

# **NALAIYA THIRAN**

## **IBM**

### **PROJECT REPORT**

**Team ID:PNT2022TMIDE52614**

### **Containment Zone Alerting Application**

#### **Presented by**

Abhishek Sharma      CITC 1905064

Arul Karthikeyan M      CITC 1905068

Deepika T      CITC 1905073

Dividharshana K      CITC 1905078

# INDEX

- 1. INTRODUCTION**
  1. Project Overview
  2. Purpose
- 2. LITERATURE SURVEY**
  1. Existing problem
  2. References
  3. Problem Statement Definition
- 3. IDEATION & PROPOSED SOLUTION**
  1. Empathy Map Canvas
  2. Ideation & Brainstorming
  3. Proposed Solution
  4. Problem Solution fit
- 4. REQUIREMENT ANALYSIS**
  1. Functional requirement
  2. Non-Functional requirements
- 5. PROJECT DESIGN**
  1. Data Flow Diagrams
  2. Solution & Technical Architecture
  3. User Stories
- 6. PROJECT PLANNING & SCHEDULING**
  1. Sprint Planning & Estimation
  2. Sprint Delivery Schedule
  3. Reports from JIRA
- 7. CODING & SOLUTIONING**
  1. Feature 1
  2. Feature 2
- 8. TESTING**
  1. Test Cases
  2. User Acceptance Testing
- 9. RESULTS**
  1. Performance Metrics
- 10.ADVANTAGES & DISADVANTAGES**
- 11.CONCLUSION**
- 12.FUTURE SCOPE**
- 13.APPENDIX**

# **INTRODUCTION**

## **1.INTRODUCTION**

Currently there are several research works undergoing to prevent COVID-19. In this paper, we mainly focus on developing an android based application to identify the COVID-19 containment zone in India. We have used Geofencing and Cloud APIs (Application Program Interface) from Google as the base for development of this application. Geofencing is a tool from Android which is used to create virtual boundaries or fences around geographical locations. The developers can add geofences at different locations by providing the latitudes and longitudes along with radius to highlight the proximity of the location. Geofencing technology senses the user's current location and checks whether that location belongs to the set of pre-identified geofences.

After creating the geofences, each geofence has the permission to get information from the location services of Google about entering, dwelling and exit events triggered by the users of the application when they enter or exit the geofence. Upon receiving a trigger event, the geofences can be programmed to notify the user about the information of the location they have entered or exited. These geofences can be useful in preventing its users from trespassing inside the targeted locations.

In this research work we use both Geofencing and Cloud technology in our proposed Android application to efficiently identify and provide alarm to the ones who trespassed the containment zones of COVID-19.

### **1.1 PROJECT OVERVIEW:**

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 to their safety movements.

### **1.2 PURPOSE**

## **2.LITERATURE SURVEY**

### **2.1 COVID-19 Contact-tracing Apps: a Survey on the Global Deployment and Challenges.**

Jinfeng Li and Xinyi Guo

#### **DESCRIPTION:**

In this paper, to address the massive spike in uncertainties triggered by the coronavirus disease (COVID-19), there is an ever-increasing number of national governments that are rolling out contact-tracing Apps to aid the containment of the virus. The first hugely contentious issue facing the Apps is the deployment framework, i.e., centralized or decentralized. Based on this, the debate branches out to the corresponding technologies that underpin these architectures, i.e. GPS, QR codes, and Bluetooth. This work conducts a pioneering review of the above scenarios and contributes a geolocation mapping of the current deployment. The Apps' vulnerabilities and the directions of research are identified, with a special focus on the Bluetooth-inspired decentralized paradigm.

#### **ADVANTAGES:**

Use of underpinning technologies like GPS, QR codes, Bluetooth.

#### **DISADVANTAGES:**

Trade-off between the data privacy and the different mobile devices exhibit a variety of Bluetooth signal intensity at the ISM band.

### **2.2 Pandemic Contact Tracing Apps: DP-3T, PEPP-PT NTK, and ROBERT from a Privacy Perspective.**

Fraunhofer Aisec

#### **DESCRIPTION:**

In this paper, we review different approaches on proximity tracing apps which are supposed to automate the current labor-intensive contact tracing approach conducted by national health officials. The purpose of these apps is to automatically notify people who are at risk of being infected with SARS-CoV-2 to interrupt infection chains as early as possible. However, a privacy-preserving and yet functional and scalable design of such apps is not trivial and in some

parts leads to counter-intuitive properties. This paper reviews the most prominent European approaches, DP-3T, the German variant “NTK” of PEPP-PT, and its closely related concept ROBERT. We discuss their design decisions from a privacy perspective and point out the fundamentally different adversary models assumed by the approaches. In addition, we touch on practical aspects such as scalability and ease of implementation.

### **ADVANTAGES:**

Strong emphasis on resilience of the system, avoid de-anonymization of all user types.

### **DISADVANTAGES:**

Manual Contact Tracing, Functionality & Scalability, Security and privacy issues in the DP-3T system, DP-3T systems are vulnerable to Gossip Attack.

## **2.3 Towards Defeating Mass Surveillance and SARS-CoV-2: The Pronto-C2 Fully Decentralized Automatic Contact Tracing System**

Gennaro Avitabile, Vincenzo Botta, Vincenzo Iovino, and Ivan Visconti

### **DESCRIPTION:**

In this paper, Mass surveillance can be more easily achieved leveraging fear and desire of the population to feel protected while affected by devastating events. Indeed, in such scenarios, governments can adopt exceptional measures that limit civil rights, usually receiving large support from citizens. The COVID-19 pandemic is currently affecting daily life of many citizens in the world. People are forced to stay home for several weeks, unemployment rates quickly increase, uncertainty and sadness generate an impelling desire to join any government effort in order to stop as soon as possible the spread of the virus. Following recommendations of epidemiologists, governments are proposing the use of smartphone applications to allow automatic contact tracing of citizens. Such systems can be an effective way to defeat the spread of the SARS-CoV-2 virus since they allow to gain time in identifying potentially new infected persons that should therefore be in quarantine. This raises the natural question of whether this form of automatic contact tracing can be a subtle weapon for governments to violate privacy inside new and more sophisticated mass surveillance programs.

### **ADVANTAGES:**

Pronto-C2 relies on Diffie-Hellman key exchange providing better privacy.

## **DISADVANTAGES:**

Privacy Attacks for Mass Surveillance, Tracing Infected Users with Trusted Server and Colluding Server Matrix attack, Brutus attack, Gossip attack, Matteotti attack.

## **2.4 PACT: Privacy Sensitive Protocols and Mechanisms for Mobile Contact Tracing**

Justin Chan, Dean Foster, Shyam Gollakota, Eric Horvitz, Joseph Jaeger, Sham Kakade, Tadayoshi Kohno, John Langford, Jonathan Larson, Puneet Sharma, Sudheesh Singanamalla, Jacob Sunshine, Stefano Tessaro.

## **DESCRIPTION:**

The global health threat from COVID-19 has been controlled in a number of instances by large-scale testing and contact tracing efforts. We created this document to suggest three functionalities on how we might best harness computing technologies to supporting the goals of public health organizations in minimizing morbidity and mortality associated with the spread of COVID-19, while protecting the civil liberties of individuals. In particular, this work advocates for a third-party free approach to assisted mobile contact tracing, because such an approach mitigates the security and privacy risks of requiring a trusted third party. We also explicitly consider the inferential risks involved in any contract tracing system, where any alert to a user could itself give rise to de-anonymizing information.

More generally, we hope to participate in bringing together colleagues in industry, academia, and civil society to discuss and converge on ideas around a critical issue rising with attempts to mitigate the COVID-19 pandemic.

## **ADVANTAGES:**

This paper uses Third party free contact tracing, Confidentiality, Re-Identification.

## **DISADVANTAGES:**

Attacks like Inferential attacks, Integrity attacks, etc., and The mobile proximity tracing does not directly inform public authorities who may be a contact.

## **2.5 Trust and Transparency in Contact Tracing Applications**

Stacy Hobson, Michael Hind, Aleksandra Mojsilovic, Kush R. Varshney

## **DESCRIPTION:**

The global outbreak of COVID-19 has led to focus on efforts to manage and mitigate the continued spread of the disease. One of these efforts include the use of contact tracing to identify people who are at-risk of developing the disease through exposure to an infected person. Historically, contact tracing has been primarily manual but given the exponential spread of the virus that causes COVID-19, there has been significant interest in the development and use of digital contact tracing solutions to supplement the work of human contact tracers. The collection and use of sensitive personal details by these applications has led to a number of concerns by the stakeholder groups with a vested interest in these solutions. We explore digital contact tracing solutions in detail and propose the use of a transparent reporting mechanism, FactSheets, to provide transparency of and support trust in these applications. We also provide an example FactSheet template with questions that are specific to the contact tracing application domain.

## **ADVANTAGES:**

A broad Fact Sheet template to support transparency of contact tracing applications.

## **DISADVANTAGES:**

Privacy and coverage, Security and Access.

## **2.6 Tracking the Covid zones through geo-fencing technique**

Anto Arockia Rosaline R., Lalitha R., Hariharan G., Lokesh N.

