Project Report Format

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 Firestore 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

"Development of An Android Application for Viewing Covid-19 Containment Zones and Monitoring Violators Who are Trespassing into It Using Firebase and Geo fencing ."

The World Health Organization has declared the outbreak of the novel corona virus, 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 Geo fencing 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.

"Mobile Health Apps That Help With COVID-19 Management: Scoping Review."

Mobile health (m Health) apps have played an important role in mitigating the corona virus disease (COVID-19) response. However, there is no resource that provides a holistic picture of the available m Health apps that have been developed to combat this pandemic. Our aim is to scope the evidence base on apps that were developed in response to COVID-19.

"COVID19-Tracker: A shiny app to produce to produce comprehensive data visualization for SARS-CoV-2 epidemic in Spain."

Data visualization is an important tool for exploring and communicating findings in medical research, and specially in epidemiological surveillance. The COVID19-Tracker app systematically produces daily updated data visualization and analysis of SARS-CoV-2 epidemic in Spain. It collects automatically daily data on COVID-19 diagnosed cases, intensive care unit admissions, and mortality, from February 24th, 2020 onwards. Two applications have already been developed; 1) to analyze data trends and estimating short-term projections; and 2) To assess the effect of the lockdown on the trend of incident data. We are currently planning to improve the app by uploading shortly new applications for data visualization and analysis, which may help for a better understanding of the SARS-CoV-2 epidemic data in Spain.

"Mobile Geo Fencing Triggers for Alerting Entries Into COVID-19 Containment Zones using IOT."

In a thickly populated nation like India, it is hard to forecast community transmission of COVID-19. Hence, a number of containment zones had been recognized all over the country separated into red, orange, and green zones, individually. People are restricted to move into these containment zones. This chapter focuses on informing the public about the containment zone when they are in travel and also sends an alert to the police when a person enters the containment zone without permission using the containment zone alert system. This chapter suggests a containment zone alert system by means of geo-fencing technology to identify the movement of public, deliver info about the danger to the public in travel and also send an alert to the police when there is an entry or exit detected in the containment zone by the use of location-based services (LBS). By creating a fence virtually called geo-fence at the containment zones established based on the government info, this system monitors public movements like entry and exit to fence.

2.2 References

Reference 1: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC7328652/
Reference 2: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC7467120/
Reference 3: https://www.medrxiv.org/content/10.1101/2020.04.01.20049684v1
Reference 4: https://www.semanticscholar.org/paper/Mobile-Geo-Fencing-Triggers-for-Alerting- Entries-Rao-Adilakshmi/c1925d5f9c97a42523e7eaf76a35d2c5eeac4ca1

2.3 Problem Statement Definition

Create a problem statement to understand your customer's point of view. The Customer Problem Statement template helps you focus on what matters to create experiences people will love. A well-articulated customer problem statement allows you and your team to find the ideal solution for the challenges your customers face. Throughout the process, you'll also be able to empathize with your customers, which helps you better understand how they perceive your product or service.

Problem Statement (PS)	Iam (Customer)	I'm trying to	But	Because	Which makes me feel
PS-1	Kala	Set an alert and notification while exceed enter the containment zone.	I am do not have awareness in this area is containment zone.	Then only she will more spend alert.	Excited.
PS-2	Vani	An bug free app for easy access.	Needs bug free up for easy access.	She know the identify containment zone in faster way.	Curious.
PS-3	Pooja	To maintain a privacy in her details.	She need privcy.	The user login detail should not be hacked.	User friendly.
PS-4	Keerthi	Monitoring the containment zone.	She need monitoring the zone.	She want to prevent anyone from entering the zone.	More efficient.
PS-5	Mohan	Provide faster alert.	He need faster alert.	He can save the time.	Excellent.

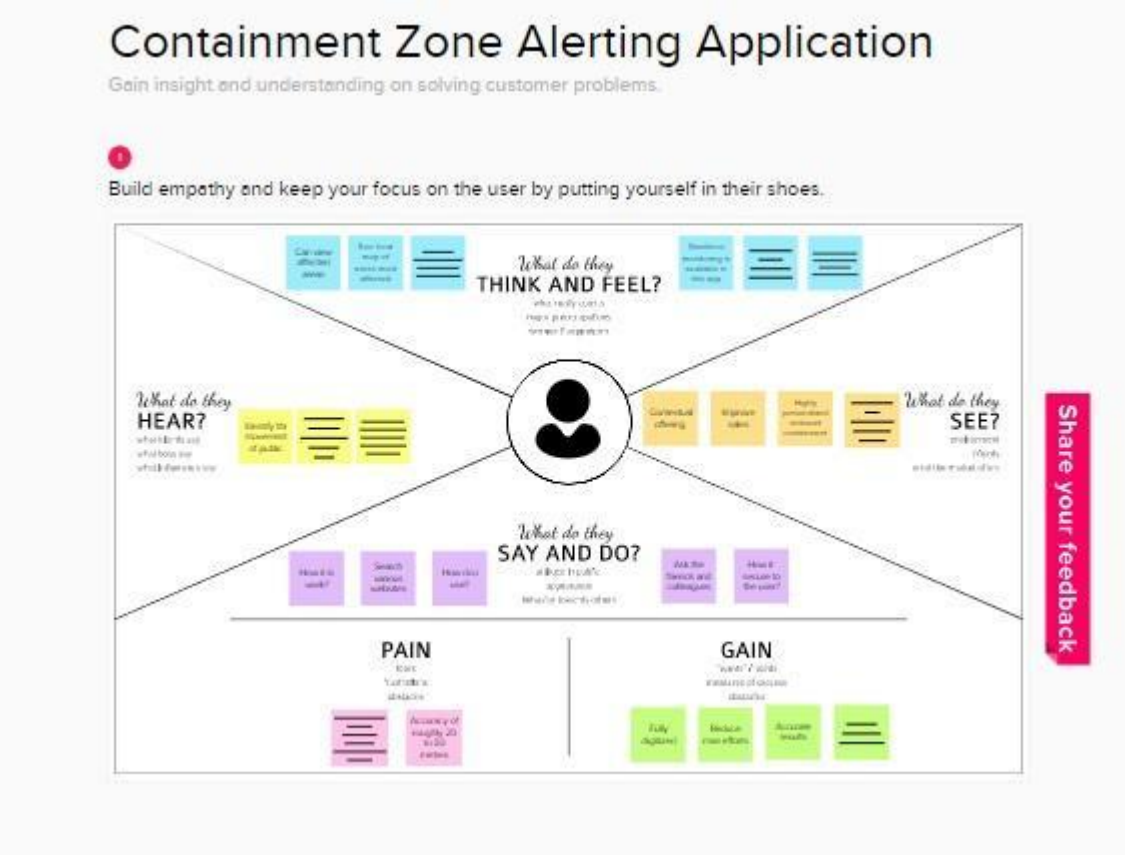
3. IDEATION & PROPOSED SOLUTION

3.1 Empathy Map Canvas

An empathy map is a simple, easy-to-digest visual that captures knowledge about user's behaviors 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.

Reference: https://app.mural.co/t/student77565/m/student77565/1662481430933/57f14d0a227dea6a83c7058613952bc6d7c04e47?sender=ube599e3b1bddeb53617e7738



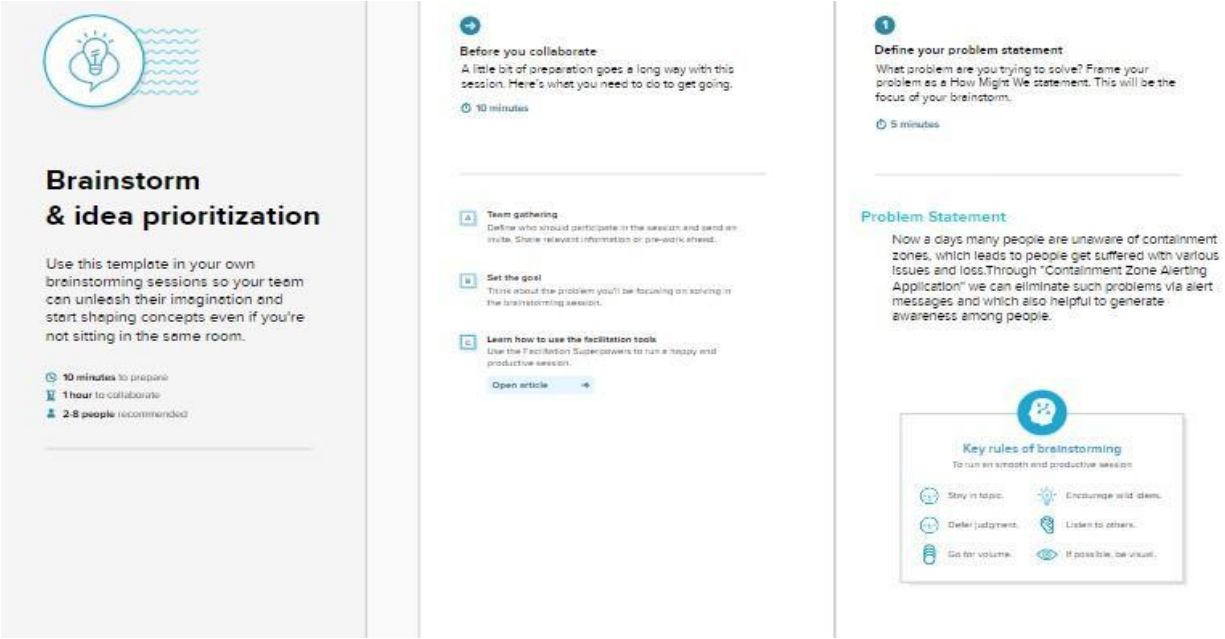
3.2 Ideation & Brainstorming

Brainstorming provides a free and open environment that encourages everyone within a team to participate in the creative thinking process that leads to problem solving. Prioritizing volume over value, out-of-the-box ideas are welcome and built upon, and all participants are encouraged to collaborate, helping each other develop a rich amount of creative solutions.

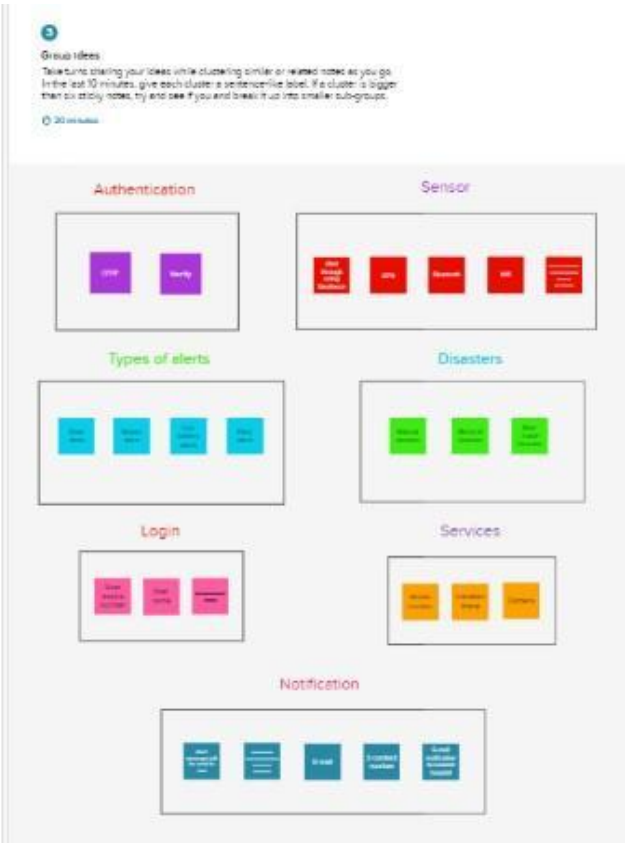
Use this template in your own brainstorming sessions so your team can unleash their imagination and start shaping concepts even if you're not sitting in the same room.

