ANDROID APPLICATION FOR USER'S REAL-TIME INFORMATION REGARDING THE POSIBILITY OF BEING CONTACT TO A COVID-19 INFECTED PERSON

Dorina Sava [1]

Mădălina Cărbureanu [2]

[1] **Romania**
[2] Petroleum-Gas University of Ploiesti, **Romania**
e-mail: mcarbureanu@upg-ploiesti.ro

DOI: 10.51865/JPGT.2023.01.01

ABSTRACT

Soon after the pandemic inflicted by Covid-19 was declared, different organizations and smart application development companies suggested that artificial intelligence could be an useful tool in helping the medical staff monitoring the virus spread for a better understanding of its behavior. This paper presents a smart application CovidWatch developed for users real-time warning regarding the possibility of contracting the Sars-CoV-2 virus from an infected person. For this purpose, the developed application uses the Bluetooth module of the user's device in order to check the social distance and to determine the possible contacts. The developed application has a number of four modules (the expert system (ES), the data retrieval, the real-time information and warning and the interface), that work on Android devices. Using the artificial intelligence techniques (knowledge-based systems), the application has a knowledge base populated with user's data, the user's time of contact and distance and the users identified as being contacts. If a user warns the android application that he has been infected with Covid-19, it further triggers the expert system inference engine, allowing the knowledge to be inferred. Based on the implemented rules, the inference engine analyses the data regarding the users that were in contact with an infected person. If those meet the conditions taken into consideration for declaring a person as being contact (a certain distance and time) they receive in real time a warning message and some advice regarding how it is recommended to act. The main advantage of android application is that it provides real-time warning regarding the possibility or probability of being infected with Covid-19, warning that can be deactivated only after the user has confirmed that the took note of this. A set of CovidWatch application simulation results is also presented taking into consideration different scenarios.

Keywords: Covid-19, Android application, real-time warning, artificial intelligence, expert system.

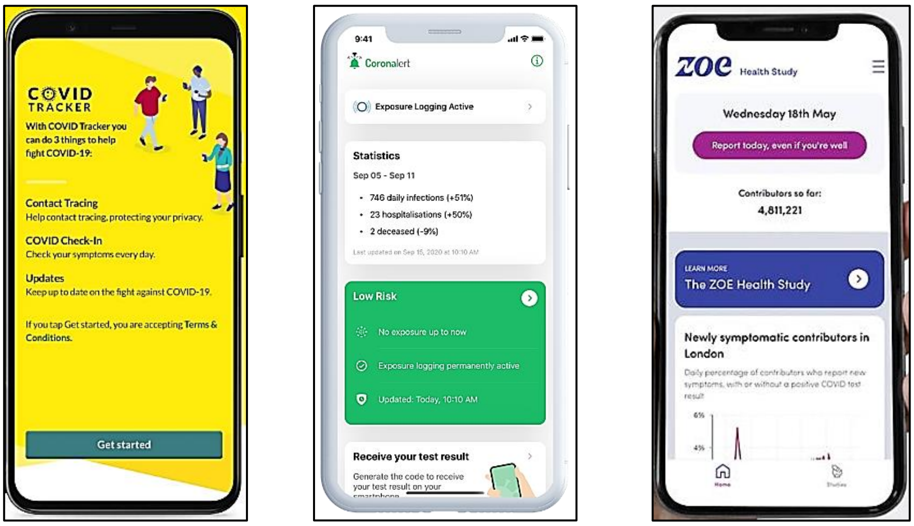
INTRODUCTION

The Covid-19 pandemic had a strong global impact and a negative influence on the health system from all over the world. Since the declaration of the Covid-19 pandemic, around 500 million people from almost 200 countries have been infected and more than six million deaths have been recorded. Since the first cases, efforts have been made to study the virus in order to stop or slow its effects until a cure was found [6, 7, 8].

The length of time a person is contagious (the virus spreads through the air as small liquid particles by an infected person when they cough, sneeze, talk, sing or breathe) can vary from person to person. Those who develop more severe forms of the disease are usually contagious for a longer period of time. It is estimated that in at least five days after the onset of symptoms the risk of infecting other people is at its peak [10]. It has been shown that people infected with the Sars-CoV-2 virus can be contagious two to three days before the onset of symptoms, with the risk increasing by one to two days before the infected person begins to feel ill and that asymptomatic people are contagious two days before a test is positive [11].