## **DESCRIPTION:**

Because of the outbreak of Covid 19, the entire world is thinking of new strategies, preventive measures to safeguard human life from the widespread effects of the pandemic. The areas where people are affected are marked as containment zones and people are not allowed to exit out of those areas. Similarly, new people are not allowed to enter those areas. Hence, the purpose of this paper is to propose a methodology to track the Covid zones, to enhance and tighten the security measures. A geo-fence is created for the containment zone. The person who enters or exits out of that particular zone will be monitored and an alert message will be sent to that person's mobile.

## **ADVANTAGES:**

It reduces operational costs by using an automated system based on wireless infrastructure. It also alerts the authority immediately to catch the violators.

## **DISADVANTAGES:**

The workflow of the system demands the required data sets and permission in a legal manner to set up the environment that maintains the constitutional law and order in practice. Privacy concern is debatable.

## **2.7. The experience of contact tracing in Singapore in the control of COVID-19: highlighting the use of digital technology**

Sean Han Sheng Lai , Camelia Qian Ying Tang , Asok Kurup , Gowreeson Thevendran

### **DESCRIPTION:**

In this paper, containing the coronavirus disease 2019 (COVID-19) pandemic would require aggressive contact tracing and isolation of suspected or confirmed COVID-19 cases. Models in published literature have suggested that digital rather than manual contact tracing might be more effective in containing the pandemic. This article seeks to examine the forms of contact tracing that Singapore, a highly dense city-state, adopts with a focus on new innovations including the use of digital technology.

### **ADVANTAGES:**

Trace Together does not track location or contacts, and data is stored locally on the phone for 21 days and will not be accessed unless the individual has been identified as a close contact, and measures are in place to protect the individual's mobile number.

### **DISADVANTAGES:**

DCT is not ready to replace the manual and meticulous work, safeguarding data safety and patient privacy.

## **2.8. Privacy, Ethics, and Contact-tracing Apps**

Teresa Scassa, Jason Millar and Kelly Bronson

### **DESCRIPTION:**

Data and analytics are being enlisted to play a role in understanding and preventing the spread of COVID-19. This chapter focuses on digital “apps,” which are being deployed by governments around the world to supplement the manual contact-tracing efforts typically performed by public health officials. Contact-tracing apps have been developed rapidly, with little time for user



testing, and their adoption raises important privacy and ethical concerns. In this chapter, we outline some of these potential concerns. We begin by tracing the history of contact-tracing as a pre-digital, or manual, method and then detail the current contract-tracing efforts, distinguishing among different types of apps and data use approaches. We then draw from our complementary expertise in law, ethics, and sociology to outline potential risks of contract-tracing apps along these dimensions. Risks include misuse of personal data for surveillance and insufficient uptake leading to inaccurate information for individuals, which could lead to increased infection. Risks also include differential access and thus the reproduction of vulnerability among marginalized communities. Overall, the chapter identifies issues relevant to the responsible development and use of big data and AI for COVID-19 mitigation efforts.

### **ADVANTAGES:**

Centralized vs Decentralized data storage.

### **DISADVANTAGES:**

Insufficient testing, test results may be substantially delayed.

## **2.9. BlueTrace: A privacy-preserving protocol for community-driven contact tracing across borders**

Jason Bay, Joel Kek, Alvin Tan, Chai Sheng Hau, Lai Yongquan, Janice Tan, Tang Anh Qu

### **DESCRIPTION:**

TraceTogether is the first national deployment of a Bluetooth-based contact tracing system in the world. It was developed by Singapore's Government Technology Agency and the Ministry of Health to help the country better respond to epidemics. Following its release, more than 50 governments have expressed interest in adopting or adapting TraceTogether for their countries. Responding to this interest, we are releasing an overview of BlueTrace, the privacy-preserving protocol that underpins TraceTogether, as well as OpenTrace, a reference implementation. OpenTrace comprises the source code for an iOS app, an Android app, a cloud-based backend, and baseline signal strength calibration data. This will be made available to the open source community at [github.com/opentrace-community](https://github.com/opentrace-community) on 9 April 2020.

### **ADVANTAGES:**

Data protection and Privacy safeguards.

## **DISADVANTAGES:**

Encounter Message replay/relay attack, Implementation-

i)Challenges -iOS background Bluetooth limitations

ii)Difference in transmission power across devices.

## **2.10 Development of An Android Application for Viewing Covid-19 Containment Zones and Monitoring Violators Who are Trespassing into It Using Firebase and Geofencing**

Ranajoy Mallik, Amlan Protim Hazarika, Sudarshana Ghosh Dastidar, Dilip Sing, and Rajib Bandyopadhyay

## **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.

## **ADVANTAGES:**

Firebase cloud Firestore database with location data of containment zones and Geofencing.

## **DISADVANTAGES:**

Updates the locations of the areas in a Google map which are identified to be the containment zones.

### **2.11 A Survey of COVID-19 Contact Tracing Apps**

Nadeem Ahmed; Regio A. Michelin; Wanli Xue; Sushmita Ruj; Robert Malaney; Salil S. Kanhere

## **DESCRIPTION:**

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.

## **ADVANTAGES:**

Data management, privacy and security Proximity estimation

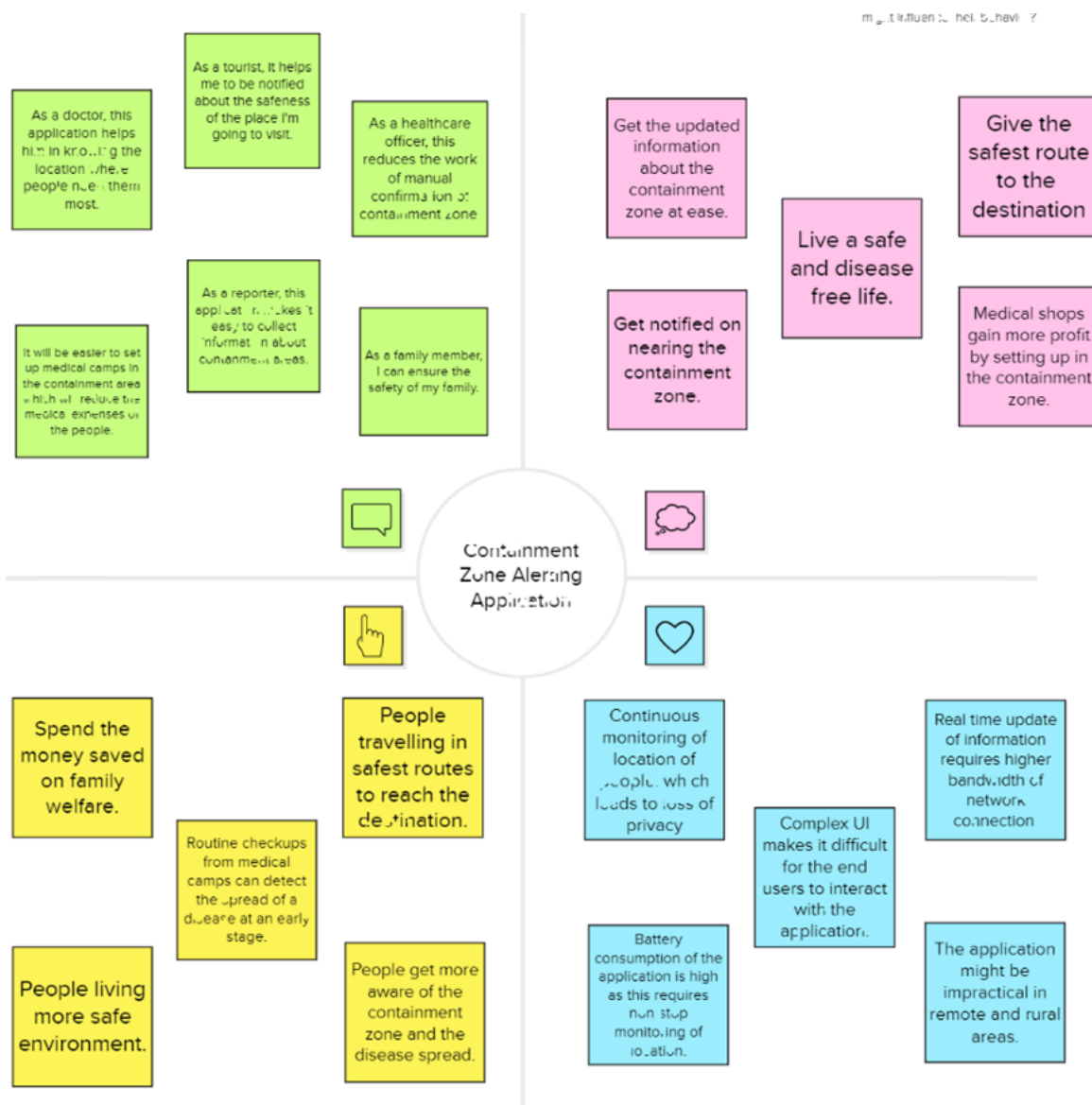
## **DISADVANTAGES:**

Wireless Device Tracking, Location confirmation, Enumeration attack, Denial of service.