Reference:https://app.mural.co/t/student77565/m/student77565/1663350235967/80d0f8fb5336b14380d45a849f0d103b411fe567?sender=ube599e3b1bddeb53617e7738

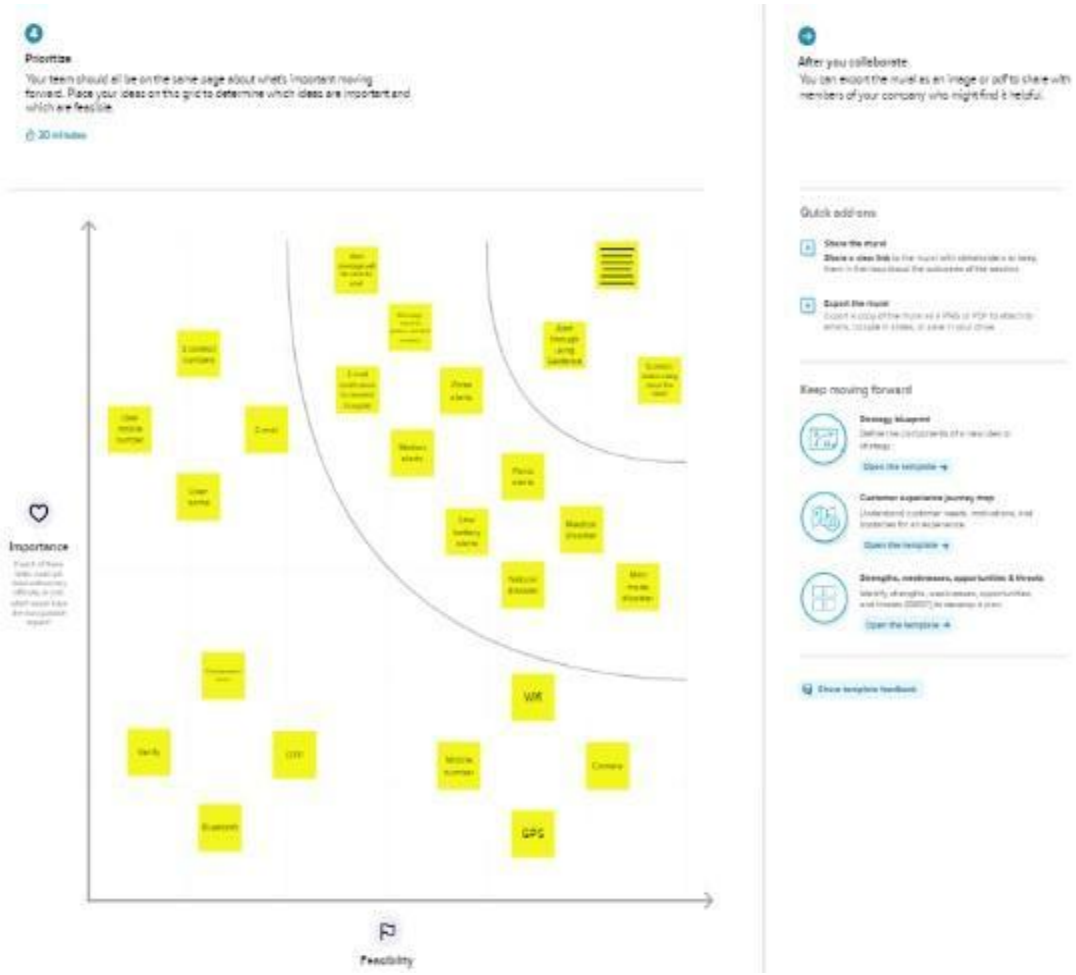
Step-1: Team Gathering, Collaboration and Select the Problem Statement



Step-2: Brainstorm, Idea Listing and Grouping



Step-3: Idea Prioritization



3.3 Proposed Solution

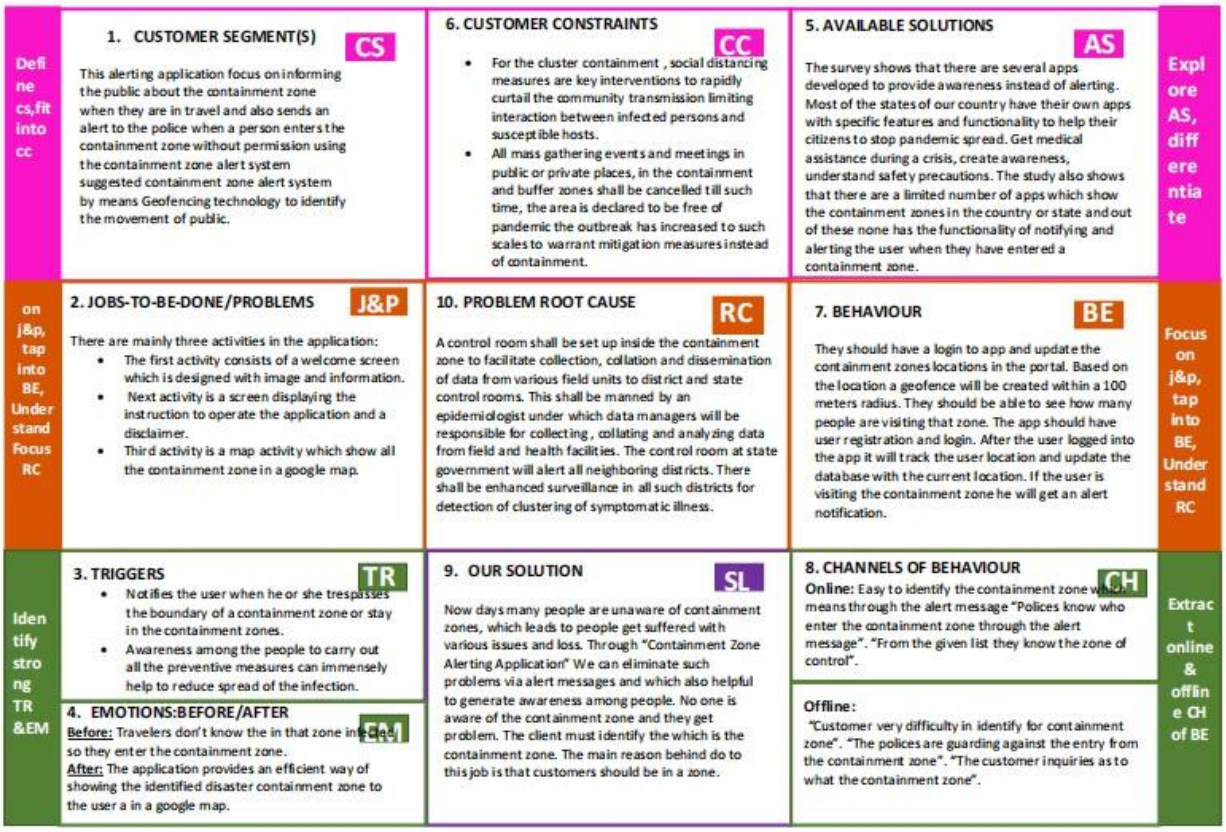
S.No.	Parameter	Description
1.	Problem Statement (Problem to be solved)	Now a days many people are unaware of containment zones, which leads to people get suffered with various issues and loss. Through "Containment Zone Alerting Application" We can eliminate such problems via alert messages and which also helpful to generate awareness among people.
2.	Idea / Solution description	(i).This application is intended to provide information about containment zones in a particular region by alerting people. (ii).Through continuous monitoring of an individual's location. (iii).Key benefits of the application are monitoring peoples activity and alerting them of their safety movements.
3.	Novelty / Uniqueness	Alert message
4.	Social Impact / Customer Satisfaction	Alerting the people on the containment zones which reduce the loss and varies issues and create awareness among people.
5.	Business Model (Revenue Model)	Government can utilize the application and alerting the people when they are entering to the containment zone.
6.	Scalability of the Solution	(i).A good rule of thumb is a Geofence radius of 150m or twice the size of your location, whichever is larger. (ii).A larger Geofence may be needed depending on the number of users, the size of your building and surrounding WiFi in your neighborhood.

3.4 Problem Solution fit

The Problem-Solution Fit simply means that you have found a problem with your customer and that the solution you have realized for it actually solves the customer's problem. It helps entrepreneurs, marketers and corporate innovators identify behavioral patterns and recognize what would work and why.

Purpose:

- Solve complex problems in a way that fits the state of your customers.
- Succeed faster and increase your solution adoption by tapping into existing mediums and channels of behavior.
- Sharpen your communication and marketing strategy with the right triggers and messaging.
- Increase touch-points with your company by finding the right problem-behavior fit and trust by solving frequent annoyances, or urgent or costly problems.
- Understand the existing situation in order to improve it for your target group.



4. REQUIREMENT ANALYSIS

4.1 Functional requirement

FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	**User Registration**	It can be registered by valid Email id or phone number.
FR-2	**Admin Registration**	It can be update all the containment zone.
FR-3	**User Confirmation**	Verification code can received by registering Email id or phone number.
FR-4	**Alert Message Via Notification**	By user access of location while entered in the alerted area the notification are send by GPS tracking system and push the grids through mail id.
FR-5	**Show Infected Zones**	Marked by Geo fencing.
FR-6	**Track Alternate Routes**	By Google map API or Google dependencies.

4.2 Non-Functional requirements

FR No.	Non-Functional Requirement	Description
NFR-1	**Usability**	GUI is easier to interact with.
NFR-2	**Security**	The data collected from the user will be stored securely.
NFR-3	**Reliability**	The user can trust the results and navigate safely.
NFR-4	**Performance**	Accurate results can be achieved due to real time location sharing.
NFR-5	**Availability**	Available if the network bandwidth of the user is of good range.
NFR-6	**Scalability**	The application can be used from anywhere and can also be implemented for both mobile and web apps for the user interact.