According [12], a person who lives in the same household with a Covid-19 infected patient, has who had face-to-face contact with a Covid-19 positive person at a distance of no more than two meters and for more than 15 minutes, or has been in the same room with a Covid-19 case at a distance of less than two meters for at least 15 minutes is considered as a direct contact for a Covid-19-infected person.

In order to assist the health system in studying and collecting information regarding the Sars-CoV-2 virus behavior, but also in order to prevent its spread, some states have developed, in collaboration with organizations and IT companies, various applications and tools based on artificial intelligence (AI) which were implemented locally and offered for citizen's use. Such applications and systems developed for user's information and warning about the risk of Sars-Cov-2 infection are COVID Tracker (developed at the initiative of the Irish government; requires a minimum version of Android 6.0), Coronalert (developed in Belgium and uses Bluetooth technology) and COVID Symptom Study (launched by medical science company ZOE in late March 2020 for tracking the evolution of Covid-19 in the United Kingdom of Great Britain, USA and Sweden.) (Fig. 1) [13, 14, 15].



COVID TRACKER

With COVID Tracker you can do 3 things to help fight COVID-19:

Contact Tracing
Help contact tracing, protecting your privacy.

COVID Check-In
Check your symptoms every day.

Updates
Keep up to date on the fight against COVID-19.

If you tap Get started, you are accepting Terms & Conditions.

Get started

9:41

Coronalert

Exposure Logging Active

Statistics
Sep 05 - Sep 11

- 746 daily infections (+51%)
- 23 hospitalisations (+50%)
- 2 deceased (-9%)

Last updated on Sep 15, 2020 at 10:10 AM

Low Risk

No exposure up to now

Exposure logging permanently active

Updated: Today, 10:10 AM

Receive your test result
Generate the code to receive your test result on your smartphone

ZOE Health Study

Wednesday 18th May

Report today, even if you're well

Contributors so far:
4,811,221

LEARN MORE
The ZOE Health Study

Newly symptomatic contributors in London
Daily percentage of contributors who report new symptoms, with or without a positive COVID test result

Figure 1. *Applications for users information and warning about the risk of Sars-Cov-2 infection*

The best solutions are searched for the development of an application or system capable of ensuring the real-time information of the proximity of a person infected with Covid-19. AI techniques identified as likely to be used in this kind of applications include: knowledge-based systems (KBS), respectively expert systems (ES), machine learning techniques, data mining, genetic algorithms and intelligent agent-based techniques.

THE CovidWatch INTELIGENT APPLICATION DEVELOPMENT

The CovidWatch (developed using PHP programming language [1], the MySQL database management system, and the web application integrated development environment MIT App Inventor) android application was implemented using one of the AI techniques, respectively the KBS (ES).

As can be seen in the application block diagram presented in Fig. 2, four main modules were developed and integrated with each other, such as: the expert system (ES) module, the data retrieval module, the real-time information and warning module, and the interface module.

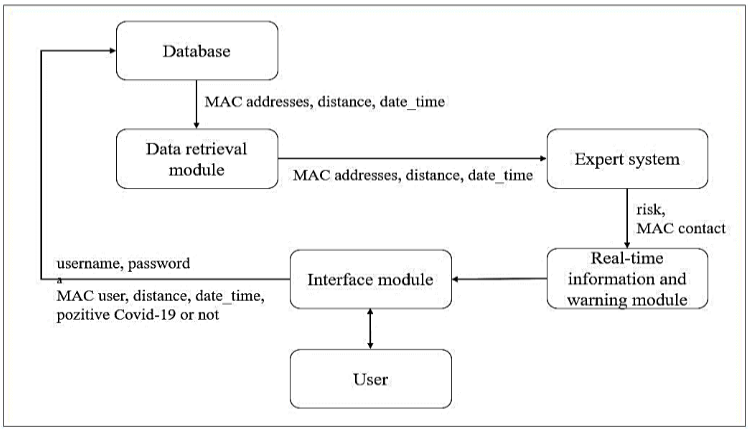

Database
MAC addresses, distance, date_time
Data retrieval module
MAC addresses, distance, date_time
Expert system
risk,
MAC contact
Real-time information and warning module
Interface module
username, password
a
MAC user, distance, date_time,
pozitive Covid-19 or not
User

***Figure 2**. Block diagram for CovidWatch Android application*

The *expert system* (ES) is the central element of the application as it analyses the retrieved data and provides certain results based on them. This module consists of two submodules: the knowledge base (KB) and the inference engine (IE) [4].

