IBM Nalaiya Thiran Course

Use-case: **Containment Zone Alerting Application**
Domain: **Cloud Application Development**

Batch: B11-5A1E

Team: Kiran George Gaurdian - 2019103029
Sudharshan K - 2019103068
Kaushik Narayan R - 2019103536
Nanda Kumar R - 2019103545

Literature Survey

COVID-19 is an infectious disease caused by the SARS-CoV-2 virus. The novel virus was first identified from an outbreak in Wuhan, China, in December 2019. Since its emergence, it has led to the deaths of more than 6 million people and more than 600 million reported cases worldwide. It is mainly transmitted as airborne droplets or from the transfer of the virus through contact with an infected person or object. The World Health Organisation (WHO) has declared the outbreak of Covid-19 as a pandemic and recommends maintaining physical contact of at least 1 metre from others, avoiding crowded areas and close contact with infected people. Hence, it would be beneficial to develop an application that can alert people when they enter a quarantined zone or have been in close proximity with someone that tested positive for COVID-19 or any other infectious disease.

When it comes to existing solutions, Aarogya Setu is the most widely used application in India. It is a mobile application developed by the Government of India to connect essential health services with the people. Its key features include automatic contact tracing using Bluetooth, risk status of users, geo-location based COVID-19 statistics and much more. It makes use of Bluetooth and GPS technology to keep track of the movement of an infected person and alerts citizens about areas that person has been to and marks them as vulnerable spots. In the absence of GPS technology, it uses cellular triangulation to estimate the location of a person. While Aarogya Setu can perform contact tracing and notify people that have been exposed to someone infected with COVID-19, it does actively maintain a list of quarantine or containment zones and warn users that enter these zones.

Many applications have been developed in the last two years to tackle COVID-19. Most of these apps deal with broadcasting COVID statistics, precautionary measures against COVID and ensuring that people get the healthcare they need. But there aren't many applications that are capable of identifying containment zones and alerting people in real-time. The difficulty of acquiring the data that is both accurate and real-time is one of the main challenges in creating such an application.

A solution to this might be to obtain or make use of containment zone data from the state governments. Ranajoy Mallik et al. 2020 have developed an android application that performs exactly this. It uses Google APIs like Firebase and Geofencing to achieve this. It is capable of updating locations of containment zones in Google maps and notifies people that are trespassing into it. They have created a real-time database in Cloud Firestore which contains all the data related to the containment zones like coordinates, radius and zone names. The android application can retrieve information from the database. As the app receives the location data, geofences are created and displayed using Google maps in the application. The user's location is also shown and updated constantly in the map.

The main disadvantage of the method proposed by Ranajoy Mallik et al. 2020 is the origin of the containment zone data. The data that they've used is dependent on the State government regularly updating containment zone data. If the government decides to stop doing so or does not release it for public use, it risks the app displaying inaccurate information or even no information. One solution to this problem might be to make use of data from multiple sources. Allowing people to update the map with newly infected persons might be one stream of data.

MoveInSync, a software development company based out of Bangalore has also developed a COVID containment zone tracker. It is mainly a dashboard that can tell users if a particular locality lies in a containment zone. It can display containment zone information for 15 cities. The containment zone boundary data was maintained with the help of geoIQ, a geolocation data specialist. The data is crowdsourced and updated every hour. For the tracker, it utilised Amazon Aurora PostgreSQL, Cloudwatch, Micrometer + Prometheus and Raygun Crash Monitoring to help scaling up. However, one disadvantage is that geoIQ is proprietary, and comes with its own set of limitations.

Akira Suyama et al. 2016 proposed a disaster information system using geofencing technology to detect movement of users and inform them of any risks that they may face. This method can be repurposed for the containment zone application since the underlying use-case is the same. This method makes use of a client-server architecture. The server collects risk information from various sources and the client monitors the user to notify them

of necessary information. Geofencing is a mechanism that makes a virtual fence in a specific area. The application sets a geofence at a dangerous area and gives risk information to the user. In this proposed method, geofencing was implemented using the Core Location framework of iOS which provides a detection of the entries and exits of the user within a specific geographic region. However, the Core Location framework is available only for iOS devices.

References

1. https://www.aarogyasetu.gov.in/
2. Mallik R, Hazarika AP, Ghosh Dastidar S, Sing D, Bandyopadhyay R. Development of An Android Application for Viewing Covid-19 Containment Zones and Monitoring Violators Who are Trespassing into It Using Firebase and Geofencing. Trans Indian Natl. Acad. Eng. 2020;5(2):163–79. doi: 10.1007/s41403-020-00137-3. Epub 2020 Jul 1. PMCID: PMC7328652.
3. A. Suyama and U. Inoue, "Using geofencing for a disaster information system," 2016 IEEE/ACIS 15th International Conference on Computer and Information Science (ICIS), 2016, pp. 1-5, doi: 10.1109/ICIS.2016.7550849.
4. https://www.expresscomputer.in/interviews/moveinsyncs-containment-zone-tracker-aims-at-democratising-information-flow/55826