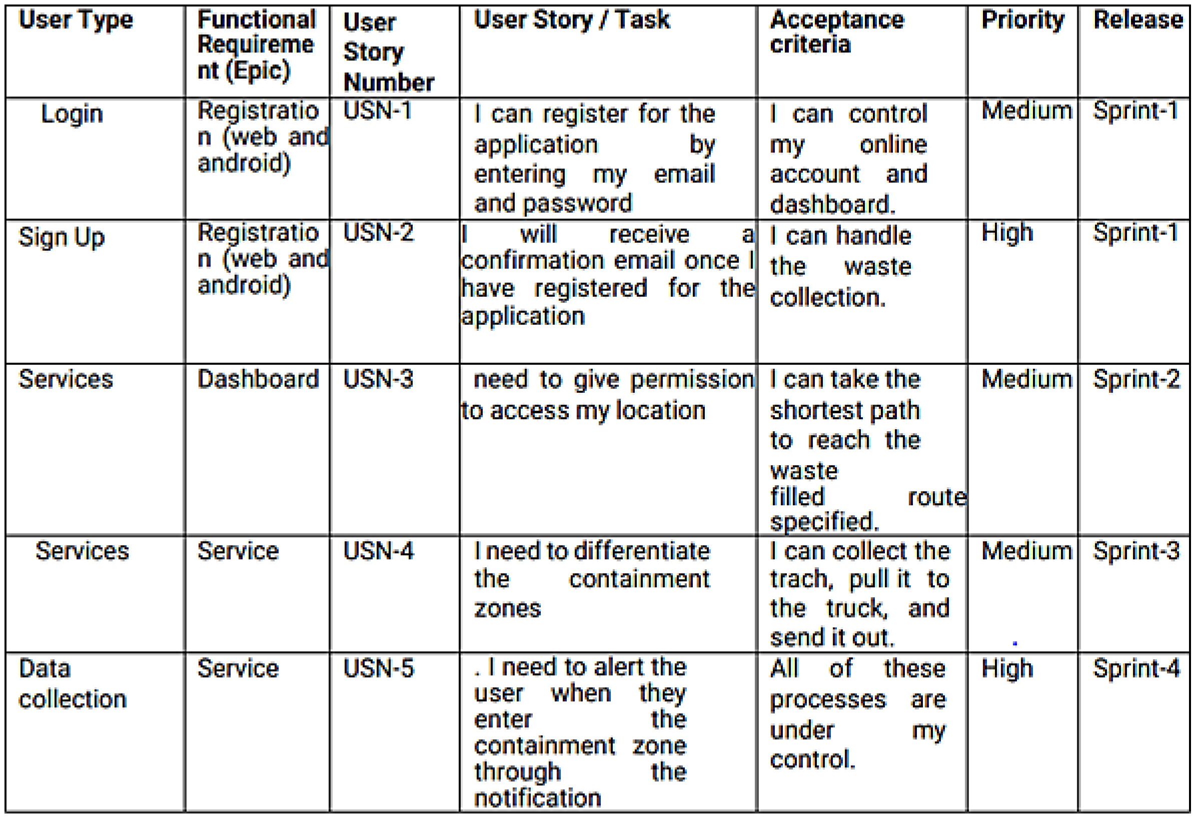
Application Characteristics:



S.no	Characteristics	Description	Technology
1.	Open-Source Frameworks	GitHub	Internet hosting service
2.	Security Implementations	Application security: Veracode.	Network automation
3.	Scalable Architecture	It provides the room for expansion more database of smart bins added additionally can be updated.	Cloud storage
4.	Availability	As the system control is connected to web server it is available 24*7 and can be accessed whenever needed.	Server, Appleixe, reple
5.	Performance	Performance is high it uses 5mb caches	Wireless Sensor Network

5.3 User Stories

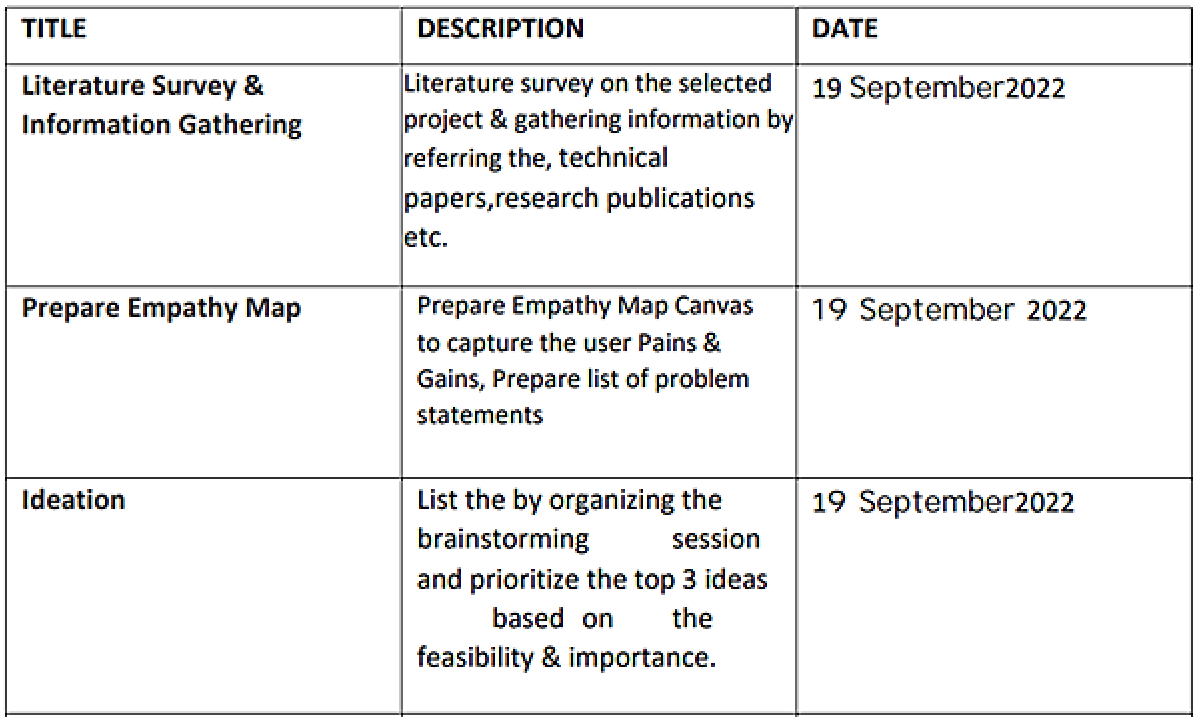
Use the below template to list all the user stories for the product.



User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
Login	Registration (web and android)	USN-1	I can register for the application by entering my email and password	I can control my online account and dashboard.	Medium	Sprint-1
Sign Up	Registration (web and android)	USN-2	I will receive a confirmation email once I have registered for the application	I can handle the waste collection.	High	Sprint-1
Services	Dashboard	USN-3	need to give permission to access my location	I can take the shortest path to reach the waste filled route specified.	Medium	Sprint-2
Services	Service	USN-4	I need to differentiate the containment zones	I can collect the trach, pull it to the truck, and send it out.	Medium	Sprint-3
Data collection	Service	USN-5	. I need to alert the user when they enter the containment zone through the notification	All of these processes are under my control.	High	Sprint-4