5. PROJECT DESIGN

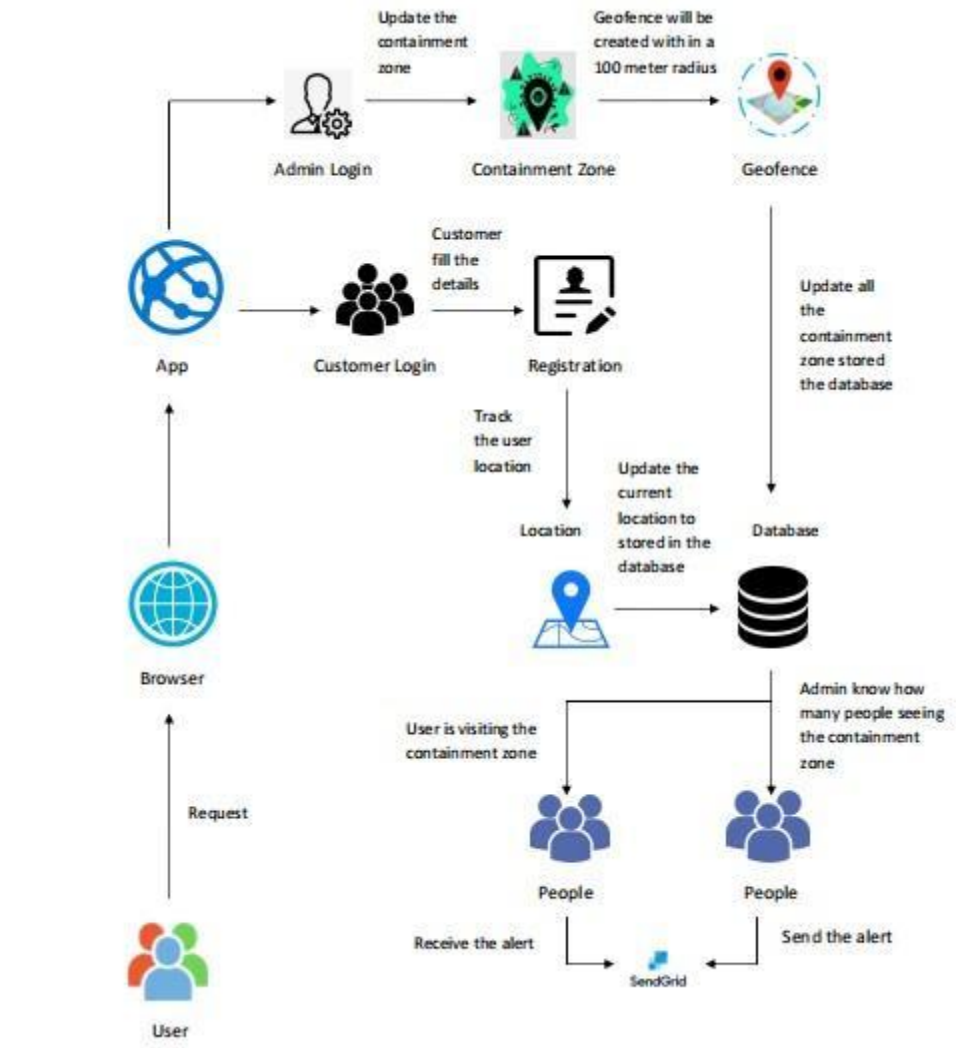
5.1 Data Flow Diagrams

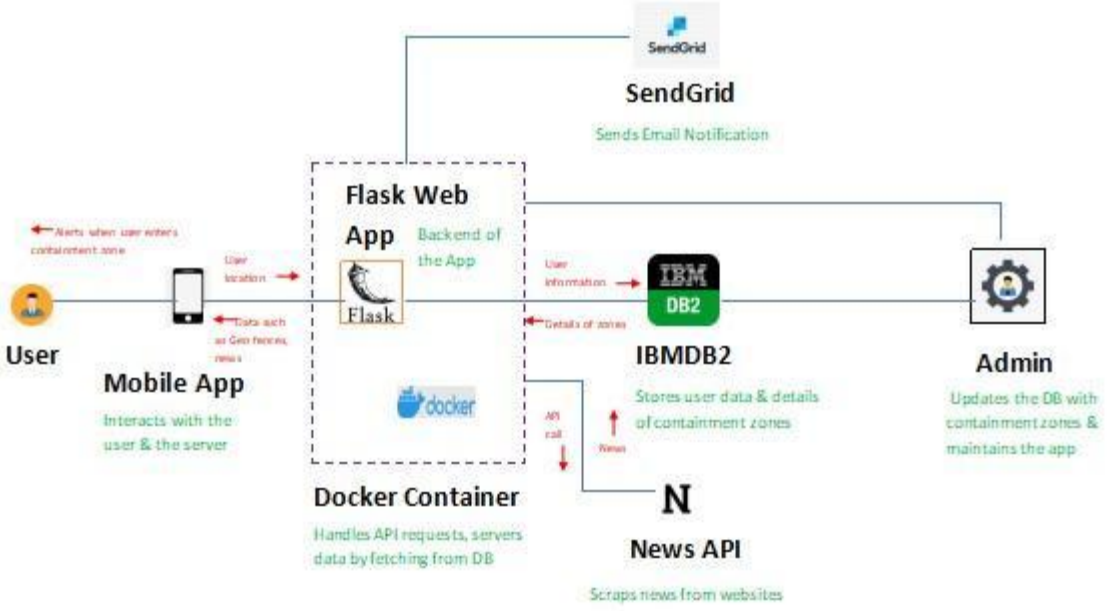
A Data Flow Diagram (DFD) is a traditional visual representation of the information flows within a system. A neat and clear DFD can depict the right amount of the system requirement graphically. It shows how data enters and leaves the system, what changes the information, and where data is stored.

5.2 Solution & Technical Architecture

Solution architecture is a complex process – with many sub-processes – that bridges the gap between business problems and technology solutions. Its goals are to:

- Find the best tech solution to solve existing business problems.
- Describe the structure, characteristics, behavior, and other aspects of the software to project stakeholders.
- Define features, development phases, and solution requirements.
- Provide specifications according to which the solution is defined, managed, and delivered.


Update the containment zone
Geofence will be created with in a 100 meter radius
Admin Login
Containment Zone
Geofence
Customer fill the details
Update all the containment zone stored the database
App
Customer Login
Registration
Track the user location
Update the current location to stored in the database
Location
Database
Browser
Admin know how many people seeing the containment zone
User is visiting the containment zone
Request
People
People
Receive the alert
Send the alert
SendGrid
User


SendGrid
SendGrid
Sends Email Notification
Flask Web App
Backend of the App
Flask
User
Mobile App
Interacts with the user & the server
IBM DB2
IBMDB2
Stores user data & details of containment zones
Admin
Updates the DB with containment zones & maintains the app
docker
Docker Container
Handles API requests, servers data by fetching from DB
N
News API
Scraps news from websites

5.3 User Stories

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
Customer (Mobile user)	Registration	USN-1	As a user, I can register for the application by entering my email, password, and confirming my password.	I can access my account / dashboard	High	Sprint-1
		USN-2	As a user, I will receive confirmation email once I have registered for the application	I can receive confirmation email & click confirm	High	Sprint-1
		USN-3	As a user, I can register for the application through Facebook	I can register & access the dashboard with Facebook Login	Low	Sprint-2
		USN-4	As a user, I can register for the application through Gmail		Medium	Sprint-1
	Login	USN-5	As a user, I can log into the application by entering email & password		High	Sprint-1
	Dashboard	USN-6	As a User , Can I manually plot the alerted zone for my convenience only.	It can be viewed in the user dashboard.	Low	Sprint - 2
Customer (Web user)	Registration	USN-1	As a user, I can register for the application by entering my email, password, and confirming my password.	User account activities can be viewed in dashboard.	High	Sprint - 2
			Confirmation code has been send through the registered mail id ,phone number or any other accounts.			
	Location Access	USN - 2	As a User , I can viewed into the page , if there is any condition to access the location.	Location can be turned through Control center.	High	Sprint - 2
	Containment Zones	USN - 3	Is it accurately show off the alerted zone If I entered into the zone the messages are properly received through email.	Alerted messages are send by send grids through the registered mail id.	High	Sprint - 3
Administrator	Frequent Updates	USN - 4	Admin are necessary to updates the recent containment through their portals and these seen through the app.	It can be accessed by Geo fencing.	Medium	Sprint - 4

6. PROJECT PLANNING & SCHEDULING

6.1 Sprint Planning & Estimation

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team members
Sprint-1	Registration	USN-1	User: I can register for the application by entering my email, password, and verifying password.	3	High	Sharmila Devi
Sprint-1	Registration	USN-2	User: I will receive confirmation email once I have registered for the application.	2	High	Shalini Rani
Sprint-1	Registration	USN-3	User: I can register for the application through Gmail.	5	Medium	Ranjith
Sprint-1	Registration	USN-4	Management: I need to register my hospitals on the site.	2	High	Manoj
Sprint-1	Login	USN-5	User: I can log into the application by entering my email & password.	3	High	Sharmila Devi
Sprint-1	Login	USN-6	Management: I need to login into my dashboard with my given hospital id and password.	5	Medium	Shalini Rani

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team members
Sprint-2	Dashboard	USN-7	User: I need to give permission to access my contacts, location, and storage.	5	High	Ranjith
Sprint-2	Dashboard	USN-8	User: I get access to the dashboard which shows a map with containment zones.	5	High	Manoj
Sprint-2	Dashboard	USN-9	Management: I need to enter the case information of the patient that visits our hospital.	5	High	Sharmila Devi
Sprint-2	Services	USN-10	Admin: I need to provide valid information about the pandemic out there.	5	High	Shalini Rani
Sprint-3	Dashboard	USN-11	Management: I need to store all the patient information on the cloud.	5	High	Ranjith
Sprint-3	Services	USN-12	Admin: I need to provide medical advice through a chat box.	5	Medium	Manoj
Sprint-3	Services	USN-13	Admin: I need to provide medical recommendations by collaborating with top hospitals.	5	Low	Sharmila Devi
Sprint-3	Services	USN-14	Admin: I need to provide preventive measures when they travel through it.	5	High	Shalini Rani
Sprint-4	Registration	USN-15	User: I can register for the application through Face book.	2	Low	Ranjith
Sprint-4	Registration	USN-16	User: I can register for the application through Twitter.	2	Low	Manoj
Sprint-4	Services	USN-17	Admin: I need to alert the user when they enter	3	Medium	Sharmila Devi

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team members
			pandemic zones.			
Sprint-4	Services	USN-18	Admin: I need to provide special services for premium by giving services like monitoring health by their smart bands.	3	Low	Shalini Rani
Sprint-4	Data Collection	USN-19	Admin: I need to store all the user information on the cloud.	5	Medium	Ranjith
Sprint-4	Data Collection	USN-20	Admin: I need to collect the recent list of diseases in the world.	5	Low	Manoj

6.2 Sprint Delivery Schedule

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	6 Days	24 Oct 2022	29 Oct 2022	20	29 Oct 2022
Sprint-2	20	6 Days	31 Oct 2022	05 Nov 2022	20	05 Nov 2022
Sprint-3	20	6 Days	07 Nov 2022	12 Nov 2022	20	12 Nov 2022
Sprint-4	20	6 Days	14 Nov 2022	19 Nov 2022	20	19 Nov 2022