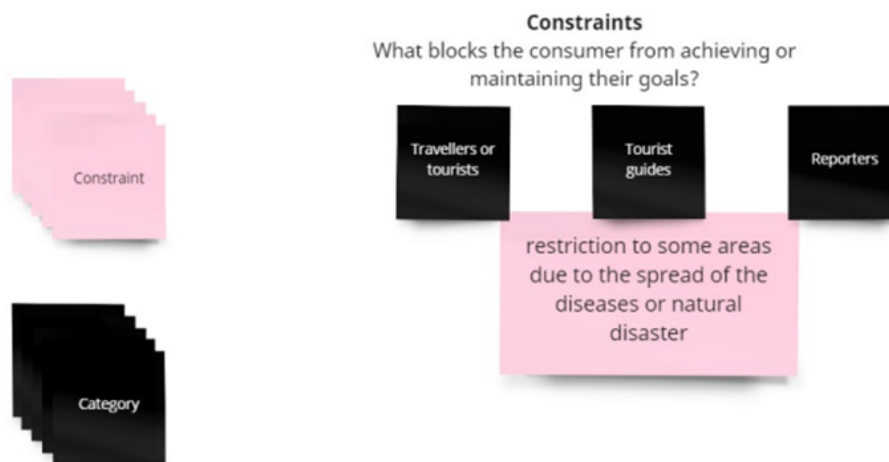
### 3. IDEATION & PROPOSED SOLUTION

#### 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 help 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.



## 3.2 Ideation & Brainstorming



### Group 1

I did it. I started a new role and finally built a team to drive my initiative.

Focused more on leadership

Focused more on managing stress

I noticed that we struggle to collaborate effectively and communication takes over my time. Worst of all, we lack space for creativity.

By being an example and engaging others, I try to champion a new way of working in my teams or organization. This is sometimes difficult as people are hesitant to embrace the change.

Let's keep this one

miro

### Job to be Done

Describe in one sentence what progress the consumer seeks in their life.

Help me to

monitor of  
people's  
location  
continuously

ensure we  
can all work  
in one place  
with ease

experience,  
effect of a  
solution

so that I can

benefit /  
progress

inspire  
everyone to  
collaborate  
more often

miro

### Choice Set

What solutions is the consumer using today?

What did they stop using?

What did they consider using?

#### Hired

A magical application  
solution

#### Considered

Design  
thinking  
trainings

GPS navigation  
and  
monitoring of  
containment  
areas

#### Fired

In-person  
workshops



### 3.3 Proposed Solution

To create an easy-to-use android application shows the location of the containment zones to the users to warn them not to enter the affected region utilizing Cloud and Geofencing . It also notifies the user when he or she trespasses the boundary of a containment zone or stays in the containment zones. This is done with the help of integration of Google maps.



### 3.4 Problem Solution fit

Project Title: Containment Zone Alerting app

Project Design Phase-I - Solution Fit Template

Team ID: PNT2022TMID52614

Define CS, fit into CC	<p>1. CUSTOMER SEGMENT(S) <span>CS</span></p> <p>This app is basically for everyone as no one is safe from virus or disease. But this app is mainly for who travels a lot.</p>	<p>6. CUSTOMER CONSTRAINTS <span>CC</span></p> <p>User should allow app to record their GPS location all the time.</p>	<p>5. AVAILABLE SOLUTIONS <span>AS</span></p> <p>Through the internet they can find containment zone and newspaper also alert all their users.</p>	Explore AS, differentiate
Focus on JAP, fit into BE	<p>2. JOBS-TO-BE-DONE / PROBLEMS <span>JP</span></p> <p>Through COVID/Disease Statistics and its GPS detect location of containment zone.If user enters to those particular zone alert those user.</p>	<p>9. PROBLEM ROOT CAUSE <span>RC</span></p> <p>As people wont be able to find by themselves that if a particular area is contained or not so more chance of spreading any user .e.</p>	<p>7. BEHAVIOUR <span>BE</span></p> <p>At first it will track user's location then if in anytime user enters containment zone app will alert the user.All containment zone and its details also shared.</p>	Focus on JAP, fit into BE
TR	<p>3. TRIGGERS <span>TR</span></p> <p>Whenever user enters the containment zone which is marked by app will alert the user.</p>	<p>10. YOUR SOLUTION <span>SL</span></p> <p>A real time tracker which will track all the time.It will alert the user when a user enters to a contained zone.</p>	<p>8. CHANNELS of BEHAVIOUR <span>CH</span></p> <p>Updating the new data to the app so it should be online while updating.</p> <p>It should allow user to download the data</p>	CH
EM	<p>4. EMOTIONS: BEFORE / AFTER <span>EM</span></p> <p><b>Before</b> using this app they will fear to travel out because of containment Zone but <b>after</b> this app they will fell safe to travel out.</p>		<p>whenever they need to so they can use it in offline mode.</p>	

## 4. REQUIREMENT ANALYSIS

### 4.1 Functional Requirements

FR No.	Functional Requirement	Sub Requirement (story / sub-task)
FR 01	User / Consumer Registration	Registration through GMAIL / Registration through phone number
FR 02	User / Consumer Confirmation	Confirmation through verification mail / Confirmation through OTP
FR 03	User Navigation Details	User location should be ON always
FR 04	Phone Battery	As this application wants the user's location details, draining of phone battery is more. So high end phones are required.

### 4.2 NonFunctional requirements

NFR No.	Non-Functional Requirements	Description
NFR 01	Usability	User friendly application, rate of providing wrong information is too less.

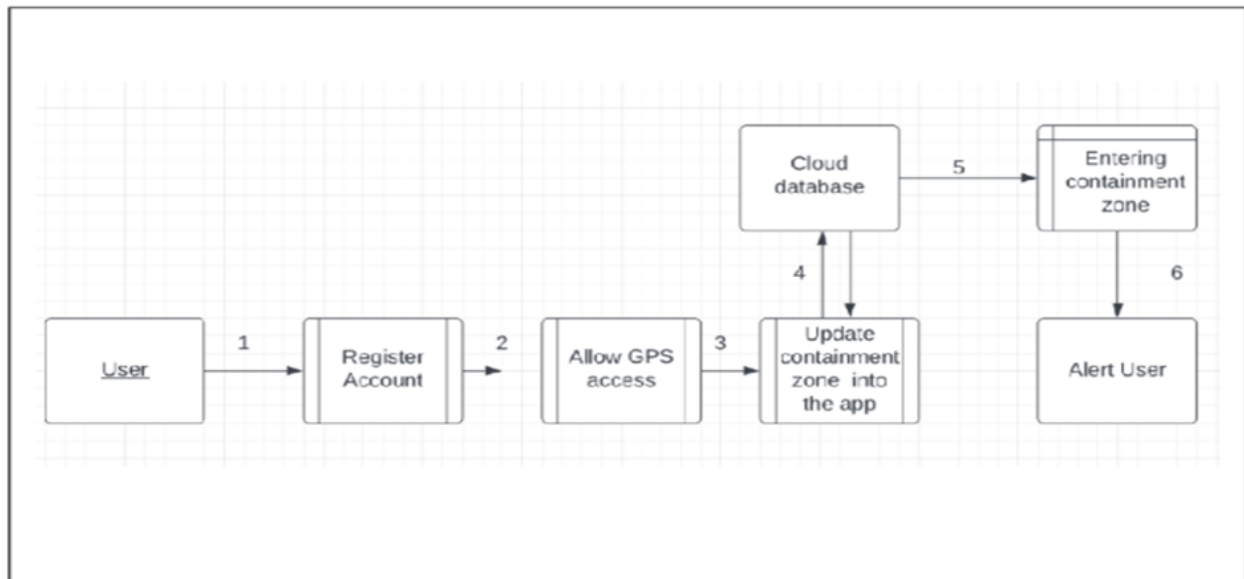
NFR 02	Security	Though the application requires user's phone number, location / navigation details, those data is a safe and secured.
NFR 03	Reliability	The information about the containment zones is precise. The system must perform without failure in 95 percent of use cases.
NFR 04	Performance	The home page supporting 5,000 users per hour must provide 6 second or less response time. Also when the user enter into the containment area without any knowledge, this application give an alert notification within 2 minutes.
NFR 05	Availability	The web dashboard must be available to end users 99.98 percent of the time every day.
NFR 06	Scalability	The system must be scalable enough to support 1,000,000 visits at the same time while maintaining optimal performance.

