Smartphone-based GNSS Electronic Monitoring for the Control of Contagious Diseases

Diogo Agostinho Borda de Água Reis
Instituto Superior Técnico
Lisboa, Portugal
diogo.a.reis@tecnico.ulisboa.pt

***Abstract*—Infectious-contagious diseases are becoming increasingly common in society, at the present time we live in. They have a great propagation capacity and it turns out to be difficult to control them in space and time. This is a problem that requires a solution.**

In order to be able to monitor the infection, a system capable of collecting navigation data from its users was designed. This collection is done through an application with the ability to collect GPS coordinates and the names of available Wi-Fi networks. Beyond the application, there is also a website, in which, not only, a map where the different concentration of infected people can be seen, but also, a list containing the names of the networks, where there are more infected people.

After the development of the entire system, it was necessary to test it. During the testing of the system, several scenarios were studied, namely the consequences of inserting positive users to the disease. These consequences created the expected results which are the marking of users who have been with infected ones and who are at risk of contracting the disease.

***Index Terms*—GPS, Wi-Fi, Tracking, App, Pandemic, Disease**

I. State of Art

The dissertation's main goal is the development of an application that monitors a population of individuals potentially infected with infectious diseases, globally. Although, some work has been developed in this direction, this application intends to innovate the technologies already used. With the use of GPS (Global Positioning System) and Wi-Fi (Wireless Fidelity), it will be possible to categorize users as being infected or at risk of contracting a disease.

A. Operating Mode

In the system's operation, there are three important entities: the Diagnostics Legitimation Server (SLD), the Diagnostics Publishing Server (SPD) and the users' application [1].

For the users' application to be able to work, Bluetooth and Google-Apple Exposure Notifications must be activated. The application is constantly listening and sending data. This data consists in pseudo-random keys which are generated on each mobile device. Every 10 minutes, a Rolling Proximity Identifier (RPI) is broadcast and received by all nearby devices. Each device locally stores the keys generated in the last 14 days and all RPIs received during those 14 days.

When a user gets a positive diagnosis of a disease, the date on which he first got the symptoms is asked. From that date, the doctor inserts the user into the system as being infected. The doctor then accesses the SLD, which generates two codes: a Legitimation Code (CL) and a Certificate of Access (CA). The CL is provided to the user. With CL, the user connects to the SLD and gets the CA. Through the CA, the application connects to the SPD and publishes on this server all of its keys that have been generated since the third day before the first symptoms appeared. In the SPD there are all the keys of all users who tested positive for the same disease, so it is necessary to access these keys in order to see if anyone else is at risk of contracting it. To do this, each application on each device communicates with the SPD twice a day. In each communication, the published keys are downloaded. At the end of download, a comparison is made between the saved received keys and the downloaded keys. If there is a coincidence, na evaluation is made whether the contact is at risk or not.

B. Existing Similar Applications

At the beginning of the pandemic, several projects were developed in order to create some type of monitoring so there could be a better control of the pandemic. Some of these projects have completed their final stage, the application. It is important that these applications comply with the General Data Protection Regulation.

Initially, StayAway COVID [2] was studied, which is the official application used in Portugal. This allowed a person to be alerted when exposed to risk contact. This application was based on several projects, specially *FollowMyHealth* and *monitorCovid19.pt*.

FollowMyHealth [4] is a project which aim was the improvement of the knowledge about transmission. With this knowledge, it would be possible to parameterize the virus transmission models, providing the user with a kind of self-surveillance indicator, through the evaluation and analysis of each location. The analysis takes into account the type of activities carried out in the place and the time that the users remained in the place. Therefore, it would be possible to alert users about possible contagions. This project also aimed to accompany users who were in prophylactic isolation.

monitorCovid19.pt [5] consisted in the development of an application capable of monitoring and tracking the movement of the population.

In Singapore, the official application is called *TraceTogether* [6]. In this application, the exchange of information is done with the users' consent. The difference between this application and *StayAway COVID* is the number of days that are used to calculate the infection risk period. Here, 21 days are used, as opposed to 14 days.

Inspired by *TraceTogether*, an European COVID-19 control system was born, the *Pan-European Privacy-Preserving Proximity Tracing* [7]. This contact-tracing system's main objective is to track proximity contacts, informing potentially exposed people.

Other similar applications have been created in other countries, such as *COVIDSafe* [8] [9], in Australia, and *SwissCovid* [10] [11], in Switzerland. In all of them, the way they work is similar.