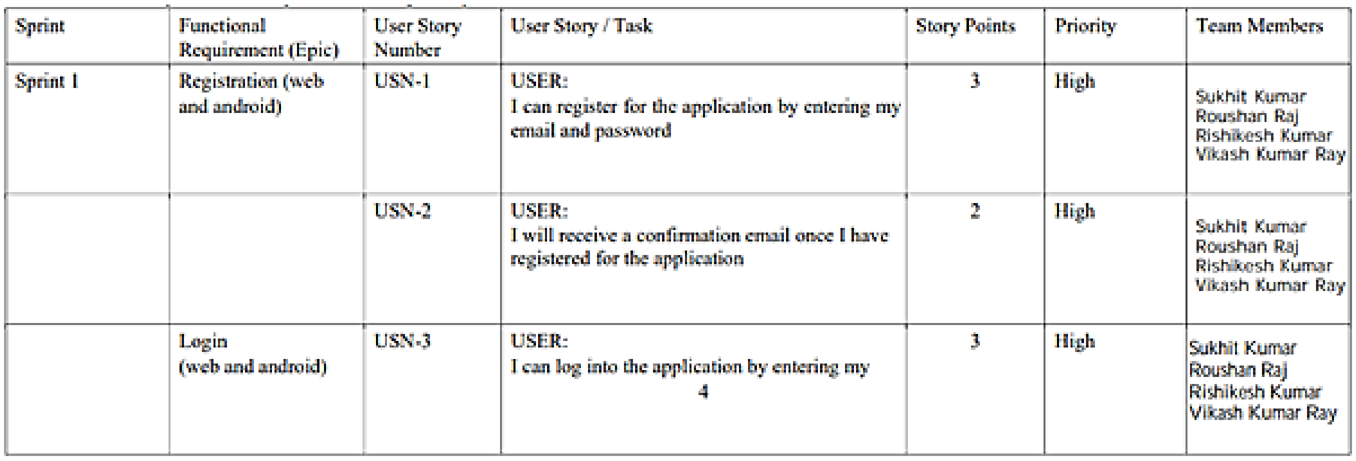
6. PROJECT PLANNING & SCHEDULING

6.1 Sprint Planning & Estimation

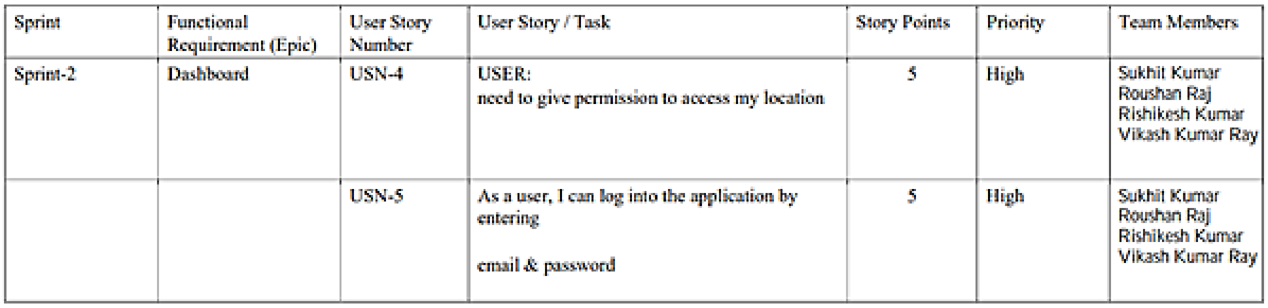


TITLE	DESCRIPTION	DATE
Literature Survey & Information Gathering	Literature survey on the selected project & gathering information by referring the, technical papers,research publications etc.	19 September2022
Prepare Empathy Map	Prepare Empathy Map Canvas to capture the user Pains & Gains, Prepare list of problem statements	19 September 2022
Ideation	List the by organizing the brainstorming session and prioritize the top 3 ideas based on the feasibility & importance.	19 September2022

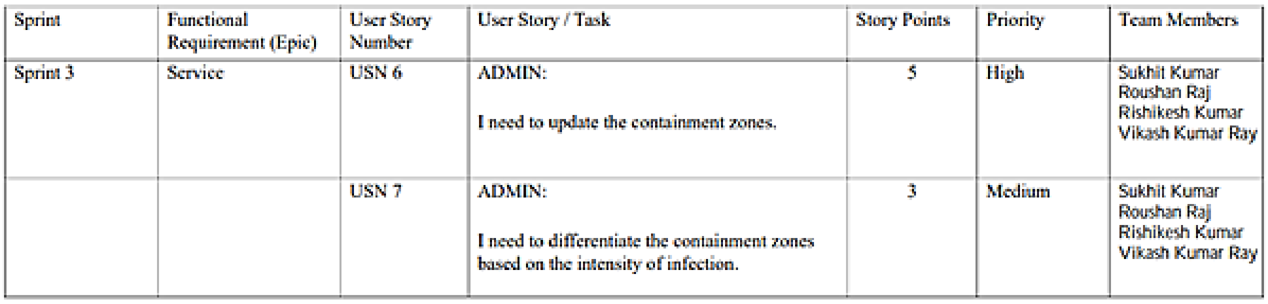
Sprint Schedule, and Estimation



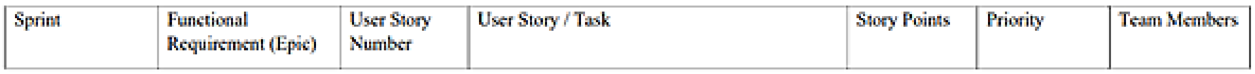
Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint 1	Registration (web and android)	USN-1	USER: I can register for the application by entering my email and password	3	High	Sukhit Kumar Roushan Raj Rishikesh Kumar Vikash Kumar Ray
		USN-2	USER: I will receive a confirmation email once I have registered for the application	2	High	Sukhit Kumar Roushan Raj Rishikesh Kumar Vikash Kumar Ray
	Login (web and android)	USN-3	USER: I can log into the application by entering my 4	3	High	Sukhit Kumar Roushan Raj Rishikesh Kumar Vikash Kumar Ray



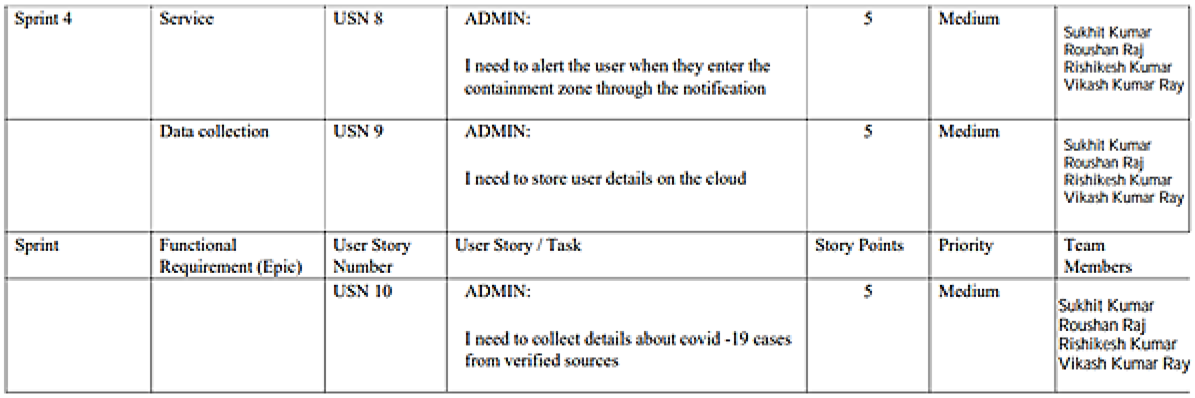
Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-2	Dashboard	USN-4	USER: need to give permission to access my location	5	High	Sukhit Kumar Roushan Raj Rishikesh Kumar Vikash Kumar Ray
		USN-5	As a user, I can log into the application by entering email & password	5	High	Sukhit Kumar Roushan Raj Rishikesh Kumar Vikash Kumar Ray



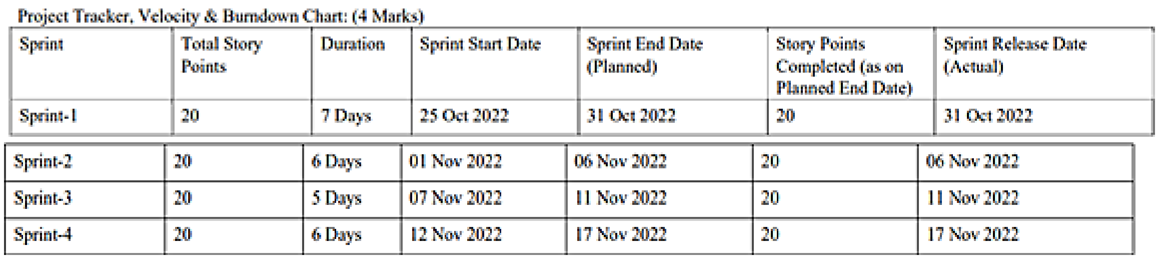
Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint 3	Service	USN 6	ADMIN: I need to update the containment zones.	5	High	Sukhit Kumar Roushan Raj Rishikesh Kumar Vikash Kumar Ray
		USN 7	ADMIN: I need to differentiate the containment zones based on the intensity of infection.	3	Medium	Sukhit Kumar Roushan Raj Rishikesh Kumar Vikash Kumar Ray



Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members



Sprint 4	Service	USN 8	ADMIN: I need to alert the user when they enter the containment zone through the notification	5	Medium	Sukhit Kumar Roushan Raj Rishikesh Kumar Vikash Kumar Ray
	Data collection	USN 9	ADMIN: I need to store user details on the cloud	5	Medium	Sukhit Kumar Roushan Raj Rishikesh Kumar Vikash Kumar Ray
Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
		USN 10	ADMIN: I need to collect details about covid -19 cases from verified sources	5	Medium	Sukhit Kumar Roushan Raj Rishikesh Kumar Vikash Kumar Ray



Project Tracker, Velocity & Burndown Chart: (4 Marks)

Sprint	Total Story Points	Duration	Sprint Start Date	Sprint End Date (Planned)	Story Points Completed (as on Planned End Date)	Sprint Release Date (Actual)
Sprint-1	20	7 Days	25 Oct 2022	31 Oct 2022	20	31 Oct 2022
Sprint-2	20	6 Days	01 Nov 2022	06 Nov 2022	20	06 Nov 2022
Sprint-3	20	5 Days	07 Nov 2022	11 Nov 2022	20	11 Nov 2022
Sprint-4	20	6 Days	12 Nov 2022	17 Nov 2022	20	17 Nov 2022

6.2 Sprint Delivery Schedule

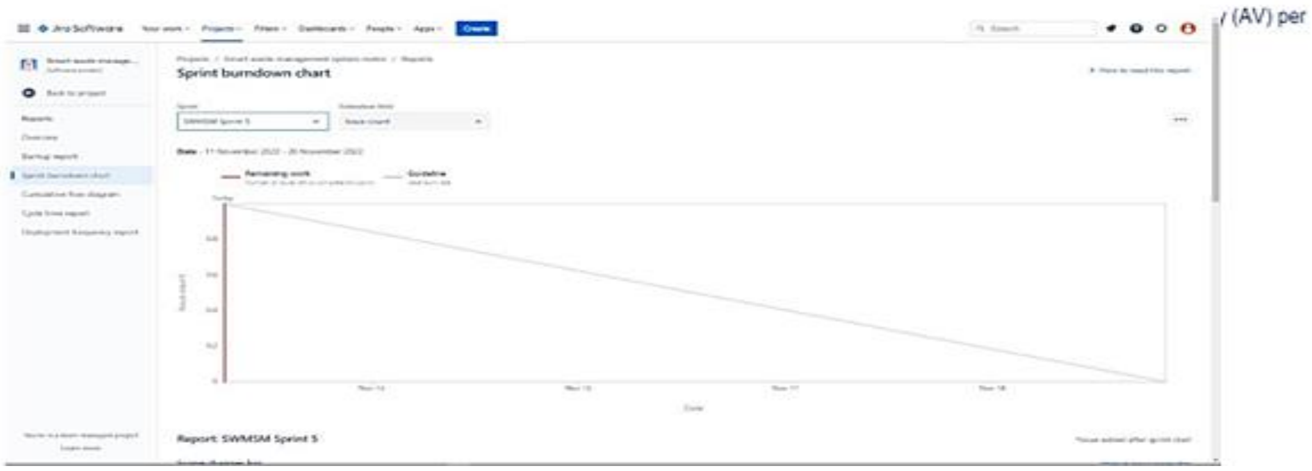
Velocity:

Imagine we have a 10-day sprint duration, and the velocity of the team is 20 (points per sprint). Let's calculate the team's average velocity (AV) per iteration unit (story points per day)

$$AV = \frac{sprint\ duration}{velocity} = \frac{20}{10} = 2$$

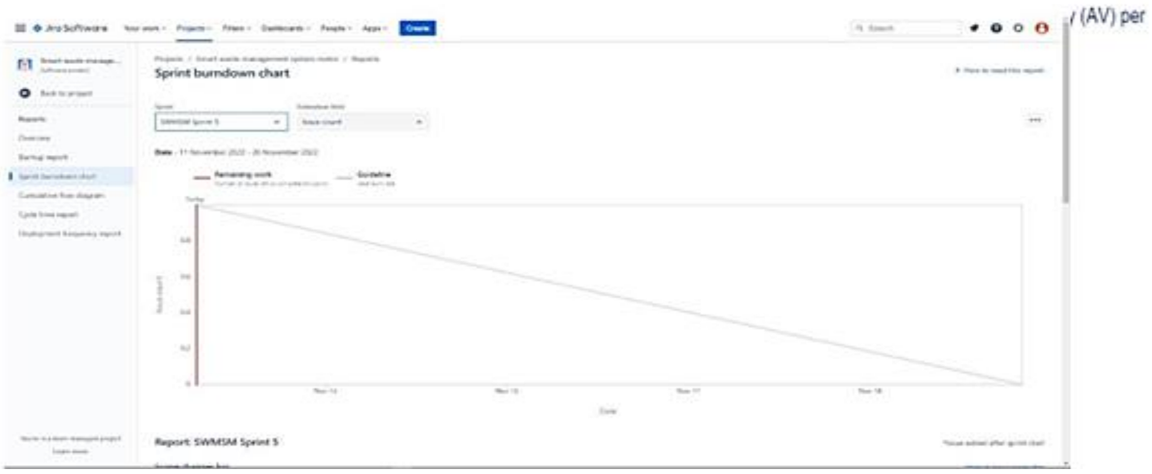
6.3 Reports from JIRA

BURNDOWN CHART



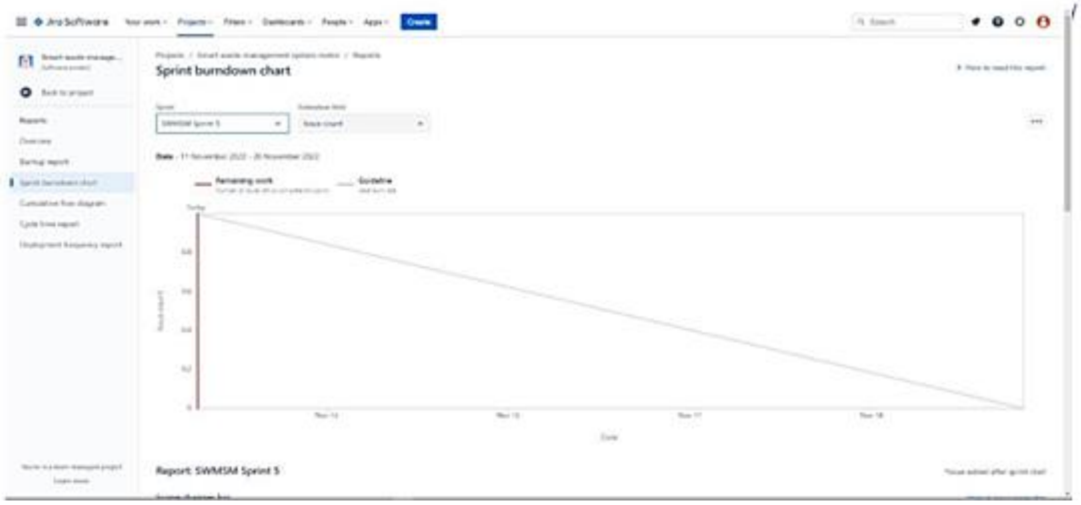
Report: SWMSM Sprint 5

BURNDOWN CHART



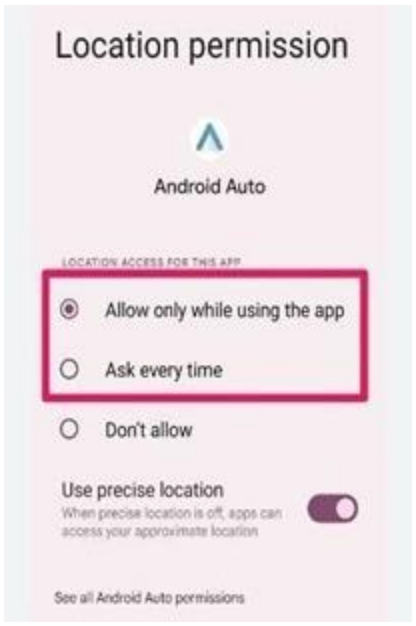
Report: SWMSM Sprint 5

BURNDOWN CHART



/ (AV) per

7.CODING & SOLUTIONING (Explain the features added in the project along with code)