## 5.PROJECT DESIGN

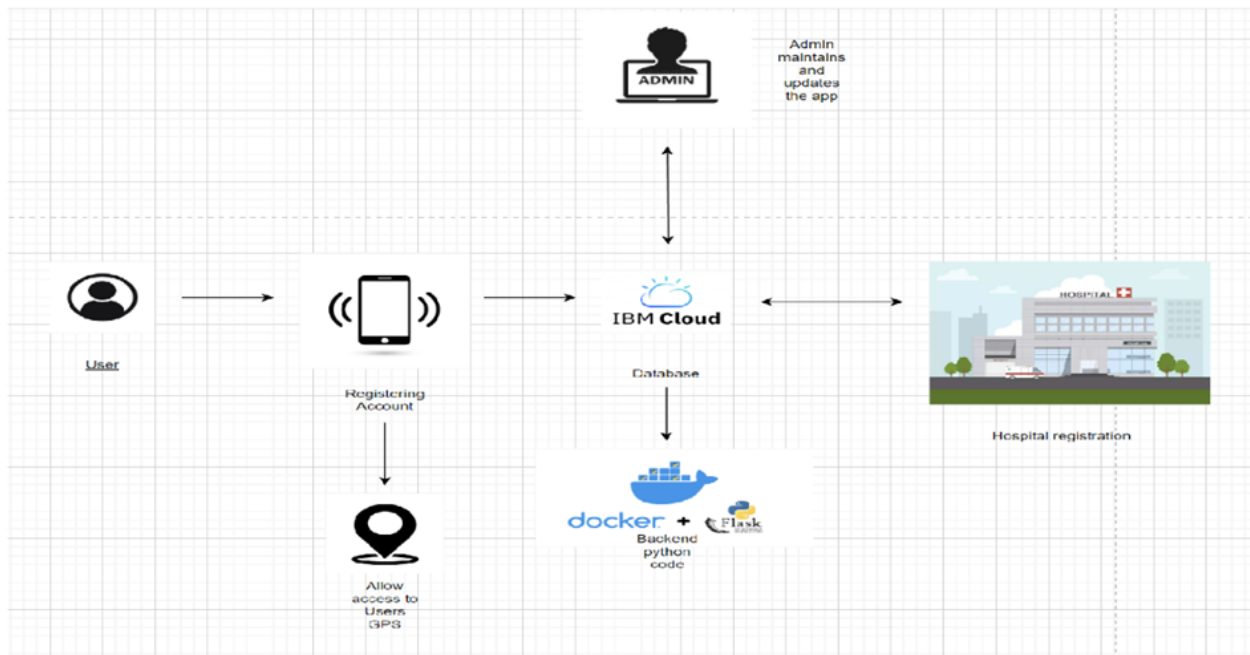
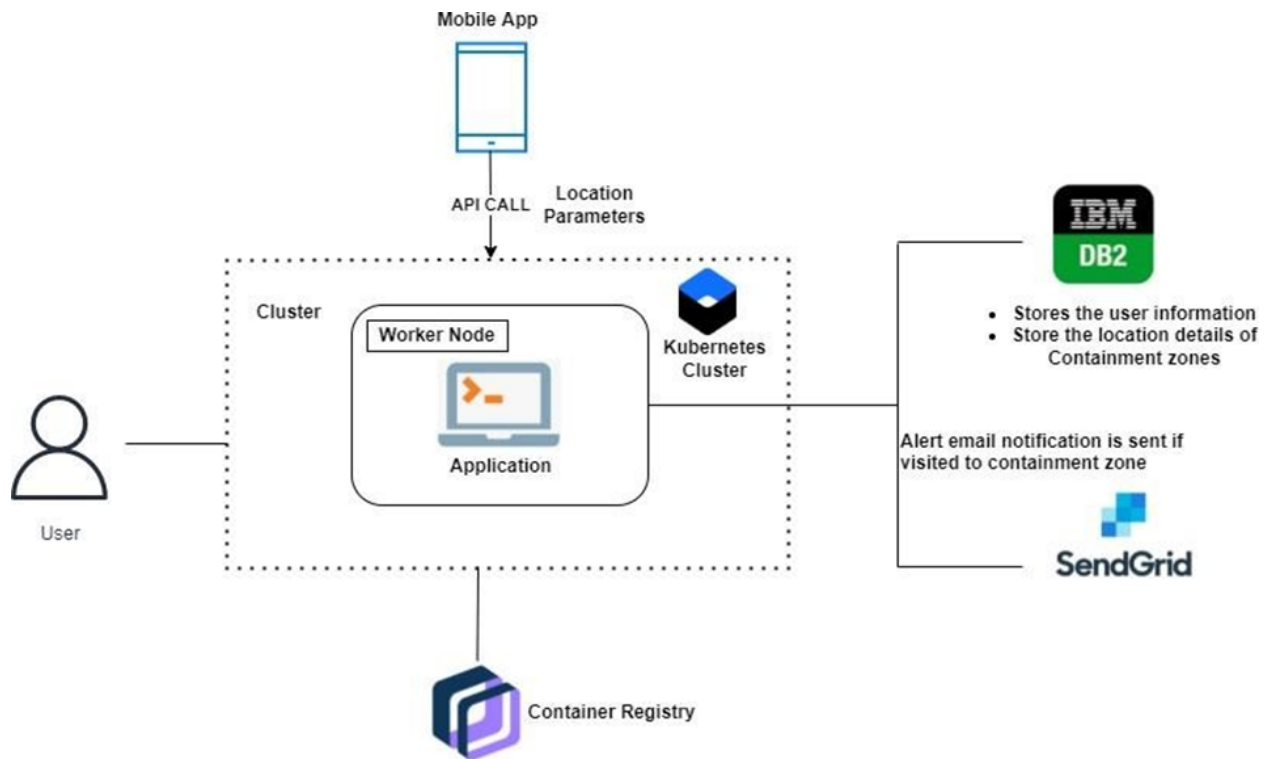
### Data Flow Diagrams

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.

#### 5.1 Data Flow Diagram



## 5.2 Solution & Technical Architecture



### 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
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	Low	Sprint-2

				Facebook Login		
		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-7	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			
	Location Access	USN-8	As a User, I can viewed into the page, if there is any condition to access the location.	Location can be turned through Control centre.	High	Sprint-2

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
	Contaminated Zones	USN-9	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-10	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
Hospital Administrator	Registration	USN-11	As a management, I need to register my hospitals on the site.	I can see registered hospital in the hospital dashboard.	High	Sprint-1
	Login	USN-12	As a management, I need to login into my dashboard with my given hospitalid and password.	I can see my dashboard after login.	Medium	Sprint-1
	Dashboard	USN-13	As a management, I need to enter the case information of the patient that visits our hospital.	I can view the patient information on the dashboard.	High	Sprint-2

		USN-1 4	As a management, I need to store all the patient information on the cloud.		High	Sprint-2
Administrator	Services	USN-1 5	As an admin, I need to provide valid information about the pandemic out there.	I can get the pandemic updates out there.	High	Sprint-2
		USN-1 6	As an admin, I need to provide medical advice through a chatbot.	I get medicinal recommendations through a chatbot.	Medium	Sprint-3
		USN-1 7	As an admin, I need to provide medical recommendations by collaborating with top hospitals.	I get medical instruction through chief doctors.	Low	Sprint-3
		USN-1 8	As an admin, I need to alert the user when they	I got a notification when I am in	Medium	Sprint-4

			enter pandemic zones.	the pandemic area.		
--	--	--	-----------------------	--------------------	--	--

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
		USN-19	As an admin, I need to provide preventive measures when they travel through it.	I got a remedies notification when I am in the pandemic area.	High	Sprint-3
		USN-20	As an admin, I need to provide special services for premium users by giving services like monitoring health by their smart bands.	I was treated special after becoming a premium member.	Low	Sprint-4
	Data Collection	USN-21	As an admin, I need to store all the user information on the cloud.	I can access my information when I needed.	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	As a user, I can register for the application by entering my email, password, and confirming my password.	2	High	Abhishek Arul Dividharshana Deepika
Sprint-1		USN-2	As a user, I will receive confirmation email once I have registered for the application	3	High	Abhishek Arul Dividharshana Deepika
Sprint-2		USN-3	As a user, I can register for the application through Facebook	3	Low	Abhishek Arul Dividharshana Deepika
Sprint-1		USN-4	As a user, I can register for the	2	Medium	Abhishek Arul

			application through Gmail			Dividharshana Deepika
Sprint-1	Login	USN-5	As a user, I can log into the application by entering email & password	3	High	Abhishek Arul Dividharshana Deepika

<b>Sprint</b>	<b>Functional Requirement (Epic)</b>	<b>User Story Number</b>	<b>User Story / Task</b>	<b>Story Points</b>	<b>Priority</b>	<b>Team Members</b>
	Dashboard	USN-6	As a User , Can I manually plot the alerted zone for my convenience only.	2	Low	Abhishek Arul Dividharshana Deepika
Sprint-2	Registration	USN-7	As a user, I can register for the application by entering my email, password, and confirming my password.	3	High	Abhishek Arul Dividharshana Deepika

			Confirmation code has been send through the registered mail id, phone number	3	Low	Abhishek Arul  Dividharshana Deepika
Sprint 2	Location Access	USN-8	As a User , I can view the page , if there is any condition to access the location.	5	High	Abhishek Arul  Dividharshana Deepika
Sprint 3	Contaminated Zones	USN-9	Is it accurately shown off the alerted zone If I Entered into the zone the messages are properly received through email.	5	High	Abhishek Arul  Dividharshana Deepika
Sprint 4	Frequent Updates	USN-10	Admin are necessary to update the recent containment through their portals and these seen through the app.	3	Medium	Abhishek Arul  Dividharshana Deepika