C. Technologies with Potential to be Used

The technology used in existing applications is Bluetooth Low Energy (BLE) [12] [13] [14]. Comparing the classic Bluetooth with the BLE, there are some differences in terms of consumption and the amount of information exchanged. Bluetooth is ideal for transfering large amounts of information. Transfering a lot of information requires using a lot of energy to do so. In contrast, BLE is used in applications that do not require the transfer of large amounts of information and, therefore, collect little information for the same communication range. Since there is less energy consumption, one of the advantages of the BLE is the battery life, which can last several days. BLE has several modes of operation, if working in pairs. There are diffuser and observer modes, in which diffuser mode devices only send data and observer mode devices only receive. In parallel, there are the central and peripheral modes. In these modes, devices in central mode only communicate with devices in peripheral mode, both to receive and to send data. Two devices connected in peripheral mode cannot communicate with each other.

The Google-Apple Exposure Notifications [15] [16] emerged as a joint operation between Google and Apple, in the context of the pandemic, to try to solve inumerous issues about user privacy and security. It is a decentralized system, based on BLE, which is implemented at the operating system level. BLE is used in this system to be able to send messages to nearby devices. In terms of security, GAEN is designed so that, every 10 minutes, the MAC address of Bluetooth changes. It is important to emphasize that the user is free to choose to receive the Exposure Notifications and that this system does not monitor the user's GPS location. It just records the relative position between two users, that is, whether they are close or not.

During the pseudo-random key creation process, some encryption standards are used. Initially, an authentication code is calculated based on a hash. Any hash function can be used in calculating the authentication code. In the case of the application, the hash function used is SHA-256 [17]. To encrypt data the Advanced Encryption Standard is used [18].

In the application to be developed, the GPS [19] functionality that mobile devices have will be used. Through GPS, it is possible to know the location and time information of its user. To determine all this information, it is necessary to have four satellites. GPS is the American tracking system. In addition to this system, there are other location systems: the European system, Galileo [21]; the Chinese system, BeiDou [20]; and the Russian system, Glonass [22].

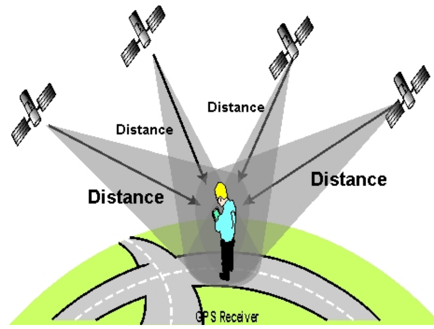

Fig. 1. Operating Mode of GPS

In addition to knowing the user's exact location, during the application's proper functioning it is also necessary to know some information about Wi-Fi. First, it should be noted that being connected to a Wi-Fi network does not imply having access to the Internet. Several machines may be connected to each other through Wi-Fi network and the network may not be connected to the internet. To be able to connect to a Wi-Fi network, the device must be within a range of an access point. The access point transmits a wireless signal that covers a range of 100 to 300 meters. Each access point has some parameters of interest. One of them is the name of the network, which is called the Service Set Identifier (SSID). For a user to connect, it will be necessary for the SSID of that network to appear in a network of available lists, if the network is visible. Otherwise, in case the network is in hidden mode, it will be necessary to know the SSID of the network.

During the operation of the system to be developed, it is important to have communication between the various components of the system. For this, it is necessary to establish standards for information exchange, that is, to pre-define which communication protocols to use [23] [24]. There are two very common communication protocols: Transmission Control Protocol (TCP) and User Datagram Protocol (UDP). These two protocols have some differences in their main points. In TCP, it is necessary to have a previous connection so that data can be exchanged. Submitted data is delivered sequentially in the order of submission. Therefore, there is a guarantee of data delivery to the receiver, with data retransmission mechanisms for when there are losses. As this is a connection-oriented protocol, data broadcast is not allowed. For UDP communication, it is not necessary to establish any connection for data exchange. In the absence of a connection, there is no guarantee of sequential data delivery to the receiver, as no retransmission of the data is possible. Unlike TCP, the UDP protocol allows data broadcast.

In the system that will be developed, there is an HTTP server responsible for processing all requests coming from

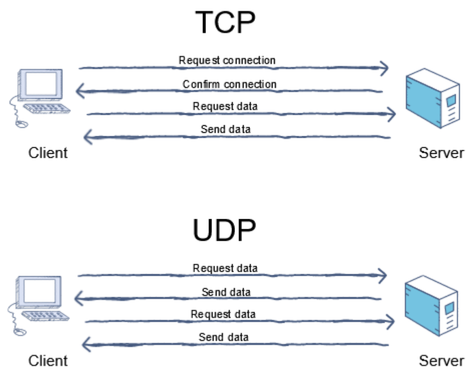

Fig. 2. Transferring Data in TCP and UDP

browsers [25]. The HTTP server is responsible for responding to these requests following a question-answer pattern, standardized by the REST-API. Requests coming from browsers can be of several types: GET, POST, PUT and DELETE. Upon receiving these requests, the HTTP server validates the data received, processes them and communicates any changes to the databases. Finally, the success or failure of orders is answered to the customer.