The *ES module* has been implemented through two main scripts: *insert_contact.php* (for ES knowledge base), which mediates the populating of the database with information needed to determine possible contacts, and *determine_contact.php* (for ES inference engine), which takes the data from the database for analysis and in which are chained the rules used by the IE to determine possible contacts.

The IE calls the data retrieval module via functions implemented in the *determine_contact.php* script. For each MAC address of possible contacts, is calculated the distance, the proximity duration, and the number of days since contact. IE will chain the results into a set of rules for determining the contacts and the risk of infection. The data obtained will be recorded in the KB.

To establish the risk of infection, the IE performs the following actions:

- retrieve, using the data retrieval module, the MAC addresses of the possible contacts, the distance and the time when the contact was recorded;
- calls the KB for each record to determine for each MAC address the duration, the distance of the contact, and the number of days since the contact;
- based on the values obtained, determines whether a user has been a contact and the associated degree of risk (red - high probability of contact; yellow - possibility of contact, green - low possibility of contact);
- if a user is assigned a red or yellow risk rating, the result is retrieved and entered into the KB in the contacts table "*MAC*" as: MAC address corresponding to the contact and risk rating.

The rules used by the IE are applied for each possible contact to determine the degree of risk. The main rules take into account the *contagious* parameter which represents the number of days passed from the contact until one of the users declared itself as positive Covid-19. According to the information provided by official sources, three possible cases were considered [2, 3, 5, 9, 11]:

- *contagious* < 2 (less than two days have passed since contact): the probability of being contagious is very high;
- *contagious* is between 2 and 3 days: the probability is moderate;
- *contagious* > 3: probability is low.

The sub-rules take into account the *distance* and *duration* parameters previously determined in the script by analyzing the data from the KB. Thus, the *distance* parameter has three possible values: *red* (the distance between devices is less than two meters), *yellow* (the distance is between two and four meters), *green* (the distance is between four and six meters). The *duration* parameter has two possible cases: the contact took place for more than 15 minutes, or the contact took place for less than 15 minutes. All parameters were considered to be equally important, so their impact in determining the contact and the degree of risk is equal. A main rule example and its sub-rules is presented next:

```
 if($contagious[$k] < 2)
 {if($distance[$k]=="red" and $duration[$k]>=15) {$risk="red"; $k = $r;}
   else if($distance[$k]=="red" and $duration[$k]<15) {$risk="yellow";}
          else if($distance[$k]=="yellow" and $duration[$k]>=15) {$risk="yellow";}
          else if($distance[$k]=="yellow" and $duration[$k]<15) {$risk="yellow"; $k=$r;}
          else if($distance[$k]=="green" and $duration[$k]>=15) {$risk="yellow";}
          else if($distance[$k]=="green" and $duration[$k]<15) {$risk="green";}
 }
```

If the risk rating is assigned the values *'red'* or *'yellow'*, the user is considered to be a probable or possible contact and the risk rating and MAC address of the contact are recorded in the table "*MAC*" as follows:

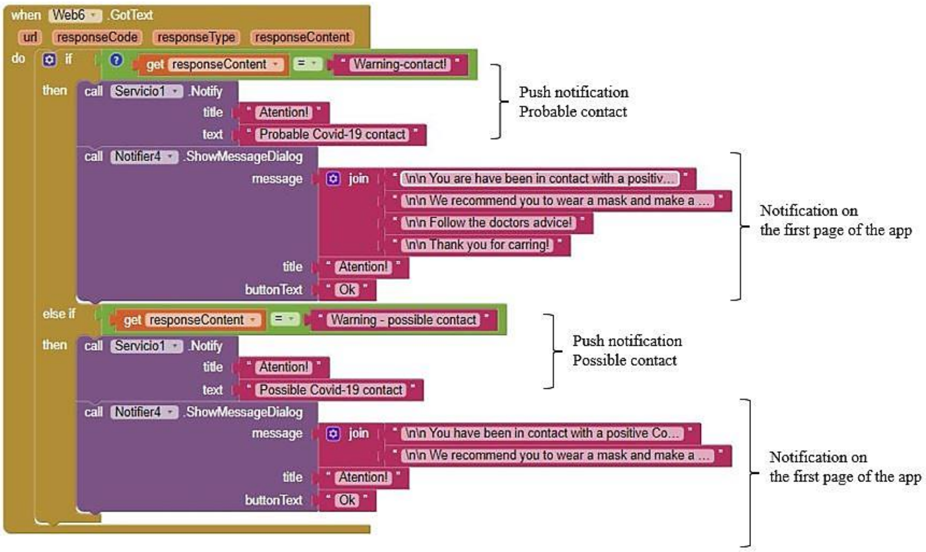
if ($risk==="red" or $risk==="yellow") {$insert = mysqli_query($dbc, "INSERT INTO MAC (mac, risk) VALUES ('$mac[$i]', '$risk')"); }

The *database retrieval/ extraction module* extracts in the "*positive*" table (Table 1) the MAC address of the positively identified Covid-19 persons (*mac_user*), the MAC addresses of the users with whom they came into contact (*mac_contact*), and all the records regarding the distance (*distance*) and the time they met (*date_time*). The data is further analyzed by the IE following the steps previously presented, in order to establish the risk of infection.