Location permission

Android Auto

LOCATION ACCESS FOR THIS APP

[✓ Allow only while using the app]

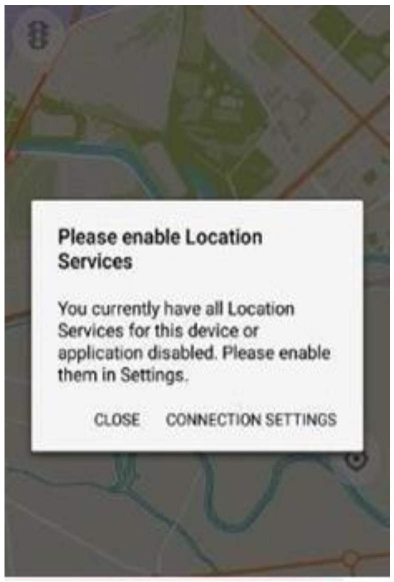
Ask every time

Don't allow

Use precise location

When precise location is off, apps can access your approximate location

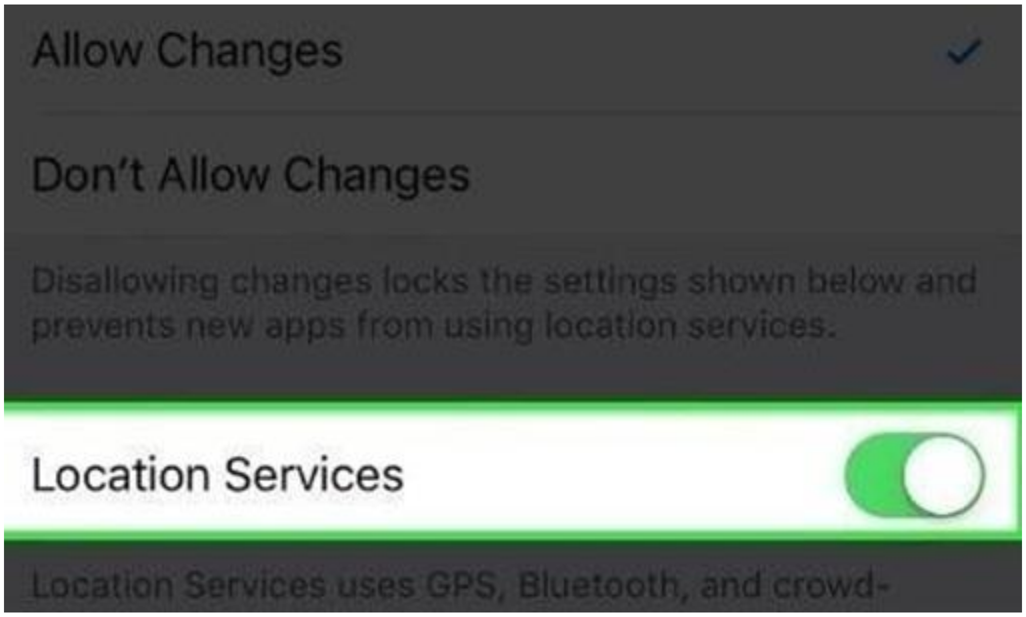
See all Android Auto permissions



Please enable Location Services

You currently have all Location Services for this device or application disabled. Please enable them in Settings.

CLOSE CONNECTION SETTINGS


Allow Changes ✓

Don't Allow Changes

Disallowing changes locks the settings shown below and prevents new apps from using location services.

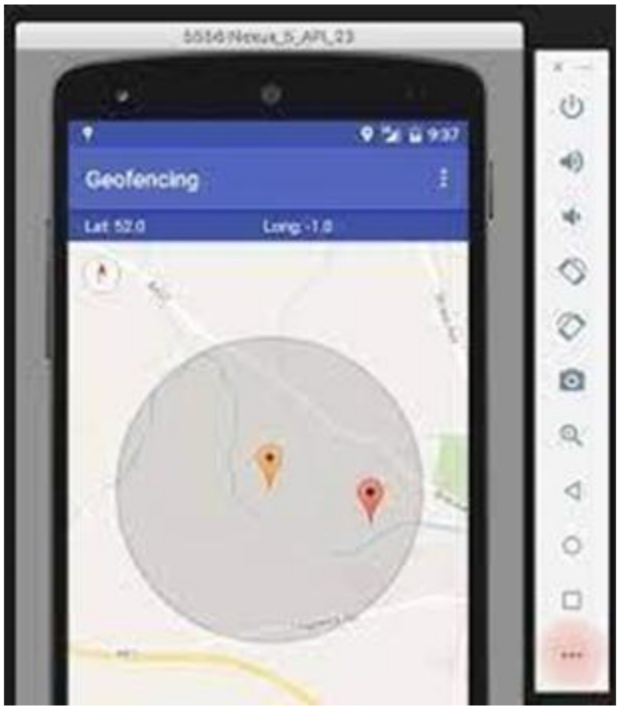
Location Services

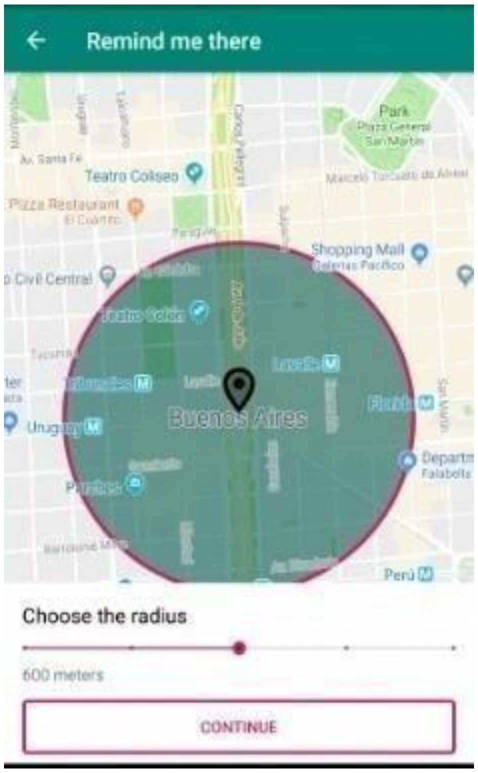
Location Services uses GPS, Bluetooth, and crowd-

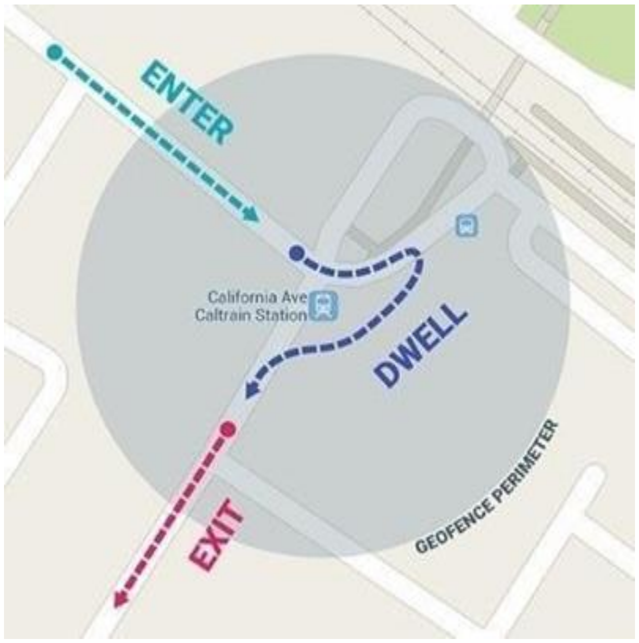
GEOFENCE IN ANDROID APP :