II. System Architecture

In order to create a system capable of monitoring a population of individuals potentially infected with infectious diseases, it is necessary to create an application that can be easily downloaded to mobile devices, without interfering with the normal use of said devices. It is in the applications that some information that needs to be saved is obtained. For such, a server capable of receiving, processing and storing these data collected by the application was created. Finally, it was necessary to create a website where it is possible to see, in real time, all the data being collected by the applications.

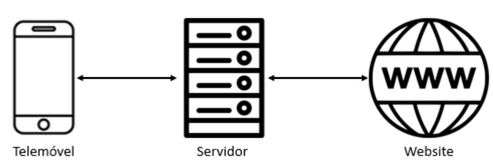

Fig. 3. System Architecture

A. Mobile Application

The application for mobile devices was developed using Android Studio and the Java programming language. This application's main function is to collect information using GPS and Wi-Fi. With GPS, the coordinates of the location of the device are obteined and, with Wi-Fi, i tis possible to analyze which ones are available on Wi-Fi networks.

This application has only two pages: a login page and another data collection page. The login page is where the user logs in, by filling out a form where they insert their username and password. If the user does not have a session yet, it is also possible to create one. The data collection page is responsible for collecting all data related to GPS and Wi-Fi. On this page, a timer is implemented in which every 15 seconds performs two functions capable of obtaining the desired data – both the GPS coordinates obtained, as well as the names of the access points available for the device to connect, are shown, in text form, on the page itself. Obtaining GPS and Wi-Fi logs is also possible by pressing the buttons available on the screen.

All these interactions are only possible when there is constant communication with the server, with message exchanges, using the UDP communication protocol. On the login page, at the end of inserting the two requested parameters, the application generates a UDP message to the server, so that the server checks if the user's data is confirmed. Once confirmed, another UDP message is sent to the application, containing a success/error code. On the data collection page, only UDP messages are sent to the server, each containing information about GPS coordinates or Wi-Fi hotspot names.

B. Server

The server is the fundamental part of the system. It is where all data is received, processed and stored.

This server is subdivided into several components, namely UDP servers, databases and an HTTP server. UDP servers are responsible for communicating with mobile device applications. These receive data from the applications and insert them into databases. Databases are responsible for storing all data. There are three different databases: one responsible for data relating to users, another responsible for storing the records of GPS coordinates and, finally, one for storing all records of available Wi-Fi networks. Finally, there is the HTTP server. This server is responsible for responding to all requests coming from the browser. When it receives a request, this server analyzes it and contacts the corresponding database. After this contact, the response to the request is sent to the browser. This HTTP server is implemented as a reverse proxy. In a reverse proxy, all requests for information are found on the internet and the responses to those requests are found on servers.

C. Site

The website is accessible to all users who need to interact with the system. It consists of several pages.

Initially there are administration pages. These pages are responsible for showing the content of the databases.

There is a page dedicated to health professionals, where users who test positive for the disease are included.

Finally, there are two pages for statistical purposes. In one of them, there is a map showing each place where the positive users have been, giving a spatial notion of frequented places. On the other page are some statistics of available Wi-Fi networks. In each Wi-Fi network, the number of occurrences

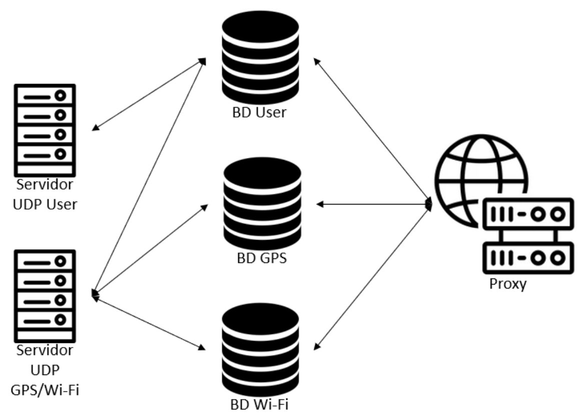

Fig. 4. Server's Internal Organization

of positive users who were in contact with that network are counted.

III. System Testing

After implementing the entire system, it is necessary to test it to detect errors and make the necessary improvements.

To this end, three users were created. For each user, routes were also created to collect navigation data from each of these users. The routes were designed in such a way so that only the second user would cross the other two users.

Then, it was simulated, for each user alone, what happens when they are inserted as positive for the disease.

It was concluded that the use of GPS proves to be effective. The calculation of distances using GPS coordinates is done in a correct and precise way, since, between each user, there is the same number of risky contacts by GPS. Regarding data obtained through Wi-Fi, the results are also positive. The calculation of risky contacts via Wi-Fi is done by comparing each user's SSID. In this part, something unexpected happens. There are Wi-Fi networks in common between users who are not geographically close. This phenomenon reveals a system flaw as this system did not foresee that two distinct and distant Wi-Fi networks can have the same SSID.