The *real-time user notification and warning module* was implemented based on the results of the IE rule walkthrough, using the MIT App Inventor platform and the PHP programming language via the *notification_contact.php* script which checks if the user's MAC is in the table (Table 1). If true, then if the MAC has attributed the code '*red*', the '*Warning-contact'* message is sent to the application, and if the MAC has attributed the '*yellow*' codes, the message *'Warning- possible contact'* shall be returned.

Afterwards, the message is retrieved in the MIT App Inventor blocks and a push notification is displayed on the main page of the application and on the main screen of the phone with a message corresponding to the user's case (red risk or yellow risk) (Fig. 3). The display of push alert/warning messages is done using the "service" extension available for free on the MIT App Inventor forum.

The notification is closed only after the user has confirmed acknowledgement by pressing the "Ok" button in the notification. Otherwise, the notification will appear every one to two minutes.



```
when Web6 .GotText
 url  responseCode  responseType  responseContent
do  if  get responseContent = " Warning-contact! "
    then  call Servicio1 .Notify
                         title " Atention! "
                          text " Probable Covid-19 contact "
          call Notifier4 .ShowMessageDialog
                    message  join " \n\n You are have been in contact with a positiv... "
                                  " \n\n We recommend you to wear a mask and make a ... "
                                  " \n\n Follow the doctors advice! "
                                  " \n\n Thank you for carring! "
                      title " Atention! "
                 buttonText " Ok "
    else if  get responseContent = " Warning - possible contact "
    then  call Servicio1 .Notify
                         title " Atention! "
                          text " Possible Covid-19 contact "
          call Notifier4 .ShowMessageDialog
                    message  join " \n\n You have been in contact with a positive Co... "
                                  " \n\n We recommend you to wear a mask and make a ... "
                      title " Atention! "
                 buttonText " Ok "
```

Push notification
Probable contact

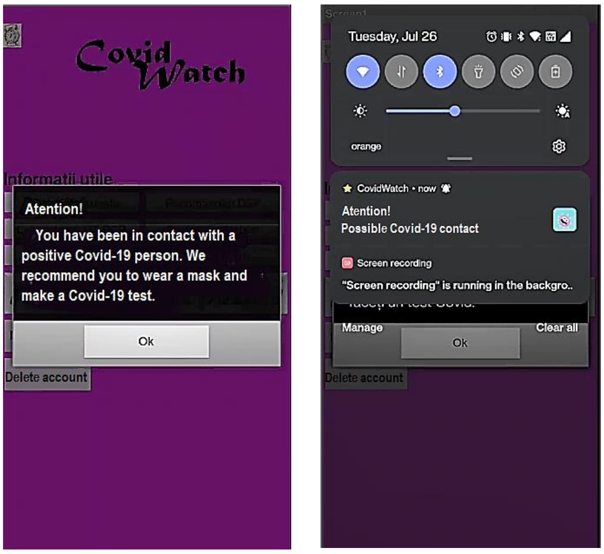
Notification on
the first page of the app

Push notification
Possible contact

Notification on
the first page of the app

Figure 3. *Notification blocks in MIT App Inventor*

The *interface module* (Fig. 4), built in the MIT App Inventor development environment, acts as a user-friendly link between the user and the elements behind the application. The CovidWatch includes guides to fill in data when registering the user, data about the application and terms of use of the application, as well as suggestive titles about the information that is made available to the user.



Atention!

You have been in contact with a positive Covid-19 person. We recommend you to wear a mask and make a Covid-19 test.