Sprint 1	Registration	USN-11	As a management, I need to register my hospitals on the site.	5	High	Abhishek Arul Dividharshana Deepika
Sprint 1	Login	USN-12	As a management, I need to login into my dashboard with my given hospital id and password.	3	Medium	Abhishek Arul Dividharshana Deepika
Sprint 2	Dashboard	USN-13	As a management, I need to enter the case information of the patient that visits our hospital.	2	High	Abhishek Arul Dividharshana Deepika

<b>Sprint</b>	<b>Functiona l Requirem ent (Epic)</b>	<b>User Story Number</b>	<b>User Story / Task</b>	<b>Story Point s</b>	<b>Priorit y</b>	<b>Team Members</b>
Sprint 2		USN-14	As a management, I need to store all the patient information on the cloud.	2	High	Abhishek Arul  Dividharshana Deepika
Sprint 2	Services	USN-15	As an admin, I need to provide valid information about the pandemic out there .	2	High	Abhishek Arul  Dividharshana Deepika
Sprint 3		USN-16	As an admin, I need to provide medical advice through a chatbot.	5	Mediu m	Abhishek Arul  Dividharshana Deepika
Sprint 3		USN-17	As an admin, I need to provide medical recommendations by collaborating with top hospitals.	5	Low	Abhishek Arul  Dividharshana Deepika



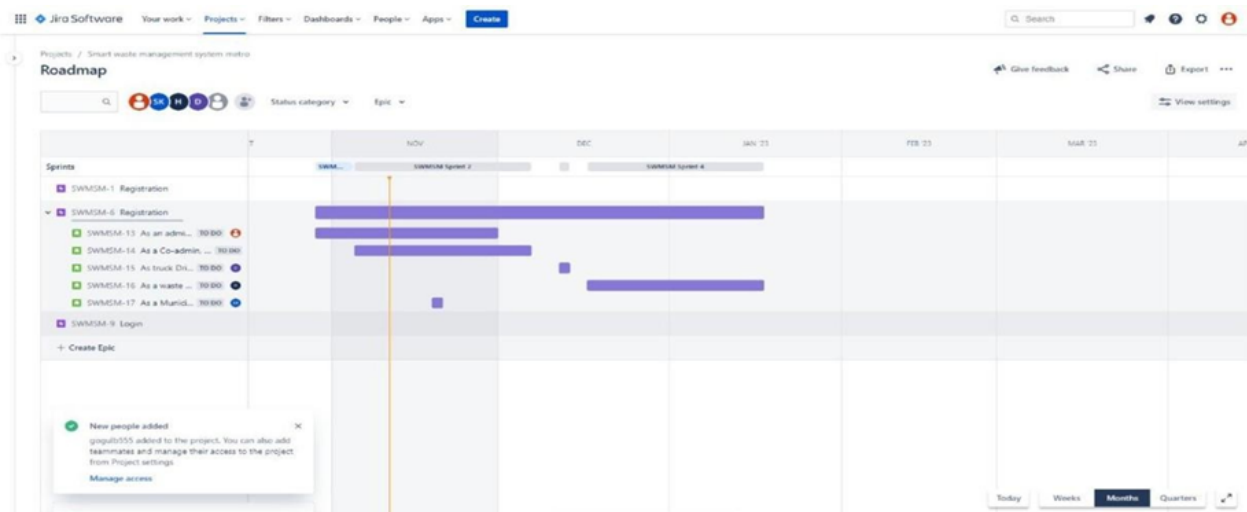
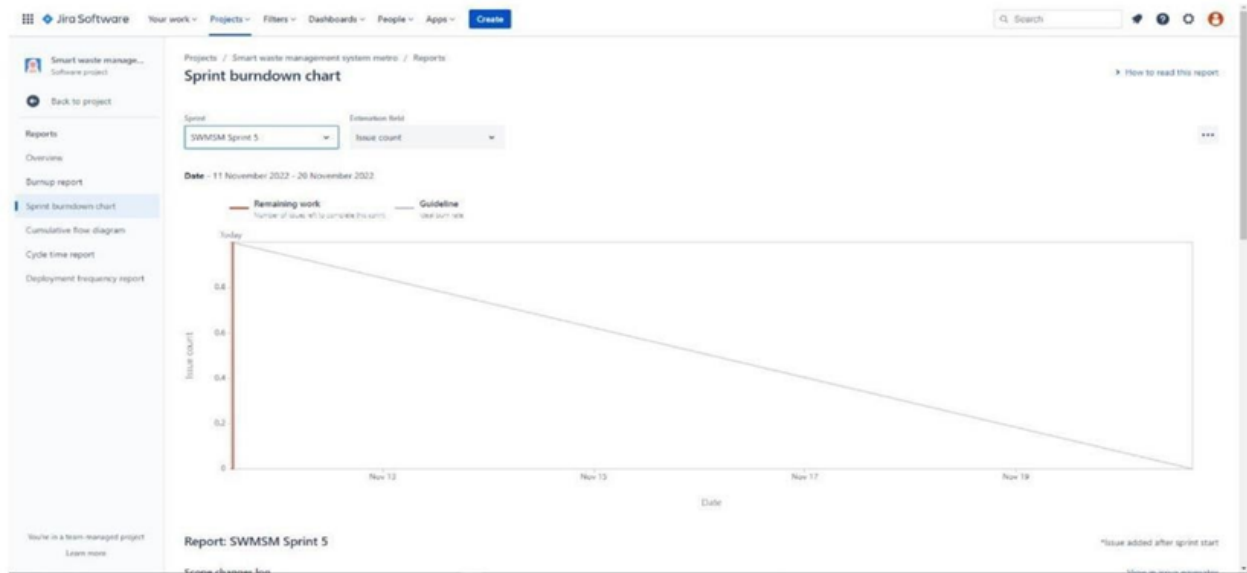
Sprint 4		USN-18	As an admin, I need to alert the user when they enter pandemic zones.	7	Medium	Abhishek Arul Dividharshana Deepika
Sprint 3		USN-19	As an admin, I need to provide preventive measures when they travel through it.	5	High	Abhishek Arul Dividharshana Deepika
Sprint 4		USN-20	As an admin, I need to provide special services for premium users by giving services like monitoring health by their smart bands.	4	Low	Abhishek Arul Dividharshana Deepika
Sprint 4		USN-21	As an admin, I need to store all the user information on the cloud.	6	Medium	Abhishek Arul Dividharshana Deepika

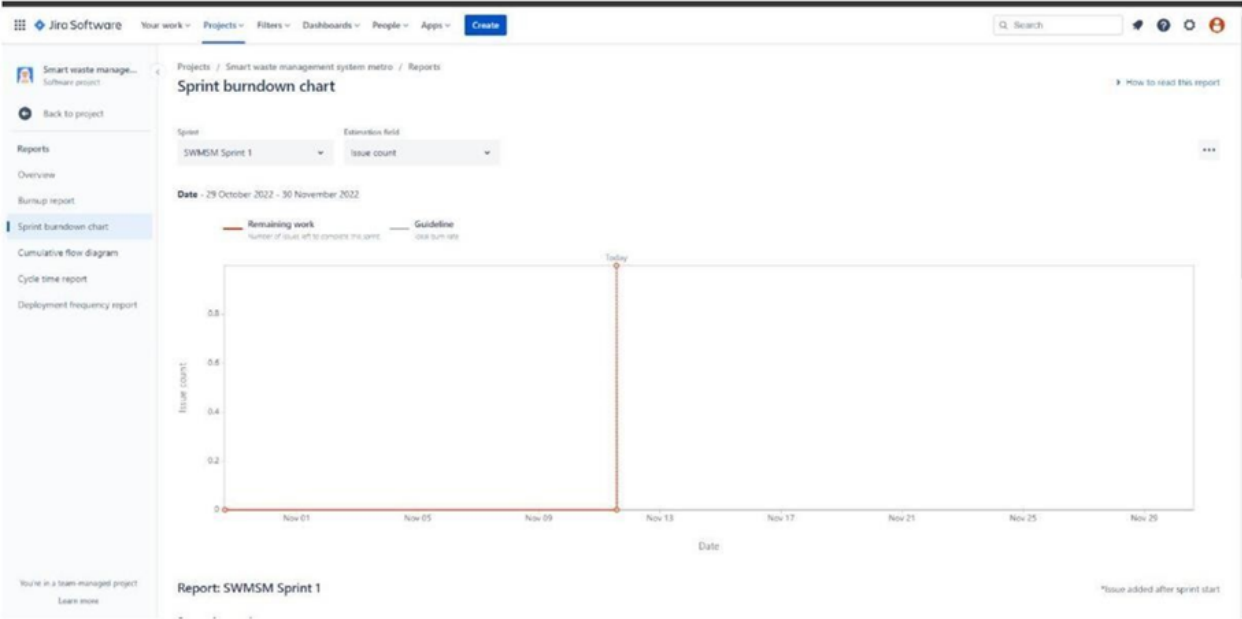
**Project Tracker, Velocity & Burndown Chart:**