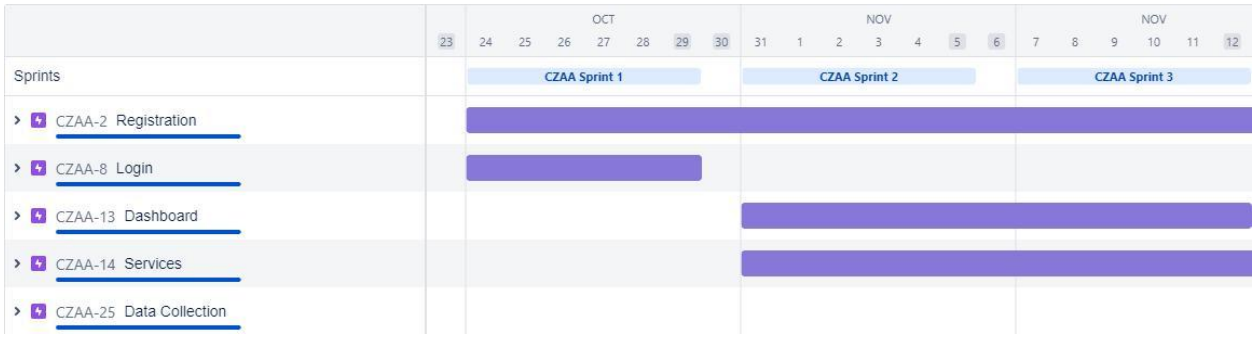
Velocity:

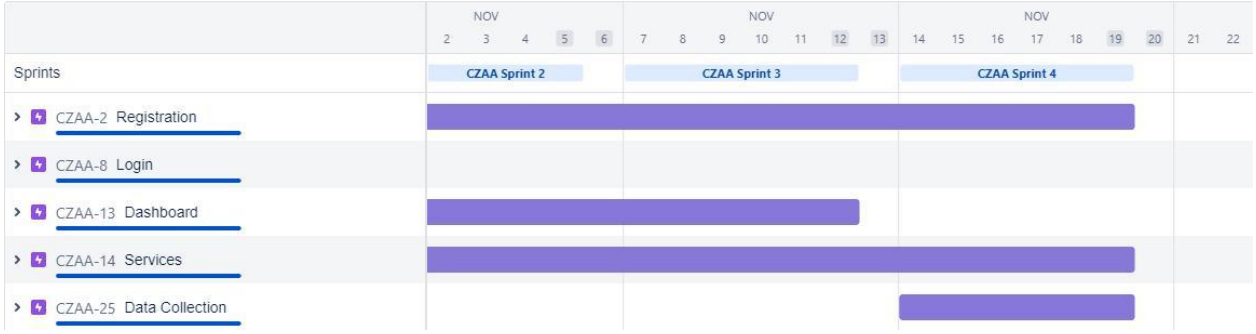
Imagine we have a 10-day sprint duration, and the velocity of the team is 20 (points per sprint). Let's calculate the team's average velocity (AV) per iteration unit (story points per day)

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

Burndown Chart:

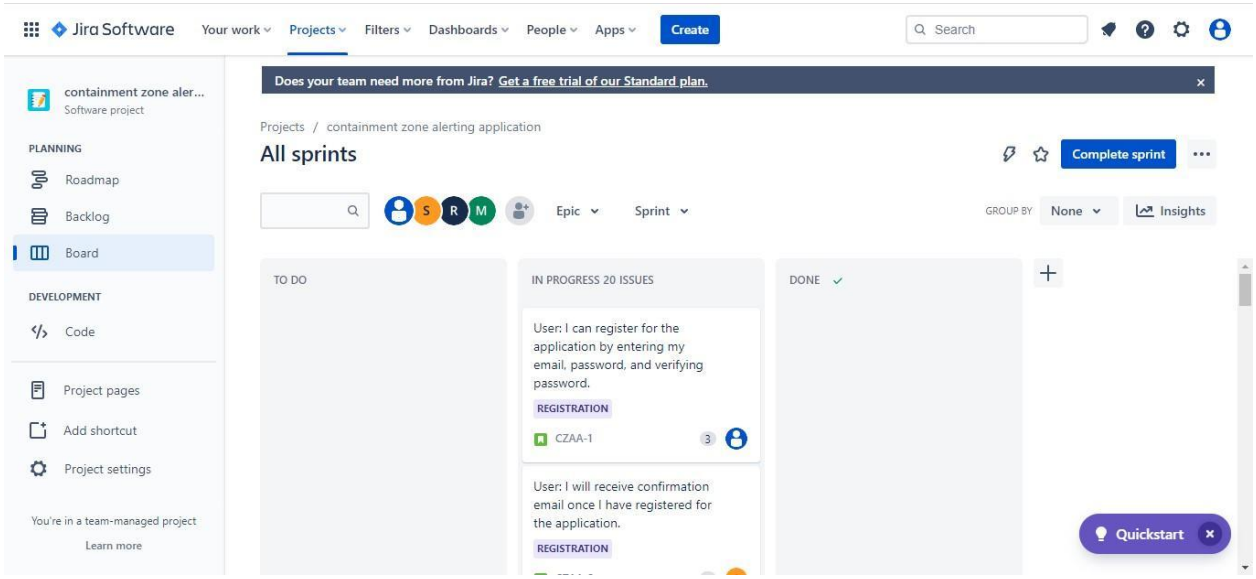
A burn down chart is a graphical representation of work left to do versus time. It is often used in agile software development methodologies such as Scrum. However, burn down charts can be applied to any project containing measurable progress over time.