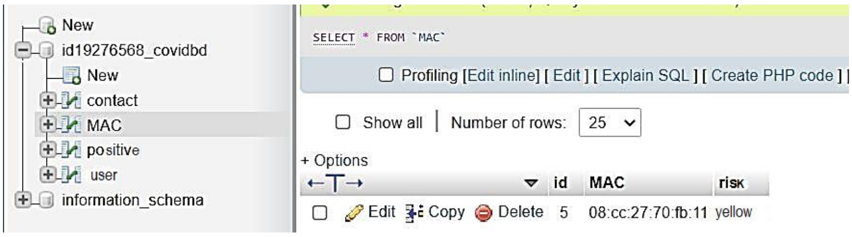
Atention!
Possible Covid-19 contact

Screen recording

"Screen recording" is running in the backgro..

Figure 4. *CovidWatch user-friendy interface*

The functionality of the application can be described as follows: when registering, the user will enter a username, the MAC address of the Bluetooth device and an account password. The information will be saved in the table "*user*" that populates the CovidWatch database, after the user has read and agreed to the application usage terms (Fig. 5). The user's own MAC address will also be stored locally (on the phone) and used by the application in order to identify the unique user. After registration, the user will be asked to log in into the application. The username - password association will be validated. If the data entered is correct, the application's main page will be opened and it will start periodically scanning for Bluetooth devices and connecting the Bluetooth module to devices whose MAC address is found in the database, in the user's table. If the connection has been made, the distance (in the form of a pre-defined code of colors), the time of registration and the MAC addresses of the connected devices are recorded in the application database.



```
SELECT * FROM `MAC`
```

	id	MAC	risk
□ Edit Copy Delete	5	08:cc:27:70:fb:11	yellow

Figure 5*. Tables that populates the CovidWatch database*

The distance parameter is determined from the relationship between the Measured power (Tx $[\mathrm{dBm}]_{\mathrm{SI}}$), the Received Signal Strength Indicator (RSSI $[\mathrm{dBm}]_{\mathrm{SI}}$) of the Bluetooth Low Energy (BLE) module of the device and the medium for which it is calculated (represented by the constant n) (equation 1). The value obtained is an estimated one and errors may occur depending on the environment and obstacles between devices. For this reason, in the application, the distance has been fitted between two experimentally determined values in a closed environment. If the distance is less than six meters, the data (MAC addresses, distance and time) is introduces into the database.

$$distance = 10^{\wedge}\left((T_x - RSSI)\right)/(10 * n)) \qquad (1)$$

Due to external factors influencing radio waves (absorption, interference, diffraction) RSSI tends to fluctuate. The greater the distance between devices, the more unstable the RSSI. T_x is a factory-calibrated constant indicating the expected RSSI at a distance of one meter from the transmitter. The constant *n* (environment factor) is used to express the sensitivity of the transmitter and has values between 2 and 4. The denser the area, the higher the value of the environmental factor should be. In the application, *n* was considered to have the value 4.

In the Covid Watch application, were taken into account one meter error between the measured and calculated distance value, based on measurements made under different conditions (indoor, outdoor, orientation of devices towards each other, possible obstacles between devices).

Information older than 15 days will be deleted from the database to avoid overcrowding. The application uses the time indicated on the user's device. If a user is Covid-19 positive, he can notify the application, releasing another chain reaction: the application will process the data in the knowledge base regarding the persons he was near using a set of rules thus further determine possible contacts. Those identified as contacts will be notified by a pop-up message that will appear on the main page of the app, but also by a notification, in case the screen is locked. The notification disappears only after the user has become aware of it within the application, and the data will be deleted from the database.

Using CovidWatch requires permissions to access Bluetooth, internet connection and location, the latter being a security measure imposed by Android. The application can work in the background and with the device screen turned off. The application performs the following main tasks: initialization of an interface according to script results, display of information messages about the completion of some actions, display of information messages about required permissions, how to use the application and the purpose for which it was created, display of notifications in case the user is a possible or probable contact.

If a Bluetooth Low Energy (BLE) connection between the devices has been established and the calculated distance between the devices is within the required limits (maximum six meters between devices), the distance values along with the time values are entered into the database and the devices are disconnected. The process repeats, periodically at approximately one-minute intervals, for each MAC address from the list created by interrogating the ”*user*” table. Along with the *read_mac.php* script, the *notify_contact.php* script will also be accessed, searching if the device is identified as contact with a Covid-19 positive user. If a user is infected with the Sars-CoV-2 virus, he can notify the application by selecting the "*I have been tested positive*" message. The user will be asked to validate the information. After validation, he will be directed to a new page with advice on what he should do and a link to the Public Health Directorate page. At the same time, the script *determine_contact.php* will be accessed which will identify the possible contacts and the associated risk level and enter the results in the *MAC* table in the database. If, after periodically running the *notify_contact.php* script, the MAC address of any user is found in the *MAC* table, then the user will be notified as possible or probable contact accordingly.