<b>Sprint</b>	<b>Total Story Points</b>	<b>Duration</b>	<b>Sprint Start Date</b>	<b>Sprint End Date (Planned)</b>	<b>Story Points Completed (as on Planned End Date)</b>	<b>Sprint Release Date (Actual)</b>
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

## 6.2 Reports from JIRA

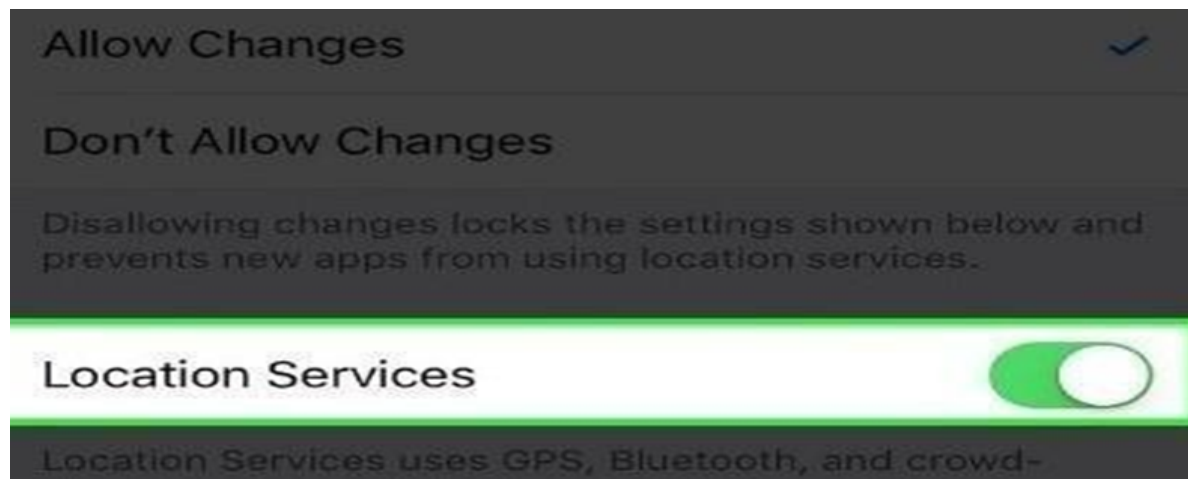
### BURNDOWN CHART



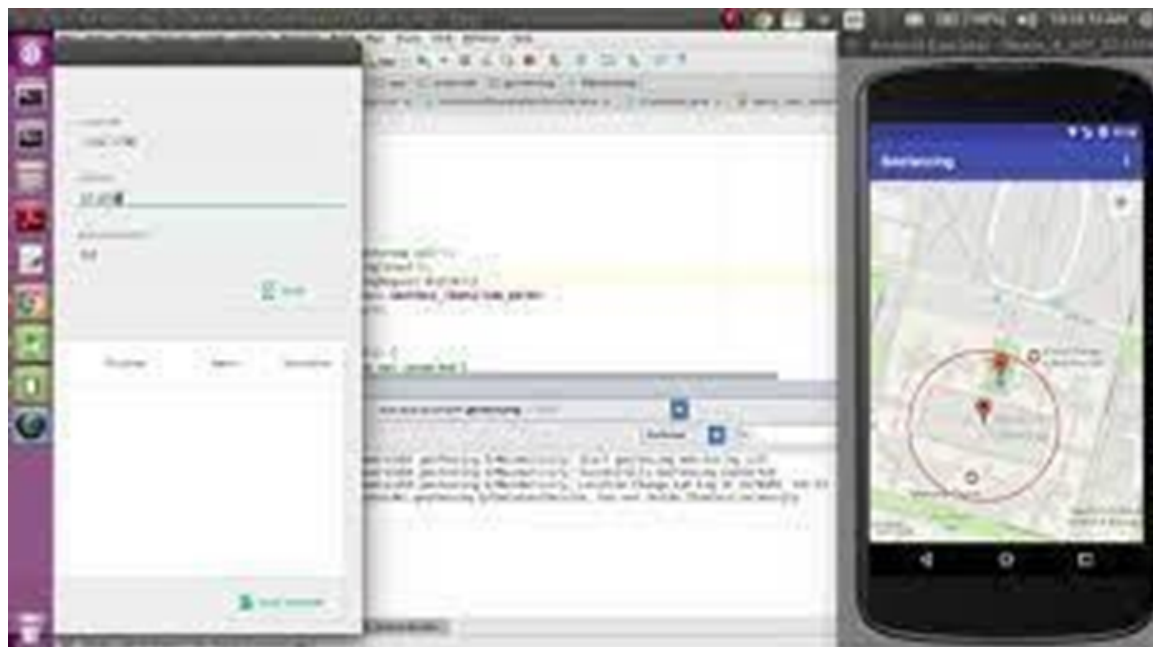
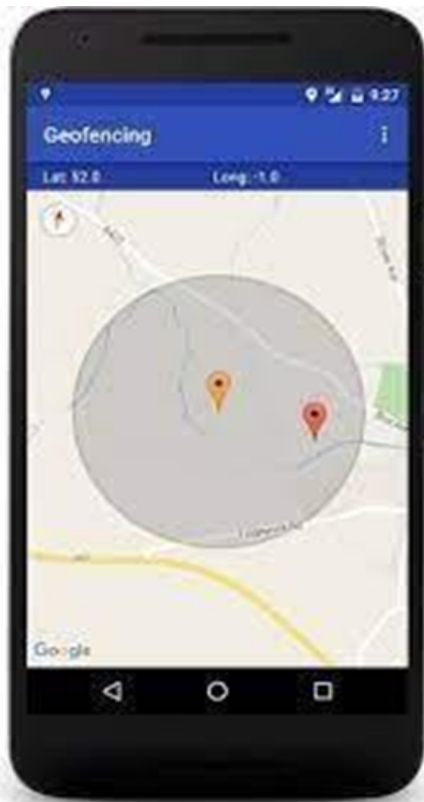


## 7.CODING AND SOLUTIONING

### 7.1 Location Services



# GEOFENCE IN ANDROID APPLICATION









## 8.TESTING

### 8.1 Testing


### 8.2 User Acceptance Testing

**Declare Containment Zone**

welcome: admin

Lat.:  Long.:

Get current Location: [Current Location](#) (Click this first)



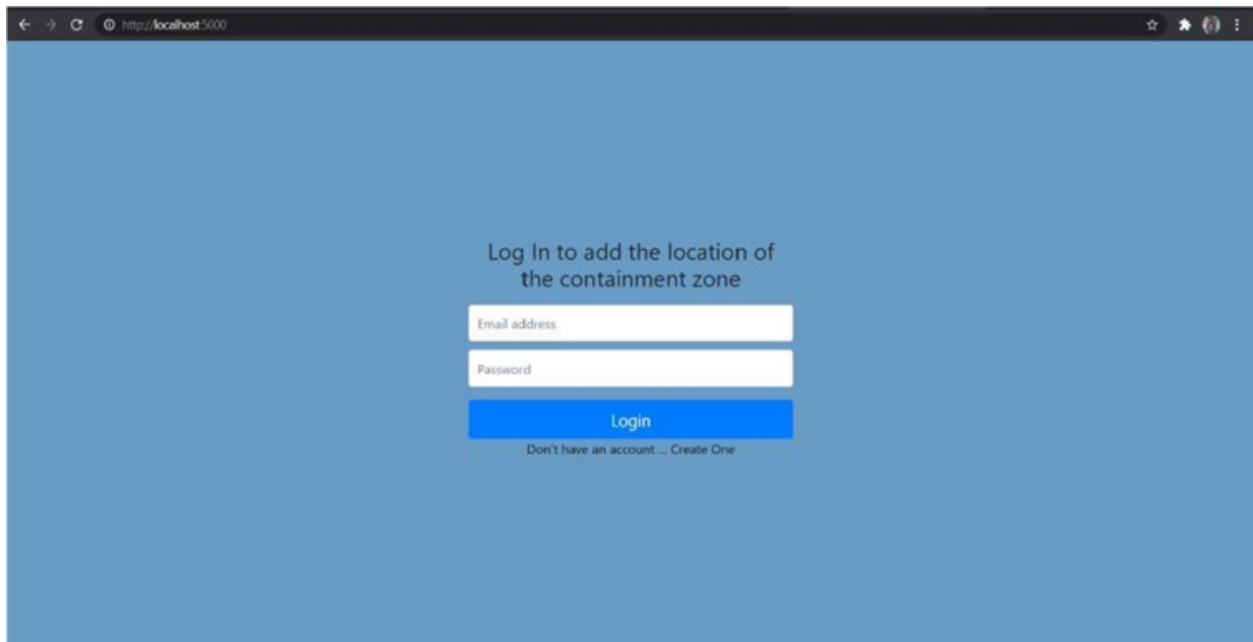
[Tutorial](#) [Declare Containment Zone](#)

## 9.RESULTS

The application efficiently displays the identified COVID-19 containment zones to users in a Google map. With the alarming increase in COVID-19 affected cases around the world, this developed application can be used to raise social awareness among the general public. This software tracks the user's location and determines whether it is in the list of identified containment zones. It sends the user separate notification alerts when entering and exiting the containment areas. The developed Android application extracts the IMEI Number of the trespasser in the containment zones, which can help local police track and identify people who frequently trespass the containment zones. As a result, this application determines the containment.