Geofencing
Lat: 52.0
Long: -1.0
Google

Geofencing





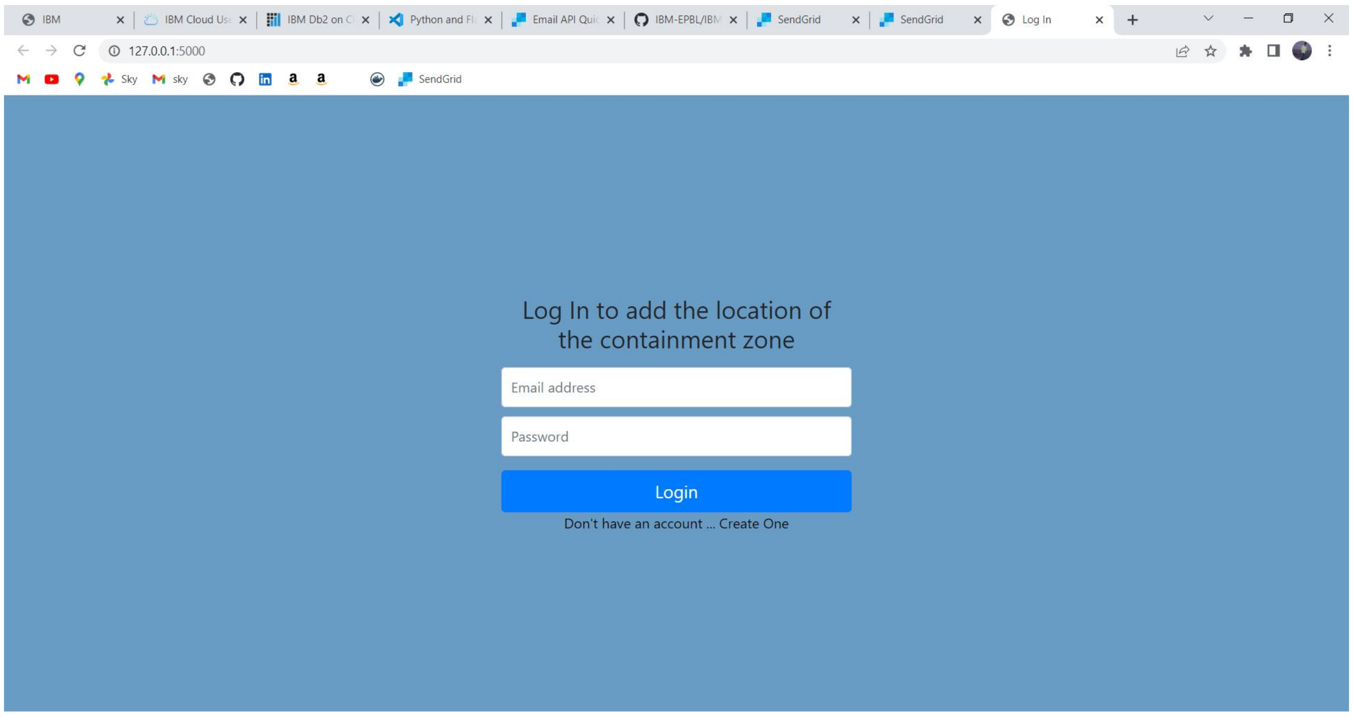


8.RESULTS:

8.1 UI Interact with Application:

8.2 Admin App:

8.2.1 Login Page:



Log In to add the location of the containment zone

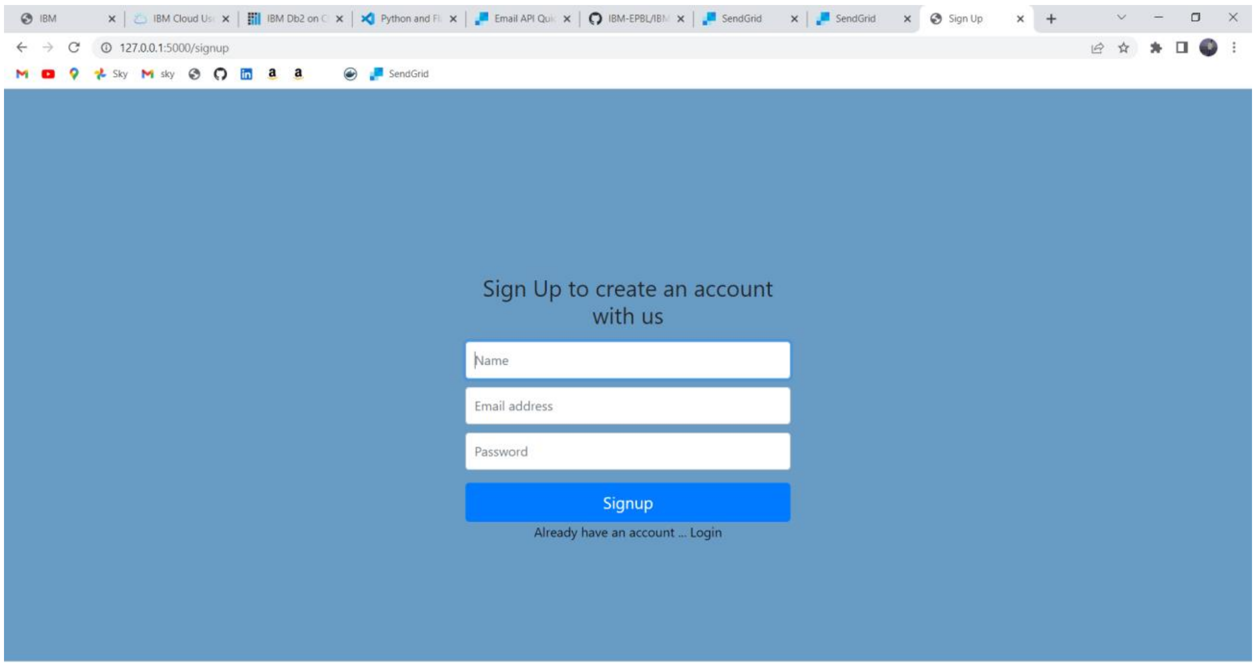
Email address

Password

Login

Don't have an account ... Create One

8.2.2 Register page:



Sign Up to create an account with us

Name

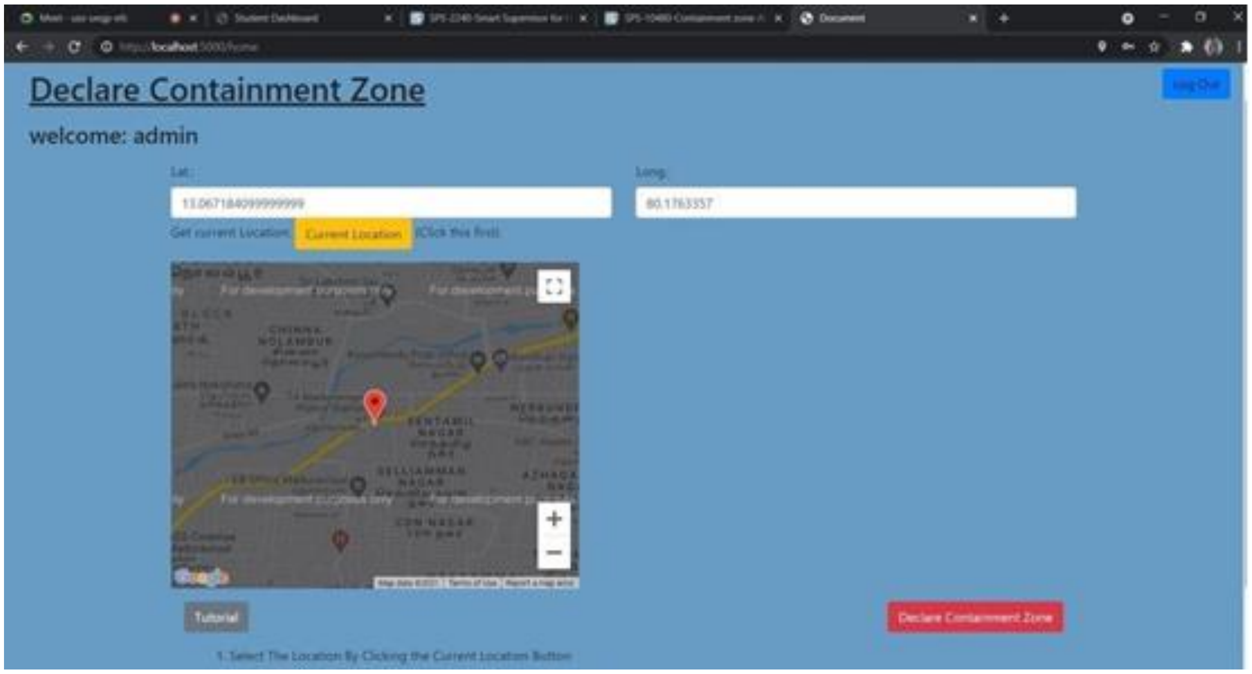
Email address

Password

Signup

Already have an account ... Login

8.2.3 Home page:



Declare Containment Zone

welcome: admin

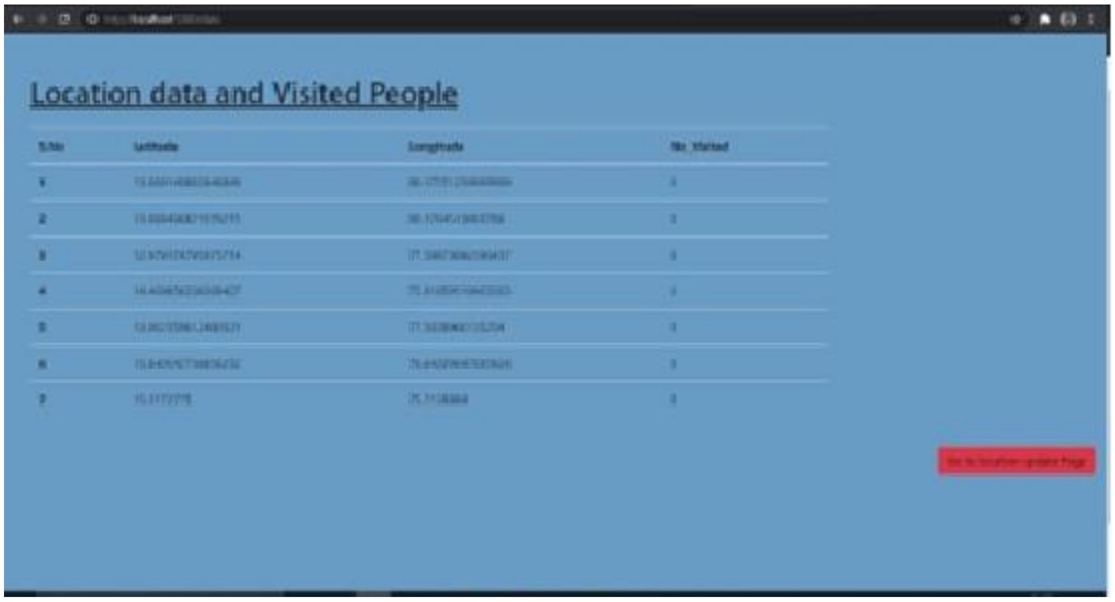
Lat: 13.067184099999999

Long: 80.1763357

Get current Location: Current Location (Click this first)

1. Select The Location By Clicking the Current Location Button

8.2.4 Location data page:

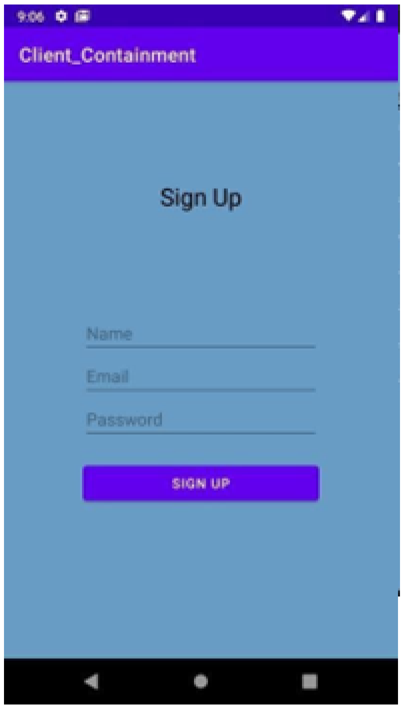


Location data and Visited People

S.No	Latitude	Longitude	No_Visited
1	[illegible]	[illegible]	[illegible]
2	[illegible]	[illegible]	[illegible]
3	[illegible]	[illegible]	[illegible]
4	[illegible]	[illegible]	[illegible]
5	[illegible]	[illegible]	[illegible]
6	[illegible]	[illegible]	[illegible]
7	[illegible]	[illegible]	[illegible]

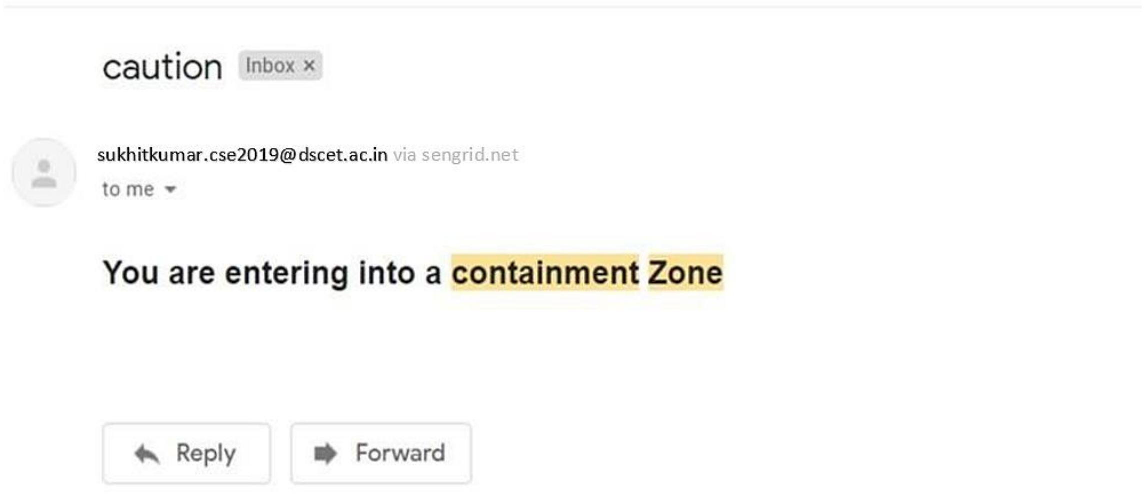
8.3 Client Application:

8.3.1 Register screen:



8.3.2 Current Location:

An Email will be sent to the registered mail id if the location is within 100 meters of the locations present in the admin app.


caution Inbox ×

sukhitkumar.cse2019@dscet.ac.in via sengrid.net
to me

You are entering into a containment Zone

9. ADVANTAGES & DISADVANTAGES

ADVANTAGES

• People can be alerted before entering containment zone.

• Further spread of virus can be reduced considerably.

DISADVANTAGES

• Accuracy of application depends on the number of data given to the application.

• Application’s accuracy is directly proportional to the number of data given to the application

• about the infected patients.

10.CONCLUSION

The COVID-19 pandemic continues to affect the way of life of everyone. The contact tracing apps are likely to play a vital role in aiding health authorities quickly identify individuals that may have been exposed to the virus. The imminent interest and adoption of tracing app technology will improve the tracing capability of health authorities; however, as this article highlighted, it is not a silver bullet. These apps still face many concerns from users, data protection agencies, and researchers. The main concerns are related to the user data management, potentially non-trivial false positive and negative instances, and the security and privacy issues of these apps. Guided by these concerns, this article presented an overview of the three common tracing app architectures: centralised, decentralised, and hybrid; and an overview of popular apps within these categories. Additionally, the paper focused on the privacy and security aspects, mapping attacks that could be possibly performed in each of the three architectures. This article also elucidates some other users’ concerns regarding battery drain, compatibility, consent withdrawal, and transparency. Finally, we discussed some of the near and long term future research directions.

11.FUTURE SCOPE

I though we tried to cover almost all of the aspects during our developmental phase, however we were forced to leave some aspects because of lack of time as well as monetary and other reasons. Just like in the field of software development where there are always some shortcomings and room for improvement our application can be enhanced further:-

1) The application can include various government organization to help act faster.