In order to demonstrate the functionality of the application, a series of simulations with a progressive degree of difficulty were carried out of which two are presented in the present paper. The first simulation aims to implement the following cases of the parameters used to determine the contact quality:

- *contagious* < 2 (less than two days have passed from contact registration to reporting of Covid-19 illness);
- *duration* < 15 (devices have been in proximity for less than 15 minutes);
- *distance* = red (devices were less than two meters apart).

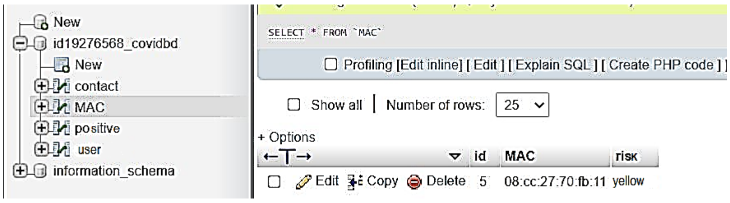
Based on the set of rules used by the IE, the case falls under the first main rule, sub-rule two, the source code of the *determination_contact.php* (selection) rules being presented below. Thus, the degree of risk recorded is *yellow* (possibility of Covid 19 contact).

```
if($contact[$k] < 2)
 {if($distance[$k]==="red" and $duration[$k]>=15) {$risc="red"; $k = $r;}
    else if($distance[$k]=="red" and $duration[$k]<15) {$risc="yellow";}
```

Two devices were used for the simulation: the OnePlus8T/ Android12 with Bluetooth MAC address d0:49:7c:375d:dc, considered the device of the Covid-19 positive user, and the MotoG7/ Android10, with Bluetooth MAC address 08:cc:27:70:fb:11, considered the device of the contact. In table 1 are presented the values extracted from the "*positive*" table (table created after interrogating the database for determining the relation between the Covid-19 infected user and its contacts) created in the application database. It can be observed that the contact duration is about eight minutes and the risk level is red (high probability of contact). The simulation was performed on 16.07.2022, the contagious parameter having the value zero. The MAC address of the MotoG7 device was registered in the "*MAC*" table with a yellow (possibility of contact) risk degree (Fig. 6) and the user received, after querying the "*MAC*" table, a notification of possible contact (Fig. 7).

Table 1. *"Positive" table records for simulation 1*

No.	***id***	***mac_user***	***mac_contact***	***date_time***	***distance***
1	*381*	*d0:49:7c:37:5d:dc*	*08:cc:27:70:fb:11*	*2022-07-16 08:57:23*	*red*
2	*383*	*d0:49:7c:37:5d:dc*	*08:cc:27:70:fb:11*	*2022-07-16 08:59:01*	*red*
3	*385*	*d0:49:7c:37:5d:dc*	*08:cc:27:70:fb:11*	*2022-07-16 09:00:51*	*red*
4	*387*	*d0:49:7c:37:5d:dc*	*08:cc:27:70:fb:11*	*2022-07-16 09:02:15*	*red*
5	*389*	*d0:49:7c:37:5d:dc*	*08:cc:27:70:fb:11*	*2022-07-16 09:03:39*	*red*



```
SELECT * FROM `MAC`
```

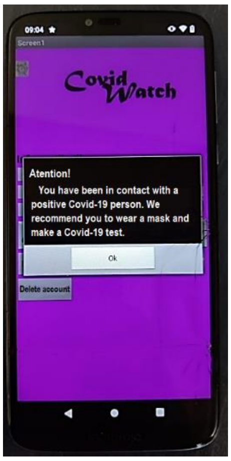
□ Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [

□ Show all | Number of rows: 25

+ Options

	id	MAC	risk
□ Edit Copy Delete	5	08:cc:27:70:fb:11	yellow

Figure 6. *MAC record and risk grade corresponding to simulation 1 in the "MAC" table*



Atention!
You have been in contact with a positive Covid-19 person. We recommend you to wear a mask and make a Covid-19 test.

Figure 7*. Possible contact alert*

The second simulation aims to implement the following cases of the parameters used to determine the contact quality:

- contagious < 2 (less than two days have passed from contact registration to reporting of Covid-19 illness);
- duration > 15 (devices were in proximity for more than 15 minutes);
- distance = red (devices were less than two metres apart).

Based on the set of rules used by the IE, the case falls under the first main rule, the first sub-rule, the source code (selection of the *determination_contact.php* source code) being presented below. Thus, the risk degree recorded is red (high probability of contact).

