CONTAINMENT ZONE ALERTING APPLICATION

PROJECT REPORT

Submitted by

R. VIGNESH (REG.NO:911719104084)

P.NAGARAJ (REG.NO:911719104037)

M.MANOJ (REG.NO:911719104030)

P.SEENIVASAN (REG.NO:911719104061)

ABSTRACT

The World Health Organization has classified the Covid-19 corona virus outbreak as a global pandemic. Lockdown and awareness (social distance, wearing of masks, etc.) among individuals are found to be the only ways to stop the community spread of this disease given the worrisome increase in affected cases around the world. Without widespread public awareness and proactive actions taken by the populace, it is exceedingly challenging to stop the communal transmission even during a lockdown in a highly populated nation like India. Recently, numerous containment zones across the nation were found and separated into red, orange, and green zones, respectively. The user can receive an alarm system from this programme if they search for or track any location, say they do so in an area that has been infected by the COVID-19 virus. The development of an python to alert users to the Covid-19 containment zones and deter trespassing there is the primary topic of this article. The positions of the places on a map that are designated as containment zones are updated by this web application. The application also alerts users to the online database if they enter a containment zone. Therefore, this application can be utilised as a tool to raise social awareness of the growing need for people to take preventative action.

1. INTRODUCTION

1.1 PROJECT OVERVIEW

Several research projects are currently underway across the nation to stop the increase in Covid-19 instances. Prior to its achievement in developing these kits, our nation used to import medical supplies like PPE (Personal Protection Kits) and masks. Our nation has taken steps to raise public awareness of the disease in addition to adopting actions to combat it. By educating the general public about the preventive steps that can shield them against infection, the news and media play a significant role in raising awareness of this issue. People's awareness of the importance of taking all preventive measures can greatly aid in limiting the spread of the virus. To stop the virus from spreading further, the nation has established containment zones throughout the cities where Covid-19 cases have been reported. To prevent contamination outside, these containment areas have been maintained closed off from the general public. The government has loosened several lockdown regulations after more than two months and allowed the reopening of government buildings, bus and other road transit facilities, and shopping centres. People can travel around the city for several reasons, including work. However, the containment zones are still maintained in isolation, and new containment zones are being established in all areas where Covid-19 cases have been reported. Despite the fact that police officers are stationed in these containment zones, it is still possible for anyone to wander inside without realising it. These containment zones present an infectious danger to city residents in this scenario where people freely roam about the metropolis. Because of this, letting people know where the containment zones are can help them get around and avoid them, lowering the likelihood of community transmission.

1.2 PURPOSE

This tracking application existing is manual system. This is a paperless work. It can be monitored and controlled. It reduces the man power required. It provides accurate information always. All years together gathered information can be saved and can be accessed at any time. The data which is stored in the repository helps in taking intelligent decisions by the management. So it is better to have a mobile application.

2. LITERATURE REVIEW

2.1 EXISTING PROBLEM

2.1.1 TITLE: Impact of COVID-19 containment zone built-environments on Mental health and their coping mechanisms

AUTHOR: Farhan Asim a, , P.S. Chani

The 2nd wave of COVID-19 in 2021 had put several higher educational institutions in India into complete lockdowns and some were converted into containment zones to prevent the further spread of infection. A study was conducted on a student population (N = 432) in one such institute campus across three different hostel complexes to understand the role of containment zone Built Environments in the prevalence of Anxiety and Depression. The results from the analysis revealed that students living in rooms which have access to qualitative Built Environment aspects such as quality window views overlooking greenery and sky in addition to presence of indoor plants and portrait/artworks, are at lower risk of depression and severity of anxiety. The linear regression results of Built Environment Variable (BEV) Score with CES-D and GAD-7 advocated for an inverted relationship. The multinomial logistic regression revealed that with each 1-unit increase in BEV Score there is a significant decrease of 1.3-unit of likelihood of Depression, 0.779, 0.712 and 0.614-unit decrease in the likelihood of mild, moderate and severe levels of anxiety respectively. For Adverse Effect on Productivity, a 1-unit increase in BEV Score increases the likelihood of No-effect on productivity by 1.277 units. In order to precisely predict the effect of individual aspects of University Campus Built environments on the mental health of students during such containment and quarantine situations, further multidisciplinary investigation is required on the spheres of Built Environment, Psychology, and Epidemiology.

2.1.2 TITLE:

A Systematic Review of Smartphone Applications Available for Corona Virus Disease 2019 (COVID19) and the Assessment of their Quality Using The Mobile Application Rating Scale (MARS)

AUTHOR: Samira Davalbhakta1 & Shailesh Advani

The global impact of COVID-19 pandemic has led to a rapid development and utilization of mobile health applications. These are addressing the unmet needs of healthcare and public health system including contact tracing, health information dissemination, symptom checking and providing tools for training healthcare providers. Here we provide an overview of mobile applications being currently utilized for COVID-19 and their assessment using the Mobile Application Rating Scale. We performed a systematic review of the literature and mobile platforms to assess mobile applications currently utilized for COVID-19, and a quality assessment of these applications using the Mobile Application Rating Scale (MARS) for overall quality, Engagement, Functionality, Aesthetics, and Information. Finally, we provide an overview of the key salient features that should be included in mobile applications being developed for future use. Our search identified 63 apps that are currently being used for COVID-19. Of these, 25 were selected from the Google play store and Apple App store in India, and 19 each from the UK and US. 18 apps were developed for sharing up to date information on COVID-19, and 8 were used for contact tracing while 9 apps showed features of both. On MARS Scale, overall scores ranged from 2.4 to 4.8 with apps scoring high in areas of functionality and lower in Engagement. Future steps should involve developing and testing of mobile applications using assessment tools like the MARS scale and the study of their impact on health behaviours and outcomes.