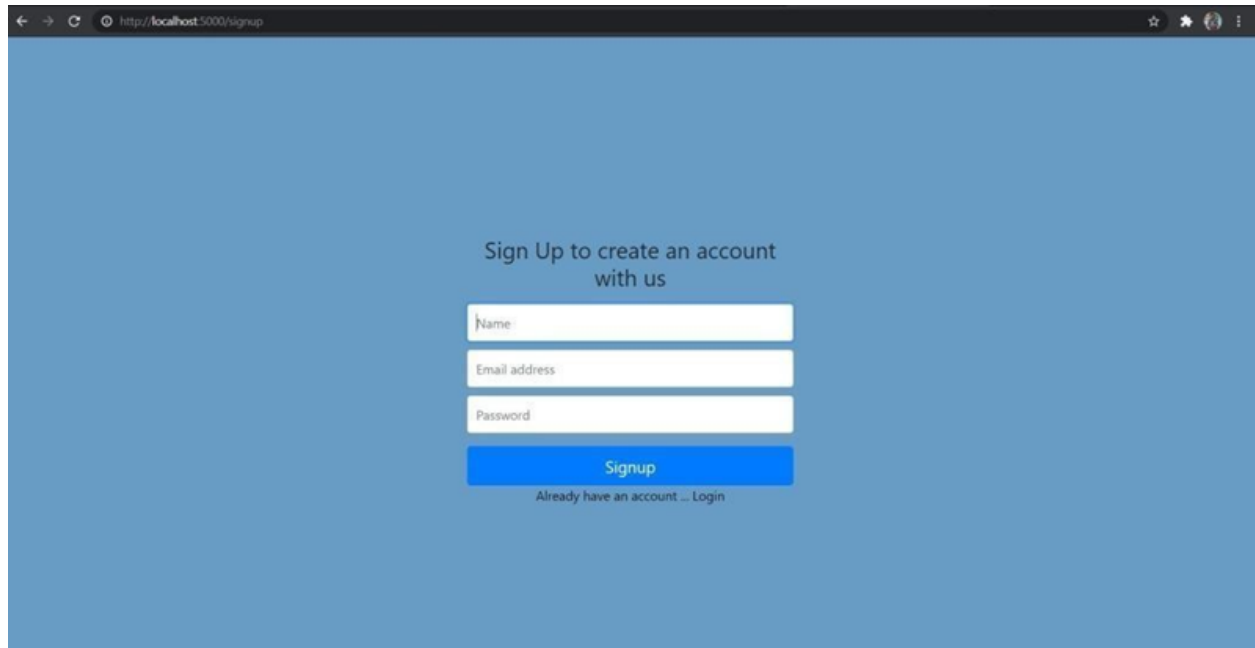
### User Interaction with Application:

#### 1)Login page:



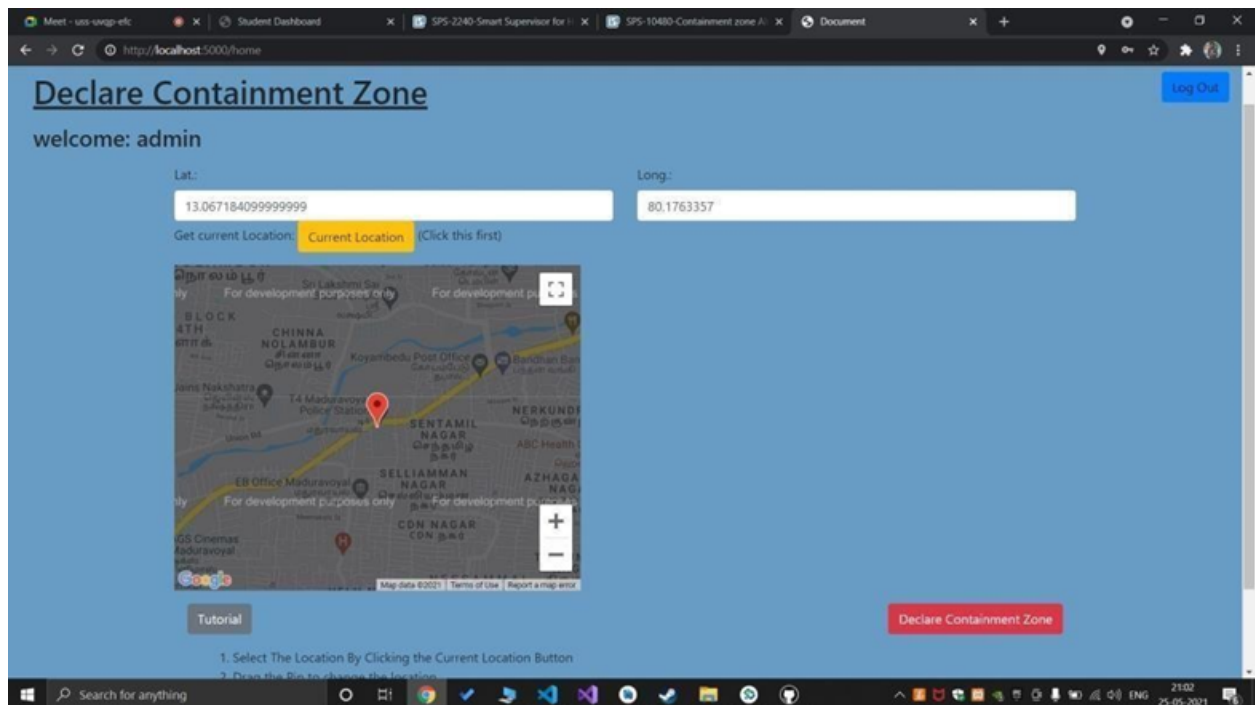
The screenshot shows a web browser window with the address bar displaying "http://localhost:5000". The page has a solid blue background. In the center, there is a white login form. The form contains the text "Log In to add the location of the containment zone" at the top. Below this text are two input fields: "Email address" and "Password". Below the input fields is a blue button labeled "Login". At the bottom of the form, there is a link that says "Don't have an account ... Create One".

## 2) Register page:



A screenshot of a web browser showing a registration page. The browser's address bar displays `http://localhost:5000/signup`. The page has a solid blue background. In the center, the text "Sign Up to create an account with us" is displayed. Below this text are three white input fields with blue borders, labeled "Name", "Email address", and "Password". A blue button with the text "Signup" is positioned below the input fields. At the bottom of the form, there is a link that says "Already have an account ... Login".

## 3) Home page:



A screenshot of a web browser showing a home page. The browser's address bar displays `http://localhost:5000/home`. The page has a solid blue background. At the top left, the text "Declare Containment Zone" is displayed in a large, bold font. Below this text, the text "welcome: admin" is displayed. There are two white input fields with blue borders. The first field is labeled "Lat:" and contains the value "13.067184099999999". The second field is labeled "Long:" and contains the value "80.1763357". Below these fields, there is a button labeled "Current Location" and a link that says "(Click this first)". Below the input fields and button, there is a map of a region in India, showing various locations and roads. Below the map, there is a button labeled "Tutorial" and a red button labeled "Declare Containment Zone". At the bottom of the page, there is a list of instructions: "1. Select The Location By Clicking the Current Location Button" and "2. Draw the Pin to choose the location". The browser's taskbar is visible at the bottom, showing various application icons and the system clock.

#### 4) Location data page:

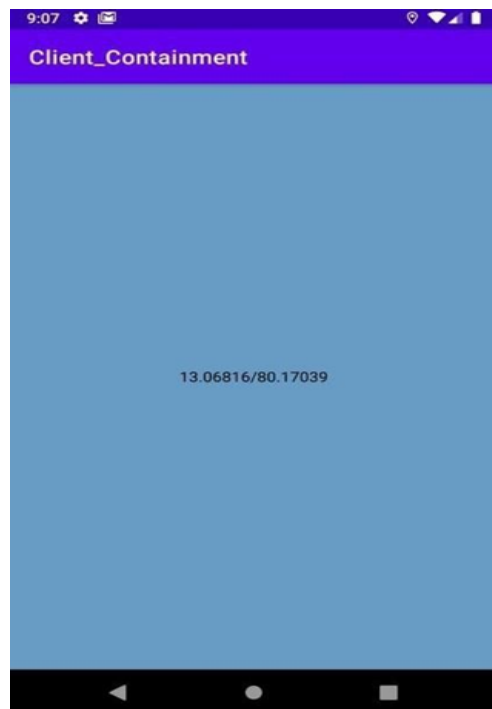


A screenshot of a web browser displaying a page titled "Location data and Visited People". The page has a light blue background. At the top, there is a title "Location data and Visited People" in a dark blue font. Below the title is a table with four columns: "S.No", "Latitude", "Longitude", and "No. Visited". The table contains seven rows of data. In the bottom right corner, there is a red button with the text "Go to location update Page".