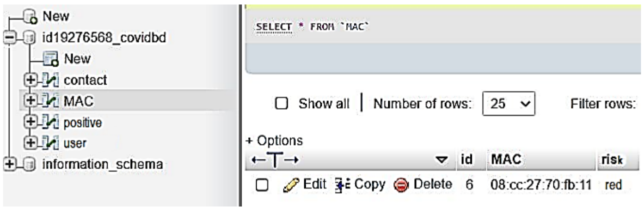
```
if($contact[$k] < 2)
    {if($distance[$k]==="red" and $duration[$k]>=15) {$risc="red"; $k = $r;}
```

For the simulation were used the same devices as in the first simulation. Table 2 shows the values extracted from the "*positive*" table of the corresponding simulation database. It can be seen that the contact duration is about 24 minutes and the risk degree is red. The simulation was performed on 16.07.2022, the contagious parameter having the value zero. The MAC address of the MotoG7 device was thus recorded in the "*MAC*" table with risk degree red (fig. 8), and the user received, after querying the "*MAC*" table, the notification of probable contact (fig. 9).

Table 2*. 'Positive' table entries for simulation 2*

No.	***id***	***mac_user***	***mac_contact***	***date_time***	***distance***
1	*393*	*d0:49:7c:37:5d:dc*	*08:cc:27:70:fb:11*	*2022-07-16 09:18:15*	*red*
2	*395*	*d0:49:7c:37:5d:dc*	*08:cc:27:70:fb:11*	*2022-07-16 09:19:27*	*red*
3	*397*	*d0:49:7c:37:5d:dc*	*08:cc:27:70:fb:11*	*2022-07-16 09:20:38*	*red*
4	*399*	*d0:49:7c:37:5d:dc*	*08:cc:27:70:fb:11*	*2022-07-16 09:21:50*	*red*
5	*400*	*d0:49:7c:37:5d:dc*	*08:cc:27:70:fb:11*	*2022-07-16 09:24:22*	*red*

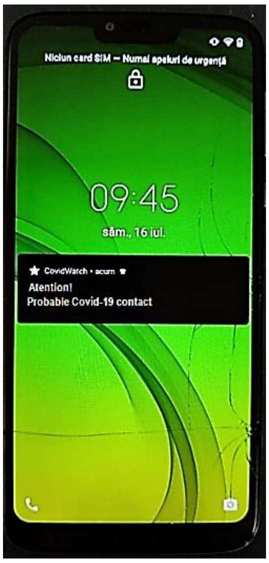
6	*401*	*d0:49:7c:37:5d:dc*	*08:cc:27:70:fb:11*	*2022-07-16 09:26:46*	*red*
7	*403*	*d0:49:7c:37:5d:dc*	*08:cc:27:70:fb:11*	*2022-07-16 09:27:57*	*red*
5	*405*	*d0:49:7c:37:5d:dc*	*08:cc:27:70:fb:11*	*2022-07-16 09:29:07*	*red*
8	*406*	*d0:49:7c:37:5d:dc*	*08:cc:27:70:fb:11*	*2022-07-16 09:31:49*	*red*
9	*408*	*d0:49:7c:37:5d:dc*	*08:cc:27:70:fb:11*	*2022-07-16 09:33:00*	*red*
10	*410*	*d0:49:7c:37:5d:dc*	*08:cc:27:70:fb:11*	*2022-07-16 09:34:10*	*red*
11	*411*	*d0:49:7c:37:5d:dc*	*08:cc:27:70:fb:11*	*2022-07-16 09:36:46*	*red*
12	*412*	*d0:49:7c:37:5d:dc*	*08:cc:27:70:fb:11*	*2022-07-16 09:37:57*	*red*
13	*413*	*d0:49:7c:37:5d:dc*	*08:cc:27:70:fb:11*	*2022-07-16 09:39:09*	*red*
14	*414*	*d0:49:7c:37:5d:dc*	*08:cc:27:70:fb:11*	*2022-07-16 09:42:45*	*red*



```
SELECT * FROM `MAC`
```

	id	MAC	risk
□	6	08:cc:27:70:fb:11	red

Figure 8. *MAC record and risk grade corresponding to simulation 2 in the "MAC" table*


Niciun card SIM – Numai apeluri de urgență

09:45

sâm., 16 iul.