2) The dataset obtained from the application can be used for predictive analysis to determine prone areas and include special method for tackling the problem in those areas.

3) Emergency signal in case of network failure and internet connection loss.

4) Tackling victim’s movements.

5) Improved Google positioning system's precision.

6) The client part of application can be integrated in a single intelligent device.

For analysis purpose, we could use machine learning (ML) algorithms as well as data mining applications. There is a sub branch of machine learning known as time series analysis (TSA), which could be used to predict and analysed the data obtained through this application. Time series analysis is used to predict crop production as well as sales in different quarter.

12 APPENDIX

Source Code

```
# Project : CONTAINMENT ZONE ALERTING APPLICATION
 # Team ID : PNT2022TMID26804
```

```
# import statements
from loggingimport error
from flaskimport *
from jinja2.utilsimport select_autoescape
import bcrypt
from flask_mysqldbimport MySQL
import json
from sendgridimport SendGridAPIClient
from sendgrid.helpers.mailimport Mail


# initialization
app = Flask(__name__)

# config
app.secret_key =
"\xc0)\x1a\xf4\xa8\xf3\x0f\xec;\xbf\x90\x82g\xbb\x90{a\xc6\x89\xe3d\x9b\xf3Y"
```

```
app.config['MYSQL_HOST'] = 'localhost'
app.config['MYSQL_USER'] = 'root'
app.config['MYSQL_PASSWORD'] = ''
app.config['MYSQL_DB'] = 'zone2'

mysql = MySQL(app)

# functions


def send_mail(email):
    print(email)
    message = Mail(from_email='sukhitkumar.cse2019@dscet.ac.in',
                   to_emails=email,
                   subject='caution',
                   plain_text_content='Please Stay Safe',
                   html_content='<h2>You are entering into a containment
Zone</h2>')

    try:
        sg = SendGridAPIClient(
            'SG.-
TH16u3SSqSBm_lL8s4wnw.dErg0rHzxC3lAimoAtxRVXqodz5qb058ZrlIAMKFEGY')
        response = sg.send(message)
        print(response.status.code)
        print(response.body)
        print(response.headers)
    exceptException ase:
        print(e)


def create_bcrypt_hash(password):
    # convert the string to bytes
    password_bytes = password.encode()
    # generate a salt
    salt = bcrypt.gensalt(14)
    # calculate a hash as bytes
    password_hash_bytes = bcrypt.hashpw(password_bytes, salt)
    # decode bytes to a string
    password_hash_str = password_hash_bytes.decode()
```

```python
    returnpassword_hash_str


def verify_password(password, hash_from_database):
    password_bytes = password.encode()
    hash_bytes = hash_from_database.encode()

    # this will automatically retrieve the salt from the hash,
    # then combine it with the password (parameter 1)
    # and then hash that, and compare it to the user's hash
    does_match = bcrypt.checkpw(password_bytes, hash_bytes)

    returndoes_match

# Api's


@app.route("/", methods=["GET", "POST"])
def login():
    if(request.method == "POST"):

        # get the data from the form
        password = request.form['password']
        email = request.form['email']

        # initialize the cursor
        signup_cursor = mysql.connection.cursor()

        # check whether user already exists
        user_result = signup_cursor.execute(
            "SELECT * FROM USERS WHERE user_email=%s", [email]
        )

        if(user_result > 0):
            data = signup_cursor.fetchone()
            data_password = data[3]
            if(verify_password(password, data_password)):
                signup_cursor.close()
                session['id'] = data[0]
                session['name'] = data[1]
                session['email'] = data[2]
                return redirect(url_for("home"))
            else:
                return render_template('login.html', error=1)
        else:
            returnrender_template('login.html', error=2)
    returnrender_template('login.html', error=3)


@app.route("/signup", methods=["POST", "GET"])
def signup():
    if(request.method == "POST"):

        # get the data from the form
        name = request.form['name']
        email = request.form['email']
        password = request.form['password']

        # hash the password
        pw_hash = create_bcrypt_hash(password)

        # initialize the cursor
        signup_cursor = mysql.connection.cursor()

        # check whether user already exists
        user_result = signup_cursor.execute(
            "SELECT * FROM USERS WHERE user_email=%s", [email]
        )
        if(user_result > 0):
            signup_cursor.close()
            returnrender_template('signup.html', error=True)
        else:
            # execute the query
            signup_cursor.execute(
                'INSERT INTO USERS(user_name,user_email,user_password,user_type)
VALUES(%s,%s,%s,%s)', (
                    name, email, str(pw_hash), "2"
                )
            )
            mysql.connection.commit()
            signup_cursor.close()
            returnredirect(url_for('login'))

    returnrender_template('signup.html', error=False)


@app.route("/home", methods=["POST", "GET"])
def home():
    if(session['id'] == None):
        returnredirect(url_for('login'))

    if(request.method == "POST"):
        # get data
        lat = request.form["lat"]
        lon = request.form["lon"]
        vis = 0
        if(lat == "" orlon == ""):
            returnrender_template('home.html', name=session['name'],
email=session['email'], id=session['id'], success=0)

        # create a location cursor
        location_cursor = mysql.connection.cursor()

        # Execute the query
        location_cursor.execute(
            'INSERT INTO LOCATION(location_lat,location_long,location_visited)
VALUES(%s,%s,%s)', (
                lat, lon, vis
            )
        )
        mysql.connection.commit()
        location_cursor.close()
        returnrender_template('home.html', name=session['name'],
email=session['email'], id=session['id'], success=True)
    returnrender_template('home.html', name=session['name'],
email=session['email'], id=session['id'])


@app.route("/logout")
```

```
def logout():
    # remove the username from the session if it is there
    session['id'] = None
    session['name'] = None
    session['email'] = None
    returnredirect(url_for('login'))


@app.route("/data")
def data():
    if(session['id'] == None):
        returnredirect(url_for('login'))

    location_cursor = mysql.connection.cursor()

    # check whether user already exists
    user_result = location_cursor.execute(
        "SELECT * FROM LOCATION"
    )
    if(user_result == 0):
        returnrender_template("data.html", responses=0)
    else:
        res = location_cursor.fetchall()
        print(res)
        returnrender_template("data.html", responses=res)


@app.route("/android_sign_up", methods=["POST"])
def upload():
    if(request.method == "POST"):

        # get the data from the form
        name = request.json['name']
        email = request.json['email']
        password = request.json['password']

        # hash the password
        pw_hash = create_bcrypt_hash(password)

        # initialize the cursor
        signup_cursor = mysql.connection.cursor()

        # check whether user already exists
        user_result = signup_cursor.execute(
            "SELECT * FROM USERS WHERE user_email=%s", [email]
        )
        if(user_result > 0):
            signup_cursor.close()
            return {'status': 'failure'}
        else:
            # execute the query
            signup_cursor.execute(
                'INSERT INTO USERS(user_name,user_email,user_password,user_type)
VALUES(%s,%s,%s,%s)', (
                    name, email, str(pw_hash), "1"
                )
            )

            mysql.connection.commit()
            id_result = signup_cursor.execute(
                'SELECT user_id FROM USERS WHERE user_email = %s', [email]
            )
            if(id_result > 0):
                id = signup_cursor.fetchone()
                return {"id": id[0]}
            signup_cursor.close()

    return {"status": "failure"}


@app.route("/get_all_users")
def getusers():
    signup_cursor = mysql.connection.cursor()

    # check whether user already exists
    user_result = signup_cursor.execute(
        "SELECT * FROM USERS"
    )
    if(user_result > 0):
        rv = signup_cursor.fetchall()
        row_headers = [x[0] for xin signup_cursor.description]
        json_data = []
        forresult inrv:
            json_data.append(dict(zip(row_headers, result)))
        returnjson.dumps(json_data)


@app.route("/post_user_location_data", methods=["POST"])
def post_user_location():
    if(request.method == "POST"):

        # get the data from the form
        lat = request.json['lat']
        lon = request.json['long']
        id = request.json['id']
        ts = request.json['timestamp']

        # initialize the cursor
        user_location_cursor = mysql.connection.cursor()

        # execute the query
        user_location_cursor.execute(
            'INSERT INTO
USER_LOCATION(location_lat,location_long,user_id,timestamp) VALUES(%s,%s,%s,%s)',
(
                lat, lon, id, ts
            )
        )

        mysql.connection.commit()

        return {"response": "success"}


@app.route("/location_data")
def location_data():
    location_cursor = mysql.connection.cursor()

    # check whether user already exists
    user_result = location_cursor.execute(
```

```python
        "SELECT * FROM LOCATION"
    )
    if(user_result != 0):
        res = location_cursor.fetchall()
        print(res)
        row_headers = [x[0] for xin location_cursor.description]
        json_data = []
        forresult inres:
            json_data.append(dict(zip(row_headers, result)))
        returnjson.dumps(json_data)
    else:
        return {"response": "failure"}


@app.route("/send_trigger", methods=["POST"])
def send_trigger():
    if(request.method == "POST"):
        # get the data from the form
        email = request.json['email']
        location_id = request.json['id']
        location_cursor = mysql.connection.cursor()

        # check whether user already exists
        user_result = location_cursor.execute(
            "SELECT location_visited FROM LOCATION WHERE location_id=%s", [
                location_id]
        )
        if(user_result == 0):
            return {"response": "failure"}
        else:
            res = location_cursor.fetchone()
            print(res[0])
            visited = res[0]
            visited = visited+1
            location_cursor.execute(
                "UPDATE LOCATION SET location_visited = %s WHERE location_id=%s",
                (visited, location_id)
            )
            mysql.connection.commit()
        send_mail(email)
        return {"response": "success"}


# main
if __name__ == "__main__":
    app.run(host='0.0.0.0', port=5000)
```