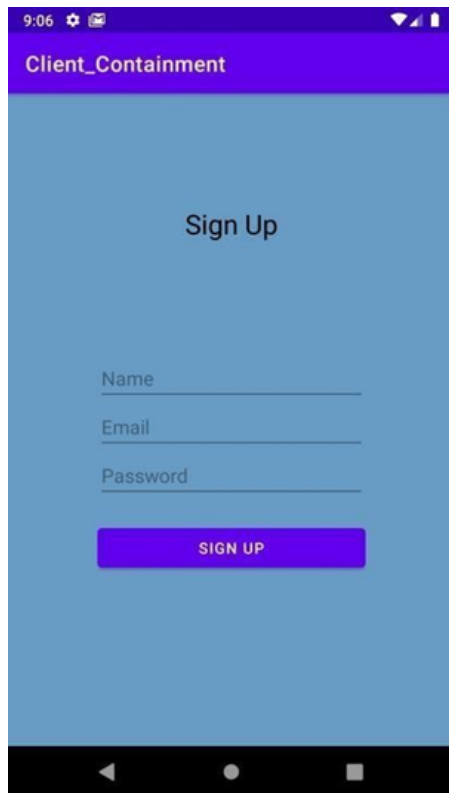
S.No	Latitude	Longitude	No. Visited
1	13.069148883848849	80.17551259999999	0
2	13.068498821079215	80.1704513893799	0
3	12.979174795975714	77.59973092596437	0
4	14.469858338289407	75.91959519903565	0
5	13.062359612480321	77.5638966135254	0
6	15.840542738858232	76.64209647695924	0
7	15.3172775	75.7138884	0

Go to location update Page

#### 5)Current Location:



## 6)Client-Containment Zone:



A screenshot of a mobile application interface titled "Client\_Containment". The screen has a light blue background. At the top, there is a purple header bar with the text "Client\_Containment". Below the header, the text "Sign Up" is centered. Underneath, there are three input fields labeled "Name", "Email", and "Password". Below these fields is a purple button with the text "SIGN UP". The bottom of the screen shows a black navigation bar with three white icons: a back arrow, a circle, and a square.

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 AND DISADVANTAGES**

### **10.1 ADVANTAGES**

- People can be alerted before entering the containment zone.
- Further spread of virus can be reduced considerably

### **10.2 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 individual's location. Key benefits of the application are monitoring people's 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, 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 organizations to help act faster.
- 2) The dataset obtained from the application can be used for predictive analysis to determine prone areas and include special methods for tackling the problem in those areas.
- 3) Emergency signal in case of network failure and internet connection loss.
- 4) Tackling the victim's movements.
- 5) Improved Google positioning system's precision.

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

For analysis purposes, 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 quarters.

## 13.APPENDIX

[1] COVID-19 Contact-tracing Apps: a Survey on the Global Deployment and Challenges.

Jinfeng Li, Xinyi Guo

[2] PANDEMIC CONTACT TRACING APPS: DP-3T, PEPP-PT NTK, AND ROBERT FROM A PRIVACY PERSPECTIVE Fraunhofer AISEC Garching near Munich

info@aisec.fraunhofer.de 27 April 2020

[3] Paper 2020/493 Towards Defeating Mass Surveillance and SARS-CoV-2: The Pronto-C2 Fully Decentralized Automatic Contact Tracing System Gennaro Avitabile, Vincenzo Botta, Vincenzo Iovino, and Ivan Visconti

[4] PACT: Privacy Sensitive Protocols and Mechanisms for Mobile Contact Tracing Justin Chan, Dean Foster, Shyam Gollakota, Eric Horvitz, Joseph Jaeger, Sham Kakade, Tadayoshi Kohno, John Langford, Jonathan Larson, Puneet Sharma, Sudheesh Singanamalla, Jacob Sunshine, Stefano Tessaro

[5] Trust and Transparency in Contact Tracing Applications Stacy Hobson, Michael Hind, Aleksandra Mojsilović, Kush R. Varshney IBM Research AI Yorktown Heights, New York.

[6] Tracking the Covid zones through geo-fencing technique Anto Arockia Rosaline R., Lalitha R., Hariharan G., Lokesh N.

[7] 2021 Jan;45(1):65-69. doi: 10.1007/s00264-020-04646-2. Epub 2020 Nov 14.The experience of contact tracing in Singapore in the control of COVID-19: highlighting the use of digital technology Sean Han Sheng Lai , Camelia Qian Ying Tang , Asok Kurup , Gowreeson Thevendran

[8] Privacy, Ethics, and Contact-tracing Apps Teresa Scassa, Jason Millar and Kelly Bronson, "Privacy, Ethics, and Contact-tracing Apps", in C.M. Flood, V. MacDonnell, J. Philpott, S. Thériault and S. Venkatapuram, eds. Vulnerable: The Law and Policy of COVID-19, University of Ottawa Press, 2020. Ottawa Faculty of Law Working Paper No. 2020-23



[9] BlueTrace: A privacy-preserving protocol for community-driven contact tracing across borders Jason Bay, Joel Kek, Alvin Tan, Chai Sheng Hau, Lai Yongquan, Janice Tan, Tang Anh Quy Government Technology Agency Singapore

[10] Development of An Android Application for Viewing Covid-19 Containment Zones and Monitoring Violators Who are Trespassing into It Using Firebase and Geofencing. Ranajoy Mallik, Amlan Protim Hazarika, Sudarshana Ghosh Dastidar, Dilip Sing, and Rajib Bandyopadhyay

[11]A Survey of COVID-19 Contact Tracing Apps IEEE Nadeem Ahmed; Regio A. Michelin; Wanli Xue; Sushmita Ruj; Robert Malaney; Salil S. Kanhere.

## SOURCE CODE:

```
<!DOCTYPE html>

<html lang="en">

<head>

<!-- Required meta tags -->

<meta charset="utf-8" />

<meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no"
/>

<!-- Bootstrap CSS -->

<link rel="stylesheet"
href="https://stackpath.bootstrapcdn.com/bootstrap/4.4.1/css/bootstrap.min.css"

integrity="sha384-Vkoo8x4CGsO3+Hhvx8T/Q5PaXtkKtu6ug5TOeNV6gBiFeWPGFN9
MuhOf23Q9Ifjh" crossorigin="anonymous" />

<link rel="stylesheet" href="style.css" />

<title>Log In</title>

<link rel="stylesheet" href="{{ url_for('static', filename='styles.css') }}">

</head>

<body class="text-center">

{% if error == 1 %}

<script>
```

```

        alert("Incorrect Password");
    </script>

    {% elif error == 2%}

    <script>

        alert("Create An Account");

    </script>

    {% else %}

    {% endif %}

    <form class="form-login" method="POST" action="/">

        <h1 class="h3 mb-3 font-weight-normal">Log In to add the location of the
containment zone</h1>

        <label for="email" class="sr-only">Email address</label>

        <input type="email" name="email" class="form-control" placeholder="Email
address" required autofocus />

        <label for="password" class="sr-only">Password</label>

        <input type="password" class="form-control" placeholder="Password"
name="password" required />

        <button type="submit" class="btn btn-lg btn-primary btn-block mt-3">

            Login

        </button>

        <a href={ {url_for("signup")} }>Don't have an account ... Create One</a>

    </form>

    <!-- Optional JavaScript -->

    <!-- jQuery first, then Popper.js, then Bootstrap JS -->

```

```
<script src="https://code.jquery.com/jquery-3.4.1.slim.min.js"
```

```
integrity="sha384-J6qa4849bIE2+poT4WnyKhv5vZF5SrPo0iEjwBvKU7imGFAV0wwj1  
yYfoRSJoZ+n"
```

```
crossorigin="anonymous"></script>
```

```
<script src="https://cdn.jsdelivr.net/npm/popper.js@1.16.0/dist/umd/popper.min.js"
```

```
integrity="sha384-Q6E9RHvbIyZFJoft+2mJbHaEWldlvI9IOYy5n3zV9zzTtmI3UksdQR  
VvoxMfooAo"
```

```
crossorigin="anonymous"></script>
```

```
<script src="https://stackpath.bootstrapcdn.com/bootstrap/4.4.1/js/bootstrap.min.js"
```

```
integrity="sha384-wfSDF2E50Y2D1uUdj0O3uMBJnjuUD4Ih7YwaYd1iqfktj0Uod8GC  
Exl3Og8ifwB6"
```

```
crossorigin="anonymous"></script>
```

```
</body>
```

```
</html>
```

```
App.py
```

```
# import statements
```

```
from logging import error
```

```
from flask import *
```

```
from jinja2.utils import select_autoescape
```

```
import bcrypt
```

```
from flask_mysql 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.7BJDtQDIS8unH0r5_TufVQ.Ykpcz19QcqgcNwYZC3a0mNRPhGksG117YURqOT
            a2HL')

        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)
```

### **GITHUB LINK:**

<https://github.com/IBM-EPBL/IBM-Project-48433-1660807341>

### **YOUTUBE LINK:**

[https://youtu.be/3R1u\\_lWHcNY](https://youtu.be/3R1u_lWHcNY)