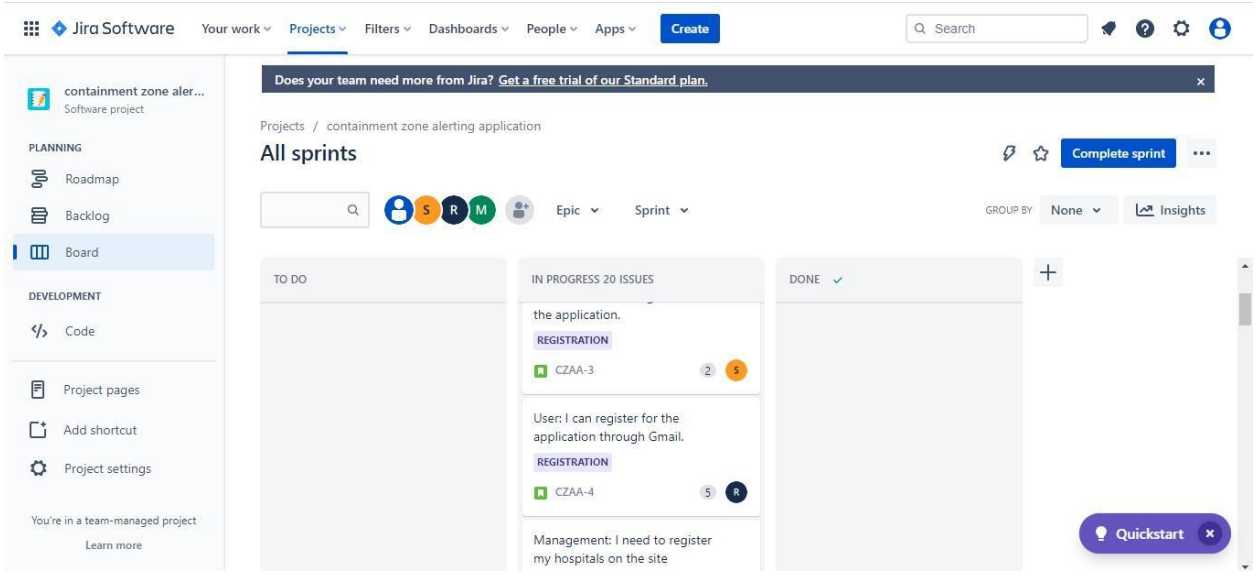


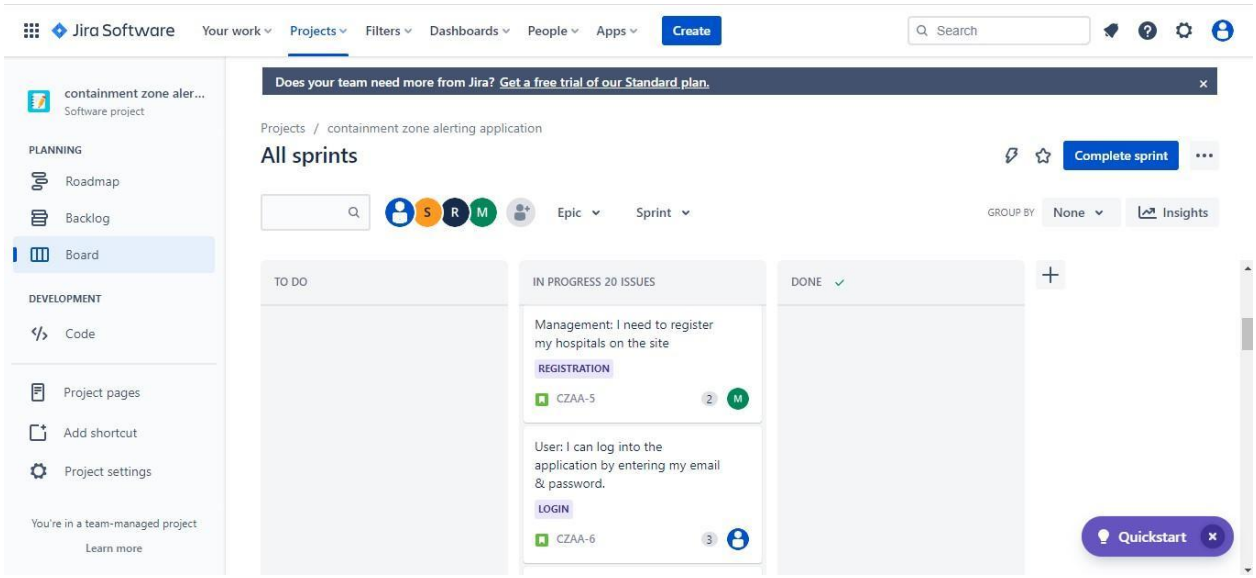


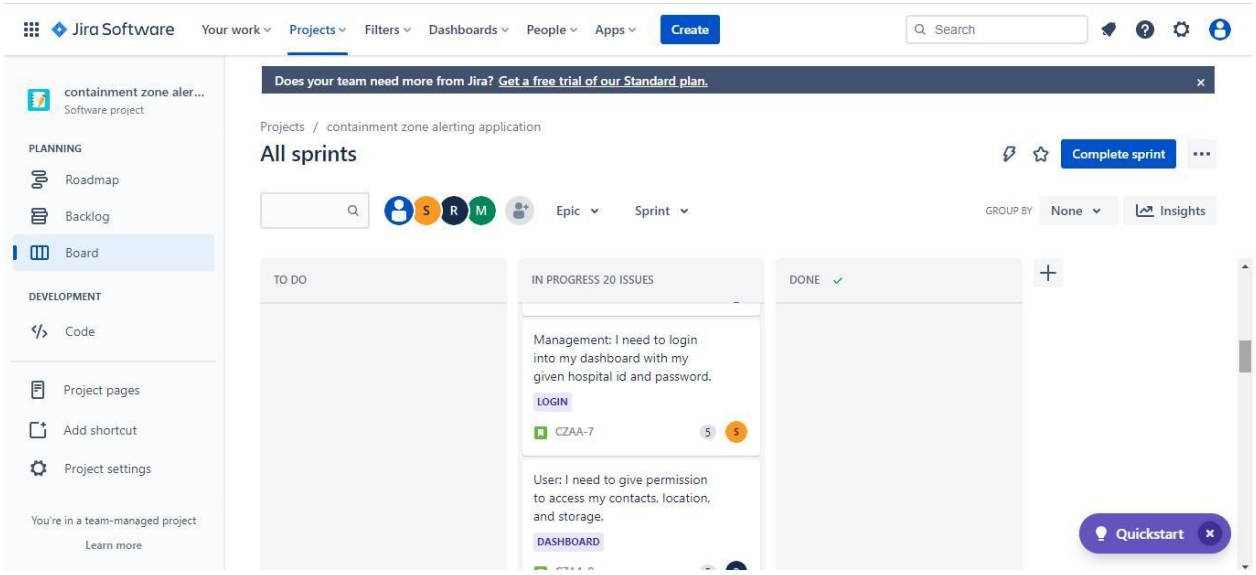
6.3 Reports from JIRA

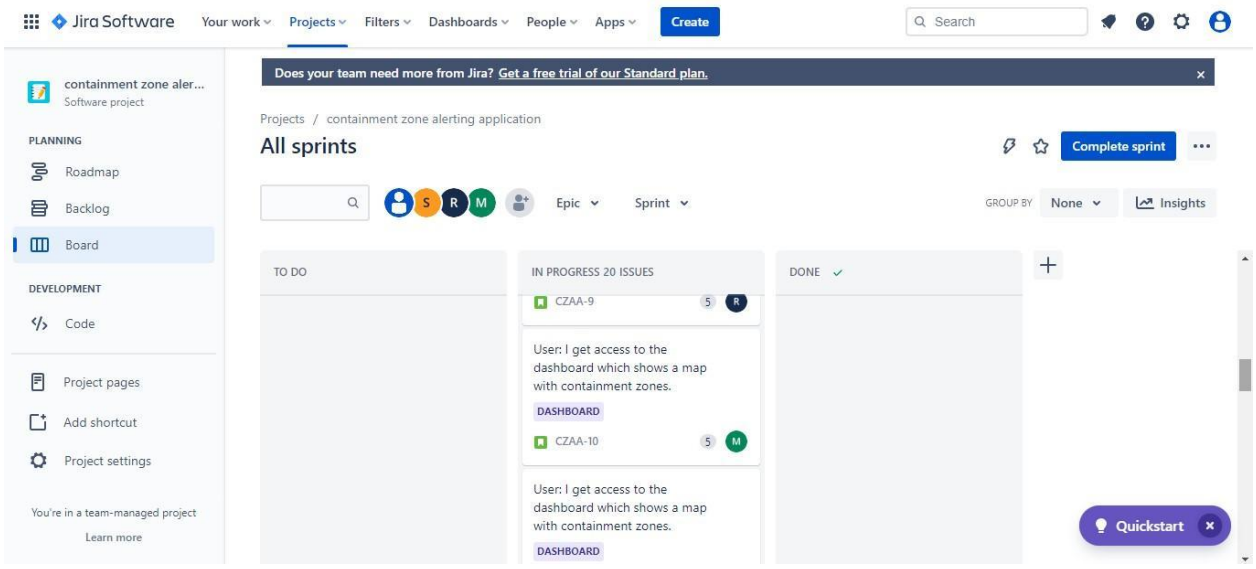
Kanban Board



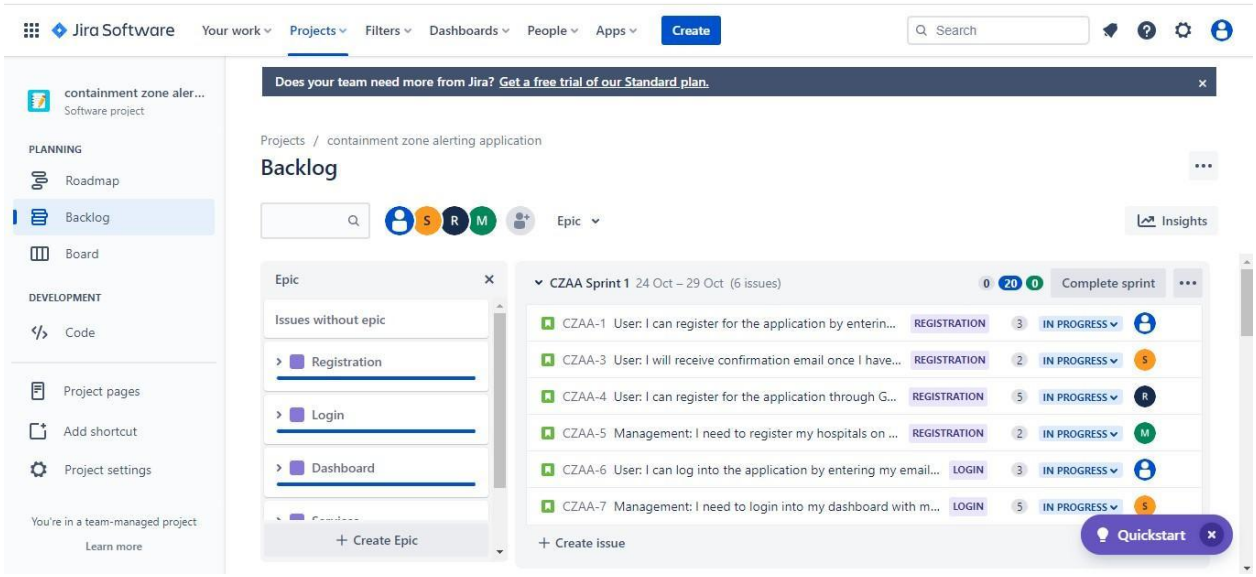


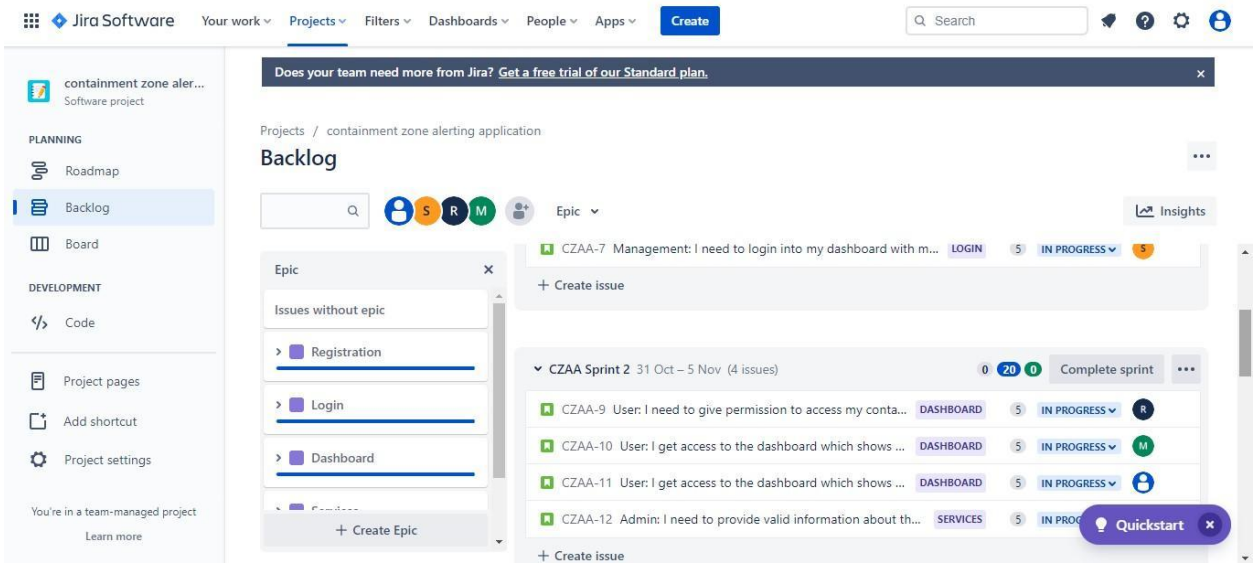

Jira Software
Your work
Projects
Filters
Dashboards
People
Apps
Create
Search
containment zone aler...
Software project
PLANNING
Roadmap
Backlog
Board
DEVELOPMENT
Code
Project pages
Add shortcut
Project settings
You're in a team-managed project
Learn more
Does your team need more from Jira? Get a free trial of our Standard plan.
Projects / containment zone alerting application
All sprints
Complete sprint
Epic
Sprint
GROUP BY
None
Insights
TO DO
IN PROGRESS 20 ISSUES
DONE
Management: I need to register my hospitals on the site
REGISTRATION
CZAA-5
User: I can log into the application by entering my email & password.
LOGIN
CZAA-6
Quickstart

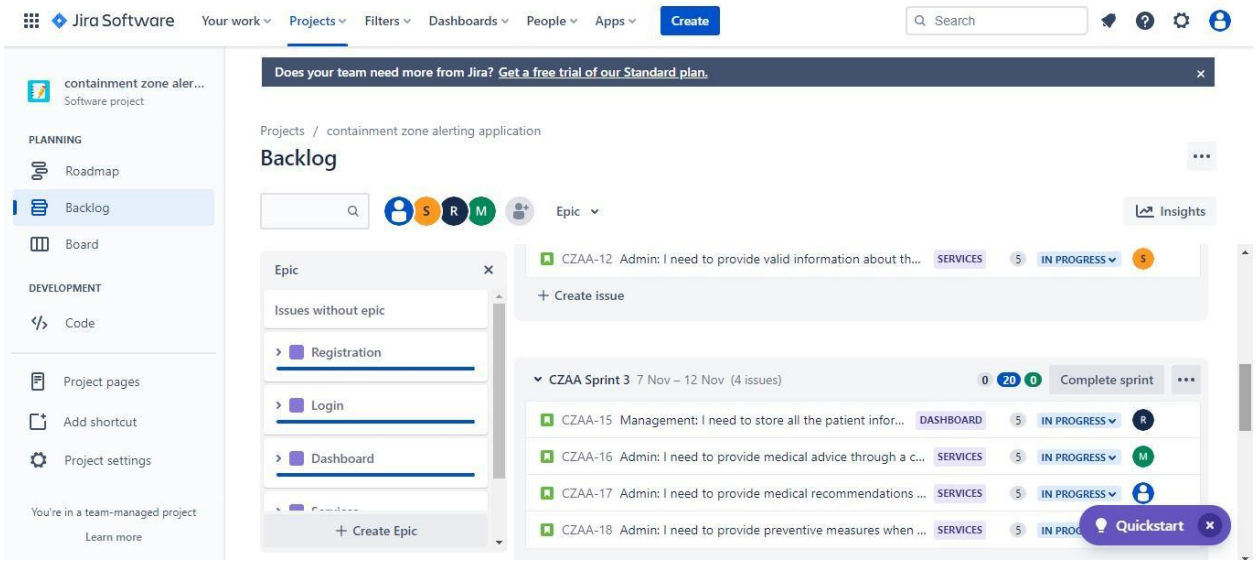

Jira Software
Your work
Projects
Filters
Dashboards
People
Apps
Create
Search
containment zone aler...
Software project
PLANNING
Roadmap
Backlog
Board
DEVELOPMENT
Code
Project pages
Add shortcut
Project settings
You're in a team-managed project
Learn more
Does your team need more from Jira? Get a free trial of our Standard plan.
Projects / containment zone alerting application
All sprints
Complete sprint
Epic
Sprint
GROUP BY
None
Insights
TO DO
IN PROGRESS 20 ISSUES
DONE
Management: I need to login into my dashboard with my given hospital id and password.
LOGIN
CZAA-7
User: I need to give permission to access my contacts, location, and storage.
DASHBOARD
Quickstart