CovidWatch • acum

Atention!
Probable Covid-19 contact

Figure 9. *Probable contact alert*

The simulations performed show the real-time behavior of the application, due to the fact that CovidWatch application display (during one to two minutes after the application is warned that a user has tested positive for Covid-19) the results under alert/warning messages form, depending on the input data. The functions implemented allow differentiation between contacts and individual analysis of each one of them in order to determine the degree of risk they fall into and subsequently inform them about the possibility or probability of being contact, in hope of slowing the spread of Covid19 virus.

CONCLUSION

With limited medical resource and growing healthcare pressure, the use of AI techniques could assist with diagnosis, prediction of prognosis, evaluation of epidemic trends, thus improving the efficiency of humans in fighting the Covid-19 pandemic. CovidWatch android application was created with the aim of contributing to this effort, highlighting the advantages of AI techniques usage in this field.

One important feature of the application is the fact that all the application functions used to determine the degree of risk of a user in relation to those around him (the MAC address list permanent update; the Bluetooth connection checking; the older data verification and deletion; the scanning for an available Bluetooth devices in the vicinity; the distance calculation between devices connected via Bluetooth, the analysis of the possibility for a user to be contact of a person infected with Covid19 virus and the possibility of warning them) are implemented in the background, without requiring user intervention.

Among the advantages of the application is its ability to be extended at national level and to be integrated into the Public Health Department system in order to create a record of infection outbreaks and Covid-19 evolution, with the potential of creating a tool that can be used and integrated both at the health system level and in other areas of research based on a target group study.

A possible drawback of the application is the instability of the Bluetooth connection which varies depending on the device security policies and instability on which it is installed and depending on the phone Bluetooth module.

A future direction is to develop the interface by adding new capabilities to make it more attractive and responsive to as many of the user's needs as possible. CovidWatch can also be adapted to interact with the user by conducting surveys analyzing the health status of individuals before, during and after Covid 19 infection. In this way, public health and research institutions are able to determine the evolution of the virus and its possible ramifications.

The authors main contributions are:

- Development of an expert system (KB and IE) to determine the contacts of a Covid-19 positive identified user;
- Development of rules for data analysis and determination of possible contacts;
- Development of a real-time user information and warning system;
- Implementation of a system for deleting information older than 15 days from the database;
- Development of a user-friendly interface using MIT App Inventor.

The developed CovidWatch application fulfils its main purpose, which is to inform users about the possibility of having been in the vicinity of a Covid-19 infected person. Once alerted, they can take timely action to avoid endangering their health and that of their loved ones.

REFERENCES

[1] Cărbureanu, M., Programare WEB – Ghid teoretic și practic, Editura Universității Petrol-Gaze din Ploiești, 2020.

[2] Narvaez, A., Guerra, J., Received signal strength indication – based COVID-19 mobile application to comply with social distancing using bluetooth signals from smartphones, Elsevier Public Health Emergency Collection, pp. 483-501, 2021.

[3] Neelavathy P.S, Geetha, A.V., Vasu, B., Jeevitha, V.K., Monitoring Social Distancing by Smart Phone App in the Effect of COVID-19, International Journal of Engineering Research & Technology (IJERT), vol 9/issue 9, pp. 946-953, 2020.

[4] Oprea, M., Nicoară, S., Inteligență artificială, Editura Universității Petrol-Gaze din Ploiești, 2005.

[5] Ramirez, R., Huang, C., Liao, C., Lin, P., Lin, H., Liang, S., A Practice of BLE RSSI Measurement for Indoor Positioning, Sensors (Basel), vol. 21/issue 15, pp. 1-24, 2021.

[6] Inter-Agency Standing Committee, Public Health and social measures for Covid-19 preparedness and response in low capacity and humanitarian settings, Version 1, pp. 1-28, 2020, available in https://reliefweb.int/report/world/interim-guidance-public-health-and-social-measures-covid-19-preparedness-and-response.

[7] World Health Organization, https://www.who.int/health-topics/coronavirus.

[8] World Health Organization, https://covid19.who.int/.

[9] Science, https://www.science.org/doi/10.1126/sciadv.abc1202.

[10] Region Varmland, https://www.regionvarmland.se/.

[11] Bioclinica, https://bioclinica.ro/pentru-pacienti/coronavirus-covid-19/.

[12] CNS, https://www.cnscbt.ro/index.php/.

[13] Covid Tracker app, https://www2.hse.ie/services/covid-tracker-app/.

[14] Corona alert, https://coronalert.be/en/.

[15] Health Study, https://health-study.joinzoe.com/.

Received: November 2022; Accepted: December 2022; Published: January 2023