IV. System Analysis

Through implementation and later testing of the system, it was possible to identify some aspects that could be improved.

A. Analysis of UDP Servers

The first limitation regarding these servers is the fact that these servers could only run on one thread, that is, while they are processing a message, they cannot be receiving the next message, there being a waiting period. In case there are few users present in the system, this limitation does not apply. However, this system was designed to accommodate as many users as possible, and this time may be limiting of the system. One solution presented is the decentralization of this server. By creating multiple server replicas, the server would be redundant, implying that if any server fails or is unavailable, packets would not be lost.

B. HTTP Server Analysis

As with UDP servers, the more users there are, the more requests the HTTP server receives. When receiving more requests, the server may experience some slowdown and overload. Therefore, the ideal would be to create replicas of this server. Another improvement would be the implementation of a memory cache on the server. Thus, the number of times the database was contacted would be reduced, causing less system slowdown.

V. Final Considerations

A. Future Aspects

The system that was designed is implemented. However, there are some aspects susceptible to improvement in the next phase of development:

- Improved application activity status;
- Application functioning in the background;
- Optimization of obtaining GPS coordinates;
- Improvement of the user login page;
- Use of encryption for username and password;
- Application correction when there is user communication;
- Transferring navigation data to the server;
- Website always available online;
- Improvement of the website's front-end;
- Creation of an administration area;
- Improved map;
- Communication of risky contacts.

B. Conclusions

This work was a demanding job, both in terms of planning and in terms of design.

Initially, a lot of research was required about how to develop this system, as well as the technologies to use.

During the development of the system, several errors arose which ended up delaying the evolution of the system itself.

Finally, with the testing of the system, it was found that the entire system was working well, and without errors, producing the expected results.

References

[1] Funcionamento das aplicações: https://pplware.sapo.pt/internet/covid-19-afinal-como-funcionam-as-solucoes-para-rastrear-contactos/
[2] StayAway Covid: https://stayawaycovid.pt/
[3] DP-3T: https://en.wikipedia.org/wiki/Decentralized_Privacy-Preserving_Proximity_Tracing
[4] FollowMyHealth: https://pplware.sapo.pt/informacao/followmyhealth-app-portuguesa-ja-disponivel-para-tracing-a-covid-19/
[5] monitorCovid19.pt: https://pplware.sapo.pt/smartphones-tablets/monitorcovid19-pt-a-app-portuguesa-para-rastrear-contagios-de-covid-19/
[6] TraceTogether: https://www.tracetogether.gov.sg/
[7] PEPP-PT: https://pplware.sapo.pt/informacao/pepp-pt-app-para-monitorizar-movimentos-dos-cidadaos-por-causa-da-covid-19/
[8] COVIDSafe: https://pplware.sapo.pt/smartphones-tablets/covidsafe-australia-lanca-uma-polemica-aplicacao-para-rastreio-da-doenca/
[9] COVIDSafe: https://www.health.gov.au/resources/apps-and-tools/covidsafe-app#about-the-app
[10] Blueooth Low Energy: https://play.google.com/store/apps/details?id=ch.admin.bag.dp3t
[11] Blueooth Low Energy: https://en.wikipedia.org/wiki/SwissCovid

[12] Blueooth Low Energy: https://pt.wikipedia.org/wiki/Bluetooth_Low_Energy#cite_note-2
[13] Blueooth Low Energy: https://developer.android.com/guide/topics/connectivity/bluetooth-le
[14] Blueooth Low Energy: https://elainnovation.com/what-is-ble.html
[15] Exposure Notifications: https://www.google.com/covid19/exposurenotifications/
[16] Exposure Notifications: https://en.wikipedia.org/wiki/Exposure_Notification
[17] HMAC-SHA-256: https://pt.wikipedia.org/wiki/HMAC
[18] AES: https://pt.wikipedia.org/wiki/Advanced_Encryption_Standard
[19] GPS: https://pt.wikipedia.org/wiki/Sistema_de_posicionamento_global
[20] Beidou: http://en.beidou.gov.cn/
[21] Galileo: https://www.euspa.europa.eu/
[22] GLONASS: https://www.glonass-iac.ru/
[23] Protocolos de Comunicação: https://www.lifesize.com/en/blog/tcp-vs-udp/
[24] Protocolos de Comunicação: https://www.geeksforgeeks.org/differences-between-tcp-and-udp/
[25] HTTP Server: https://www.welivesecurity.com/br/2019/12/20/o-que-e-um-proxy-e-para-que-serve/