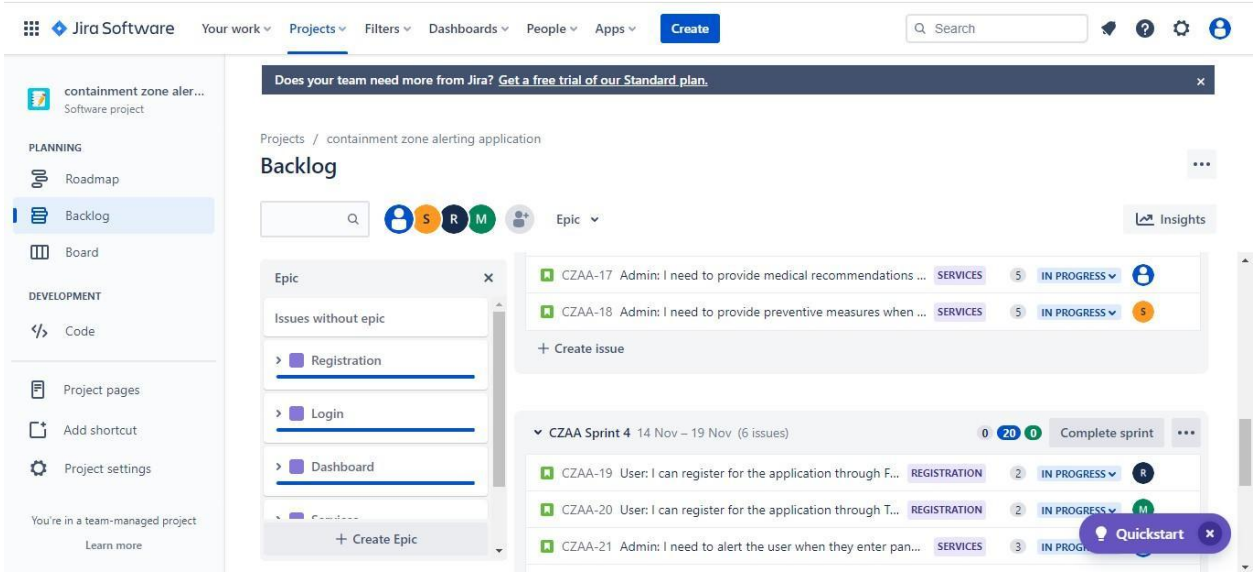


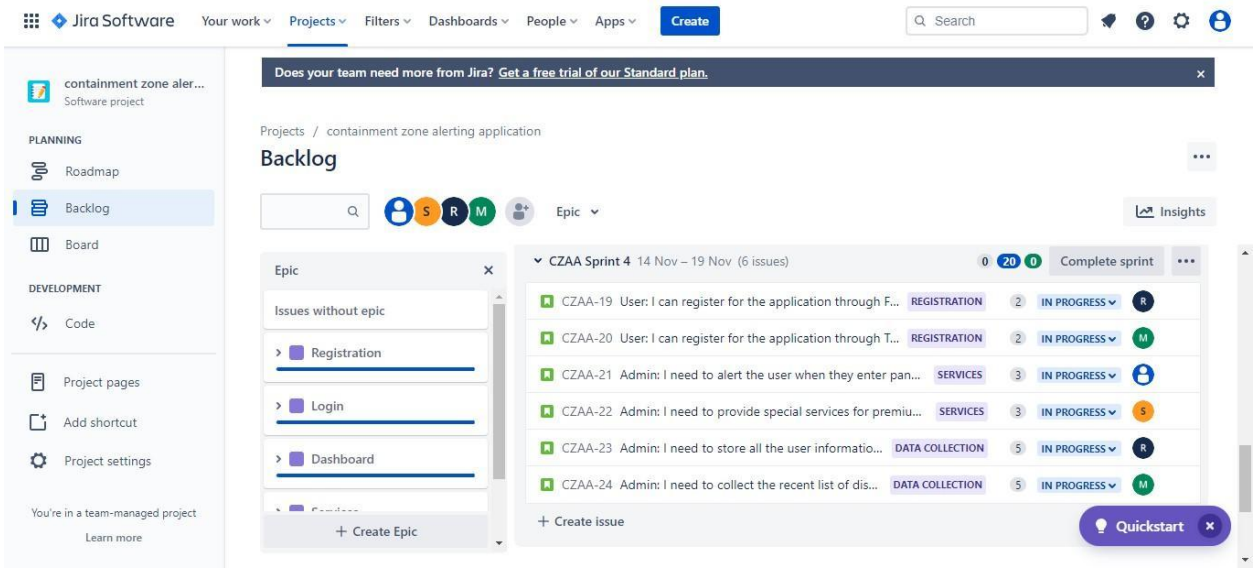
Project Backlog




Jira Software
Your work
Projects
Filters
Dashboards
People
Apps
Create
Search
containment zone aler...
Software project
PLANNING
Roadmap
Backlog
Board
DEVELOPMENT
Code
Project pages
Add shortcut
Project settings
You're in a team-managed project
Learn more
Does your team need more from Jira? Get a free trial of our Standard plan.
Projects / containment zone alerting application
Backlog
Epic
Insights
Epic
Issues without epic
Registration
Login
Dashboard
+ Create Epic
CZAA-7 Management: I need to login into my dashboard with m... LOGIN 5 IN PROGRESS
+ Create issue
CZAA Sprint 2 31 Oct – 5 Nov (4 issues)
0 20 0
Complete sprint
CZAA-9 User: I need to give permission to access my conta... DASHBOARD 5 IN PROGRESS
CZAA-10 User: I get access to the dashboard which shows ... DASHBOARD 5 IN PROGRESS
CZAA-11 User: I get access to the dashboard which shows ... DASHBOARD 5 IN PROGRESS
CZAA-12 Admin: I need to provide valid information about th... SERVICES 5 IN PROG
Quickstart
+ Create issue

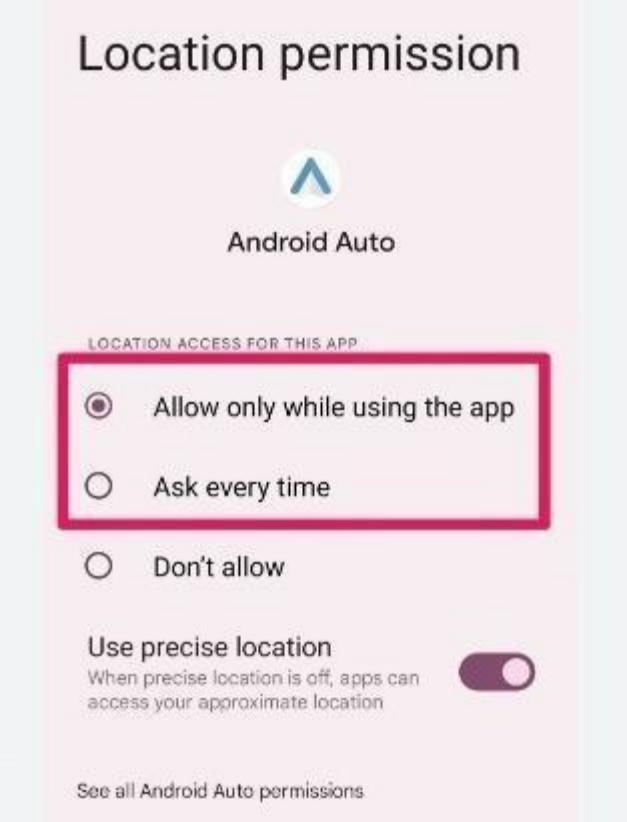

Jira Software
Your work
Projects
Filters
Dashboards
People
Apps
Create
Search
containment zone aler...
Software project
PLANNING
Roadmap
Backlog
Board
DEVELOPMENT
Code
Project pages
Add shortcut
Project settings
You're in a team-managed project
Learn more
Does your team need more from Jira? Get a free trial of our Standard plan.
Projects / containment zone alerting application
Backlog
Epic
Insights
Epic
Issues without epic
Registration
Login
Dashboard
+ Create Epic
CZAA-12 Admin: I need to provide valid information about th... SERVICES 5 IN PROGRESS
+ Create issue
CZAA Sprint 3 7 Nov – 12 Nov (4 issues)
0 20 0
Complete sprint
CZAA-15 Management: I need to store all the patient infor... DASHBOARD 5 IN PROGRESS
CZAA-16 Admin: I need to provide medical advice through a c... SERVICES 5 IN PROGRESS
CZAA-17 Admin: I need to provide medical recommendations ... SERVICES 5 IN PROGRESS
CZAA-18 Admin: I need to provide preventive measures when ... SERVICES 5 IN PROG
Quickstart