DATA.HTML

```
<!DOCTYPEhtml>
<htmllang="en">

<head>
    <metacharset="UTF-8">
    <metahttp-equiv="X-UA-Compatible"content="IE=edge">
    <metaname="viewport"content="width=device-width, initial-scale=1.0">
    <title>Zones</title>

<linkrel="stylesheet"href="https://stackpath.bootstrapcdn.com/bootstrap/4.4.1/css
/bootstrap.min.css"
        integrity="sha384-
Vkoo8x4CGsO3+Hhxv8T/Q5PaXtkKtu6ug5TOeNV6gBiFeWPGFN9MuhOf23Q9Ifjh"crossorigin="ano
nymous"/>
    <style>
        body {
            padding-top: 30px;
            padding-bottom: 30px;
            background-color: #699cc5;
        }

        a {
            color: black;
        }
    </style>
</head>

<body>
    <divclass="m-4 container">
        <h1><u>Location data and Visited People</u></h1>
    </div>
    <divclass="m-4 container">
        <tableclass="table">
            <thead>
                <tr>
                    <thscope="col">S.No</th>
                    <thscope="col">Latitude</th>
                    <thscope="col">Longitude</th>
                    <thscope="col">No_Visited</th>
                </tr>
            </thead>
            <tbody>


                <tr>
                    <thscope="row">{{loop.index}}</th>
                    <td>{{row[1]}}</td>
                    <td>{{row[2]}}</td>
                    <td>{{row[3]}}</td>
                </tr>

            </tbody>
        </table>
    </div>
    <divclass="m-3 float-right">
        <buttontype="button"class="btn btn-danger"><ahref="home.html">Go to
location update Page</a></button>
    </div>

</body>

</html>
```

HOME.HTML

```
<!DOCTYPEhtml>
<htmllang="en">

<head>
    <metacharset="UTF-8">
    <metahttp-equiv="X-UA-Compatible"content="IE=edge">
    <metaname="viewport"content="width=device-width, initial-scale=1.0">
    <title>Document</title>

<linkrel="stylesheet"href="https://stackpath.bootstrapcdn.com/bootstrap/4.4.1/css
/bootstrap.min.css"
        integrity="sha384-
Vkoo8x4CGsO3+Hhxv8T/Q5PaXtkKtu6ug5TOeNV6gBiFeWPGFN9MuhOf23Q9Ifjh"crossorigin="ano
nymous"/>
    <style>
        body {
            padding-top: 30px;
            padding-bottom: 30px;
            background-color: #699cc5;
        }

        a {
            color: black;
        }
    </style>
</head>

<body>
    {% if success == True %}
    <script>
        alert("Location Uploaded Successfully");
    </script>
    {% elif success == 0 %}
    <script>
        alert("Enter Proper Location data");
    </script>
    {% endif %}
    <divclass="m-3 float-right">
        <buttontype="button"class="btn btn-primary"><ahref="home.html">Log
Out</a></button>
    </div>
    <divclass="container m-3">
        <h1><u>Declare Containment Zone</u></h1>
    </div>
    <divclass="container m-3">
        <h3>welcome: {{name}}</h3>
    </div>
    <formmethod="POST"action="/home">
        <divclass="container">
            <divclass="form-group row">
                <divclass="col-sm-6">
                    <labelclass="control-label">Lat.:</label>
                    <inputtype="text"class="form-control"id="lat"name="lat"/>
                </div>
                <divclass="col-sm-6">
                    <label>Long.:</label>
                    <inputtype="text"class="form-control"id="lon"name="lon"/>
                </div>
                <divclass="col-sm-6">
                    <label>Get current Location:</label>
                    <buttontype="button"class="btn btn-warning"
onclick="getLocation()">Current Location</button>
                    <label>(Click this first)</label>
                </div>
            </div>

            <!-- map -->
            <divid="map_disp"style="height: 400px;width: 500px;"></div>
            <divclass="m-3 float-right">
                <buttontype="submit"class="btn btn-danger">Declare Containment
Zone</button>
            </div>
            <divclass="m-3">
                <buttononclick="toggleTips()"type="button"class="btn btn-
secondary">Tutorial</button>
                <divid="tips"class="m-3">
                    <ol>
                        <li>Select The Location By Clicking the Current Location
Button</li>
                        <li>Drag the Pin to change the location</li>
                        <li>Click on Declare Containment Zone to save the
location to the database </li>
                    </ol>
                </div>
            </div>
            <divclass="m-3 float-right">
```

```
                <buttontype="button"class="btn btn-
warning"><ahref="{{url_for('data')}}">Click Here To View The
                        Containment Zones and Number of
                        people visited</a></button>
            </div>
        </div>


<scriptsrc="https://cdn.jsdelivr.net/npm/bootstrap@4.6.0/dist/js/bootstrap.min.js
"
            integrity="sha384-
+YQ4JLhjyBLPDQt//I+STsc9iw4uQqACwlvpslubQzn4u2UU2UFM80nGisd026JF"
            crossorigin="anonymous"></script>
        <scriptsrc="https://code.jquery.com/jquery-2.2.4.min.js"></script>

<scriptsrc="https://maps.google.com/maps/api/js?sensor=false&amp;libraries=places
"></script>
        <script
            src="https://rawgit.com/Logicify/jquery-locationpicker-
plugin/master/dist/locationpicker.jquery.js"></script>

        <script>
            functiongetLocation() {
                if (navigator.geolocation) {
                    navigator.geolocation.getCurrentPosition(showPosition);
                } else {
                    alert("No location");
                }
            }
            functionshowPosition(position) {
                $('#map_disp').locationpicker({
                    location: {
                        latitude:position.coords.latitude,
                        longitude:position.coords.longitude
                    },
                    radius:0,
                    inputBinding: {
                        latitudeInput:$('#lat'),
                        longitudeInput:$('#lon'),
                    },
                    enableAutocomplete:true,
                    onchanged:function (currentLocation, radius, isMarkerDropped)
{
                        // Uncomment line below to show alert on each Location
Changed event
                        // alert("Location changed. New location (" +
currentLocation.latitude + ", " + currentLocation.longitude + ")");
                    }
                });
            }
            functiontoggleTips() {
                var x = document.getElementById("tips");
                if (x.style.display === "none") {
                    x.style.display = "block";
                } else {
                    x.style.display = "none";
                }
            }
        </script>
</body>

</html>
```

GitHub Link:

https://github.com/IBM-EPBL/IBM-Project-28587-1660113984

Video Demo Link:

https://vimeo.com/772813932