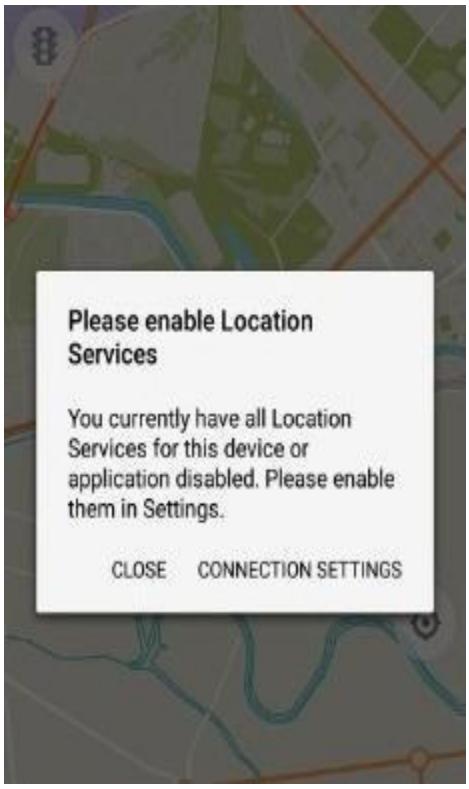


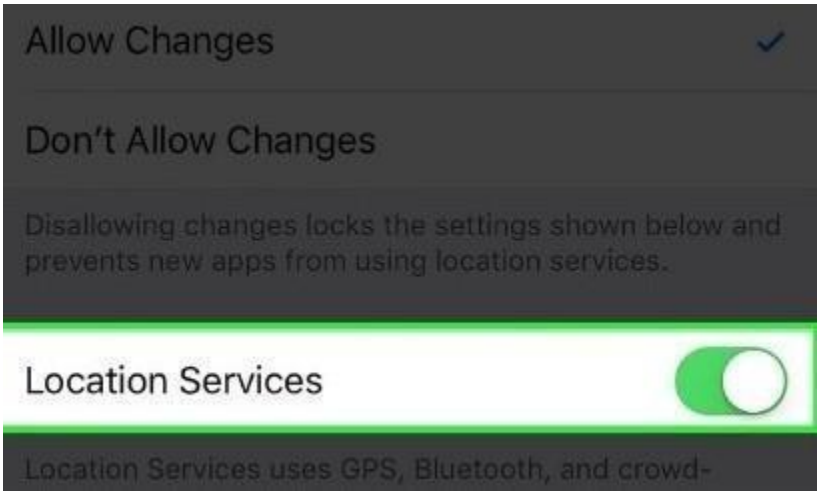


7. CODING & SOLUTIONING (Explain the features added in the project along with code)

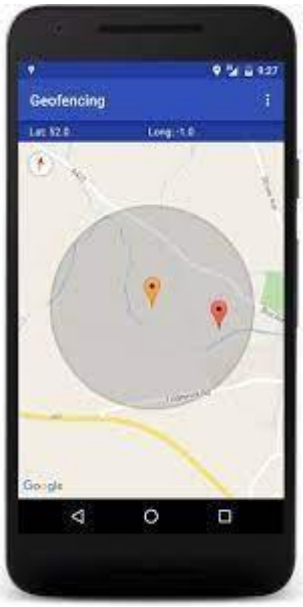
7.1 Feature 1


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-

7.2 Feature 2



7.3 Database Schema (if Applicable)


Remind me there
Park
Plaza General San Martín
Teatro Colseo
Pizza Restaurant
El Cuatro
Shopping Mall
Galerías Pacífico
Civil Central
Buenos Aires
Uruguay
Perú
Choose the radius
600 meters
CONTINUE


ENTER
California Ave
Caltrain Station
DWELL
EXIT
GEOFENCE PERIMETER

8. TESTING

8.1 Test Cases

Test case ID	Feature Type	Component	Test Scenario	Steps To Execute	Test Data	Expected Result	Actual Result	Status	Comments	TC for Automation(Y/N)	BUG ID	Executed By
LoginPage_TC_001	Functional	Home Page	Verify user is able to see the Login/ Signup popup when user clicked on My account button	1.Enter URL and click go 2.Scroll down 3.Verify login/Signup popup displayed or not	http://169.51.204.215:30106/	Login/ Signup popup should display	Working as expected	PASS	Successful			Sharmila Devi
LoginPage_TC_002	UI	Home Page	Verify the UI elements in Login/ Signup popup	1. Enter URL and click go 2.Click on Signup button for User 3. Verify login/ Signup popup with below UI elements: a.id text box b.password text box c.Login button d.New customer? Create account link e.Last password? Recovery password link	http://169.51.204.215:30106/	Application should show below UI elements: a.email text box b.password text box c.Login button with orange colourd.New customer? e.Last password? Create accou	Working as expected	PASS	Successful			Shalini Rani

						nt link Recovery password Link						
LoginPage_TC_003	Functional	Home page	Verify user is able to log into application with Valid credentials	1.Enter URL(https://shopeneer.com /) and click go 2.Click on My Account dropdown button 3.Enter Valid ID in ID text box 4.Enter valid password in password text box 5.Click on login Button	ID: 5342 password: Testing123	User should navigate to user account home page	Working as expected	PASS	Successful			Ranjith
LoginPage_TC_004	Functional	Login page	Verify user is able to log into application with In Valid credentials	1.Enter URL(http://169.51.204.215 :30106/) and click go 2.Click on My Account dropdown button 3.Enter In Valid ID in ID text box 4.Enter valid password in password text box 5.Click on login button	ID: 5342 password: Testing123	Application should show 'Incorrect email or password' validation message.	Working as expected	PASS	Successful			Manoj
LoginPage_TC_005	Functional	Login page	Verify user is able to log into application with In Valid credentials	1.Enter URL(http://169.51.204.215 :30106/) and click go 2.Click on My Account dropdown button 3.Enter In Valid ID in ID text box 4.Enter valid password in password text box 5.Click on login button	ID: 5342 password: Testing123	Application should show 'Incorrect email or password' validation message.	Working as expected	PASS	Successful			Sharmila Devi
LoginPage_TC_006	Functional	Login page	Verify user is able to log into application with In Valid credentials	1.Enter URL(http://169.51.204.215 :30106/) and click go 2.Click on My Account dropdown button 3.Enter In Valid ID in ID text box 4.Enter valid password in password text box 5.Click on login button	ID: 5342 password: Testing123	Application should show 'Incorrect email or password' validation message.	Working as expected	PASS	Successful			Shalini Rani

LoginPage_TC_007	Functional	Login page	Verify user is able to log into application with In Valid credentials	1.Enter URL(http://169.51.204.215:30106/) and click go 2.Click on My Account dropdown button 3.Enter In Valid ID in ID text box 4.Enter valid password in password text box 5.Click on login button	ID: 5342 password: Testing123	Application should show 'Incorrect email or password' validation message.	Working as expected	PASS	Successful			Ranjith
LoginPage_TC_008	Functional	Login page for Admin	Verify User is able to log into application with Valid Credentials	1.Enter URL(http://169.51.204.215:30106/) and click go 2.Click on My Account dropdown button 3.Enter Valid ID in ID text box 4.Enter valid password in password text box 5.Click on login button	ID: 1111 password: 5678	Application should show 'correct email or password ' validation message.	Working as expected	PASS	Successful			Manoj
LoginPage_TC_009	UI	Admin Page	Verify all the Customer database is visible	1.Enter URL(http://169.51.204.215:30106/) and click go 2.Click on My Account dropdown button 3.Enter Invalid ID in ID text box 4.Enter Invalid password in password text box 5.Click on login button	https://169.51.204.215:30106/	Customer database is visible	Working as expected	PASS	Successful			Sharmila Devi
LoginPage_TC_010	Functional	User Register	Verify Id sent to customer email address	1.Enter URL(http://169.51.204.215:30106/) and click go 1.Register the account by giving credentials 2.Click on button Submit	https://169.51.204.215:30106/	Email sent successfully	Working as expected	PASS	Successful			Shalini Rani
LoginPage_TC_011	Functional	Agent Register	Verify AGENT is able to log into application with Valid Credentials	1.Enter URL(http://169.51.204.215:30106/) and click go 2.Click on My Account dropdown button 3.Enter Invalid ID in ID text box 4.Enter Invalid password in password text box 5.Click on login	ID: 5342 password: Testing123	ID sent successfully	Application should show 'correct email or password' validation	PASS	Successful			Ranjith

				button			message					
LoginPage_TC_012	Functional	Login page for Admin	Verify User is able to log into application with Invalid Credentials	1.Enter URL(http://169.51.204.215:30106/) and click go 2.Click on My Account dropdown button 3.Enter In Valid ID in ID text box 4.Enter Invalid password in password text box 5.Click on login button	ID: 1111 password: 5678	Application should show 'Incorrect ID or password ' validation message.	Working as expected	PASS	Sucessful			Manoj
LoginPage_TC_013	UI	Home page for Agent	Verify user is able to see the agent home page when user finish on submitting Credentials	1.Enter URL(http://169.51.204.215:30106/) and click go 2.To the Agent Login page and submit Your Credentials	ID: 1111 password: 5678	AGENT Home Page popup should display	Working as expected	PASS	Sucessful			Sharmila Devi
LoginPage_TC_014	UI	Home page for User	Verify user is able to see the User home page when user finish on submitting Credentials	1 Enter URL(http://169.51.204.215:30106/) and click go 2 To the User Login page and submit Your Credentials	http://169.51.204.215:30106/	USER Home Page popup should display	Working as expected	PASS	Sucessful			Shalini Rani
LoginPage_TC_015	UI	Home page for Admin	Verify user is able to see the ADMIN home page when user finish on submitting Credentials	1.Enter URL(http://169.51.204.215:30106/) and click go 2. To the User Login page and submit Your Credentials	http://169.51.204.215:30106/	ADMIN Home Page popup should display	Working as expected	PASS	Sucessful			Ranjith
LoginPage_TC_016	Functional	Agent Page	On delete	1.Enter URL(http://169.51.2	http://169.51.20	Admin Home	Working as	PASS	Sucessful			Manoj

			Button the user Credentials will be deleted	04.215:30106/) and click go 2 To the Admin Page and delete an User Credentials	4.215:30106/	Page popup should display	expected					

8.2 User Acceptance Testing

1. Purpose of Document

The purpose of this document is to briefly explain the test coverage and open issues of the [Containment Zone Alerting Application] project at the time of the release to User Acceptance Testing (UAT).

2. Defect Analysis

This report shows the number of resolved or closed bugs at each severity level, and how they were resolved.

Resolution	Severity1	Severity2	Severity3	Severity4	Subtotal
By Design	10	3	1	3	17
Duplicate	1	0	3	0	4
External	2	3	0	1	6
Fixed	11	2	4	20	40
Not Reproduced	0	0	1	0	1
Skipped	0	0	1	1	2
Won'tFix	0	5	2	1	8
Totals	24	13	12	25	78

3. Test Case Analysis

This report shows the number of test cases that have passed, failed, and untested.

Section	Total Cases	Not Tested	Fail	Pass
Print Engine	10	0	0	10
Client Application	50	0	0	50
Security	2	0	0	2
Outsource Shipping	3	0	0	3
Exception Reporting	8	0	0	8
Final Report Output	4	0	0	4
Version Control	2	0	0	2

9. RESULTS

9.1 Performance Metrics

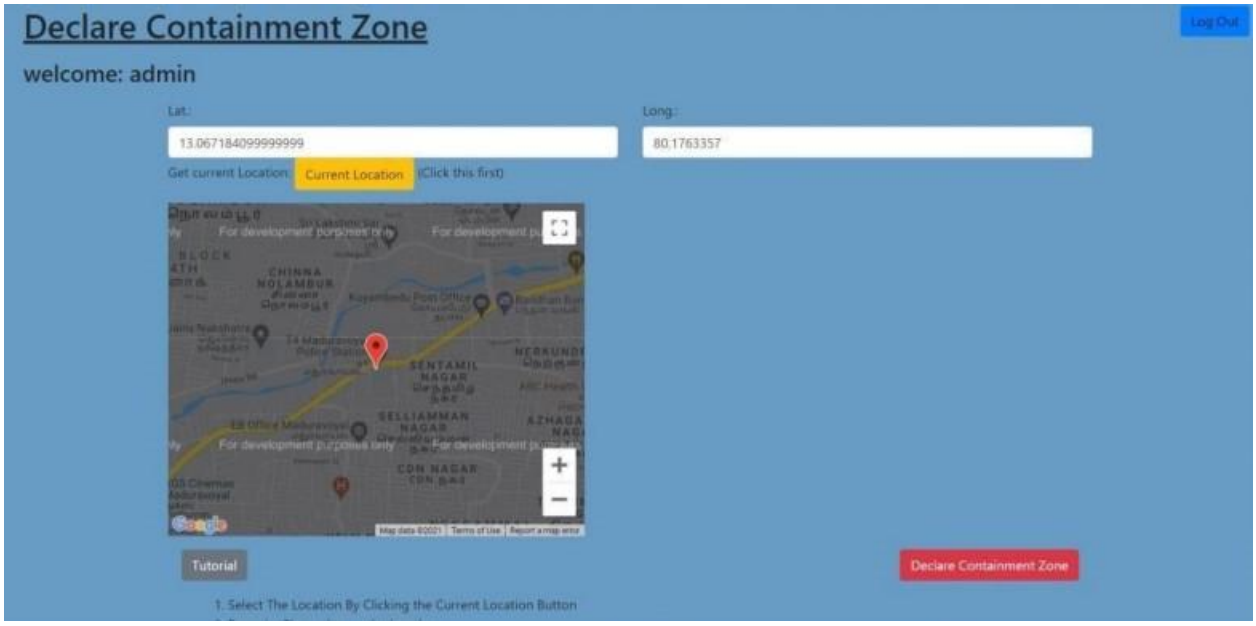
UI Interact with Application:

Admin App:

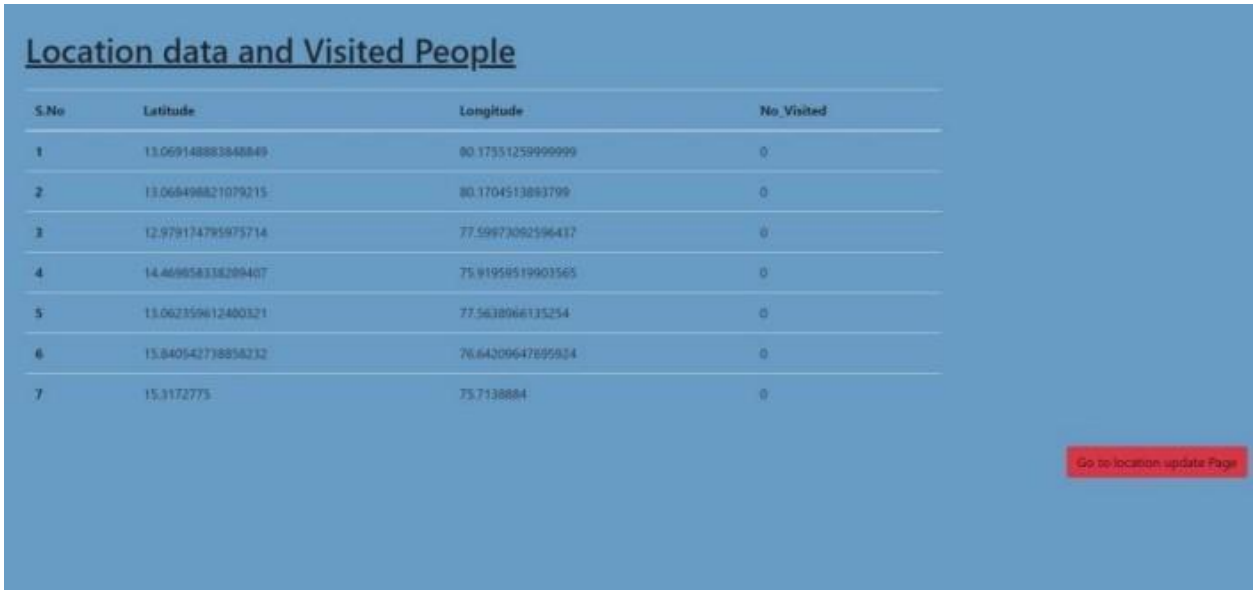
Login Page:

Register page:

Home page:



Location data page:



Client Application:

Register screen:

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.

10. 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.

11. CONCLUSION

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

12. FUTURE SCOPE

Although 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 analyze the data obtained through this application. Time series analysis is used to predict crop production as well as sales in different quarter.

13. APPENDIX

Source Code

```
# Project: CONTAINMENT ZONE ALERTING APPLICATION
# Team ID: PNT2022TMID48366
```

APP.PY

```
from logging import error
from flask import *
from jinja2.utils import select_autoescape
import bcrypt
from flask_mysqldb import MySQL

import json
from sendgrid import SendGridAPIClient
from sendgrid.helpers.mail import Mail
# initialization
app = Flask(__name__)
# config
app.secret_key = "\x19Ts\xbe\xe7\x8c_\r\x12Q\x14\x13>q\xb7'WTH0\x9f\xe4\xec\xb1"
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='varundutia.h@gmail.com',
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.7BJDtQDlS8unH0r5_TufVQ.Ykpcz19QcqgcNwYZC3a0mNRPhGksG117YURqOTa2HL')
response = sg.send(message)
print(response.status.code)
print(response.body)
print(response.headers)
except Exception as e:
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()
return password_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)
return does_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:
return render_template('login.html', error=2)
return render_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()
return render_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()
return redirect(url_for('login'))
return render_template('signup.html', error=False)

@app.route("/home", methods=["POST", "GET"])
def home():
if(session['id'] == None):
return redirect(url_for('login'))
if(request.method == "POST"):
# get data
lat = request.form["lat"]
lon = request.form["lon"]
vis = 0
if(lat == "" or lon == ""):

return render_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()
return render_template('home.html', name=session['name'], email=session['email'],
id=session['id'], success=True)
return render_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
return redirect(url_for('login'))
@app.route("/data")
def data():
if(session['id'] == None):
return redirect(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):
return render_template("data.html", responses=0)
else:
res = location_cursor.fetchall()
print(res)
return render_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 x in signup_cursor.description]
json_data = []
for result in rv:
json_data.append(dict(zip(row_headers, result)))
return json.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(
"SELECT * FROM LOCATION"
)
if(user_result != 0):
res = location_cursor.fetchall()
print(res)
row_headers = [x[0] for x in location_cursor.description]
json_data = []
```

```
for result in res:
json_data.append(dict(zip(row_headers, result)))
return json.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

```
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8">
<meta http-equiv="X-UA-Compatible" content="IE=edge">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>Zones</title>
<link rel="style sheet"
href="https://stackpath.bootstrapcdn.com/bootstrap/4.4.1/css/bootstrap.min.css"
integrity="sha384-
Vkoo8x4CGsO3+Hhxv8T/Q5PaXtkKtu6ug5TOeNV6gBiFeWPGFN9MuhOf23Q9Ifjh"
```

```
crossorigin="anonymous" />
<style>
body {
padding-top: 30px;
padding-bottom: 30px;
background-color: #699cc5;
}
a {
color: black;
}
</style>
</head>
<body>
<div class="m-4 container">
div>
<div class="m-4 container">
<table class="table">
<thead>
<tr>
<th scope="col">S.No</th>
<th scope="col">Latitude</th>
<th scope="col">Longitude</th>
<th scope="col">No_Visited</th>
</tr>
</thead>
<tbody>
{%- for row in responses %}
<tr>
<th scope="row">{{loop.index}}</th>
<td>{{row[1]}}</td>
<td>{{row[2]}}</td>
<td>{{row[3]}}</td>
</tr>
{%- endfor %}
</tbody>
</table>
</div>
<div class="m-3 float-right">
<button type="button" class="btn btn-danger"><a href={{url_for("home")}}>Go to location
update Page</a></button>
</div>
</body>
</html>
```

HOME.HTML

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

```
<head>
<meta charset="UTF-8">
<meta http-equiv="X-UA-Compatible" content="IE=edge">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>Document</title>
<link rel="style sheet"
href="https://stackpath.bootstrapcdn.com/bootstrap/4.4.1/css/bootstrap.min.css"
integrity="sha384-
Vkoo8x4CGsO3+Hhxv8T/Q5PaXtkKtu6ug5TOeNV6gBiFeWPGFN9MuhOf23Q9Ifjh"
crossorigin="anonymous" />
<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 %}
<div class="m-3 float-right">
<button type="button" class="btn btn-primary"><a href={{url_for("logout")}}>Log
Out</a></button>
</div>
<div class="container m-3">
<h1><u>Declare Containment Zone</u></h1>
</div>
<div class="container m-3">
<h3>welcome: {{name}}</h3>
</div>
<form method="POST" action="/home">
<div class="container">
<div class="form-group row">
<div class="col-sm-6">
<label class="control-label">Lat.:</label>
<input type="text" class="form-control" id="lat" name="lat" />
</div>
<div class="col-sm-6">
<label>Long.:</label>
```

```
<input type="text" class="form-control" id="lon" name="lon" />
</div>
<div class="col-sm-6">
<label>Get current Location:</label>
<button type="button" class="btn btn-warning" onclick="getLocation()">Current
Location</button>
<label>(Click this first)</label>
</div>
</div>
<!-- map -->
<div id="map_disp" style="height: 400px;width: 500px;"></div>
<div class="m-3 float-right">
<button type="submit" class="btn btn-danger">Declare Containment Zone</button>
</div>
<div class="m-3">
<button onclick="toggleTips()" type="button" class="btn btn
secondary">Tutorial</button>
<div id="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>
<div class="m-3 float-right">
<button type="button" class="btn btn-warning"><a href="{{url_for('data')}}">Click Here To
View The
Containment Zones and Number of
people visited</a></button>
</div>
</div>
<script src="https://cdn.jsdelivr.net/npm/bootstrap@4.6.0/dist/js/bootstrap.min.js"
integrity="sha384-
+YQ4JLhjyBLPDQt//I+STsc9iw4uQqACwlvpslubQzn4u2UU2UFM80nGisd026JF"
crossorigin="anonymous"></script>
<script src="https://code.jquery.com/jquery-2.2.4.min.js"></script>
<script
src="https://maps.google.com/maps/api/js?sensor=false&libraries=places"></script>
<script
src="https://rawgit.com/Logicify/jquery-locationpicker
plugin/master/dist/locationpicker.jquery.js"></script>
<script>
function getLocation() {
if (navigator.geolocation) {
navigator.geolocation.getCurrentPosition(showPosition);
} else {
alert("No location");
}
}
```

```
function showPosition(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 + ")");
}
});
}
function toggleTips() {
var x = document.getElementById("tips");
if (x.style.display === "none") {
x.style.display = "block";
} else {
x.style.display = "none";
}
}
</script>
</body>
</html>
```

GitHub Link & Project Demo Link

GitHub Link:

https://github.com/IBM-EPBL/IBM-Project-38176-1660374197

Project Demo Link:

https://youtu.be/KVwyAqZCHHY