A cost-efective alertness-rating tool to enable situational awareness

2.1.3 TITLE:

among on-duty static security guards in Covid-19 pandemic

AUTHOR: Debabrata Bej1, · Shubhranil Chakraborty

The workload of the static security guards has doubled due to the Covid-19 outbreak. In addition to their regular duties, they undertake some additional tasks to evaluate each individual's body temperature and welcome them with a hand sanitizer. In this scenario, their situational awareness is hugely desirable to perform these activities for the entire campus's safety. This situational awareness of guards means their ability to observe, inspect, and make the right decisions. However, due to their fatigue and other secondary activities, such as cell phone use, they cannot perform their duties correctly. In this context, this paper presents a method for sending random alarms in real-time to the on-duty guards, who are executing their work at the campus gates, remotely monitoring the alertness throughout the day from the head security ofce. For alertness detection, the system uses a simple client–server model. The system is designed using NodeMCU Wi-Fi modules. The algorithm of the Client, server, and repeater has been developed. The prototype has been tested by placing it on the working individuals' desk inside the departmental lab inside the campus. The system records the response time of the working individuals. These data are further used to calculate their percentage of alertness. In addition, an alertness-rating/scoring method has been developed to improve their work performance. This system can be an economical solution to enable the awareness of on-duty guards.

A robust tracking system for COVID-19 like pandemic using advanced hybrid technologies

AUTHOR: Shibli Nisar1 · Muhammad Ali Zuhaib

2.1.4 TITLE:

In order to track patients in coronavirus (COVID-19) like pandemic, this paper proposes a novel model based on hybrid advance technologies, which is capable to trace and track COVID-19 affectees with high accuracy. The hybrid technologies include, cellular, cyber and low range wireless technologies. This technique is capable to trace patients through call data record using cellular technology, voice over Internet protocol calls using cyber technology and physical contact without having a call history using low range wireless technologies. The proposed model is also capable to trace COVID-19 suspects. In addition to tracking, the proposed model is capable to provide surveillance capability as well by geo tagging the patients. In case of any violation by the patients an alert is sent to the concerned department. The proposed model is cost effective and privacy preserved as the entire process is carried out under the umbrella of a concerned government department. The potential outcomes of the proposed model are tracking of COVID-19 patients, monitoring of isolated patients, tracking of suspected ones and inform the mass about the safest path to use. Nowadays, trend of VOIP calls are exponentially increasing due to availability, reliability and speed of the Internet. Since calls made on VOIP are not registered in CDR log, therefore, it is not possible to track such cases using call data record analysis (CDRA). In addition to this, if call is made on such apps and mobile is not taken along then the contact tracing strategy fails to track the COVID19 patients. In order to overcome these limitations, the proposed model includes additional tracking feature to track COVID-19 patients based on the VOIP calls data log. The proposed model is capable of monitoring and tracing patients with and without call history of COVID19 patients along with the calls made.

2.1.5

TITLE: Research and Development of a COVID-19 Tracking System in Order to Implement Analytical Tools to Reduce the Infection Risk

AUTHOR: Erik Vavrinsky , Tomas Zavodnik , Tomas Debnar

The whole world is currently focused on COVID-19, which causes considerable economic and social damage. The disease is spreading rapidly through the population, and the effort to stop the spread is entirely still failing. In our article, we want to contribute to the improvement of the situation. We propose a tracking system that would identify affected people with greater accuracy than medical staff can. The main goal was to design hardware and construct a device that would track anonymous risky contacts in areas with a highly concentrated population, such as schools, hospitals, large social events, and companies. We have chosen a 2.4 GHz proprietary protocol for contact monitoring and mutual communication of individual devices. The 2.4 GHz proprietary protocol has many advantages such as a low price and higher resistance to interference and thus offers benefits. We conducted a pilot experiment to catch bugs in the system. The device is in the form of a bracelet and captures signals from other bracelets worn at a particular location. In case of contact with an infected person, the alarm is activated. This article describes the concept of the tracking system, the design of the devices, initial tests, and plans for future use. The alarm (threat level) is displayed optically via RGB light emitting diodes (LEDs) in a bracelet. The user can also check their status using the web interface, and if the wearer pairs the bracelet with their phone, he or she may also receive an SMS notification. However, it must be clearly stated to the wearer that the bracelet does not check their current state of infection in any way. It only tracks contacts and helps to find at-risk contacts. It may be advantageous if these portable tracker devices also include sensors for monitoring human physiology. In particular, measuring body temperature can provide information about the disease and give an early indication of elevated body temperature. The person will be informed and can seek professional help for a test. In the initial stages, of course, we do not calculate the mass deployment of the tracking system.

2.2 REFERENCES

1. Farhan Asim a, P.S. Chani, Impact of COVID-19 containment zone builtenvironments on students' mental health and their coping mechanisms, 2021.

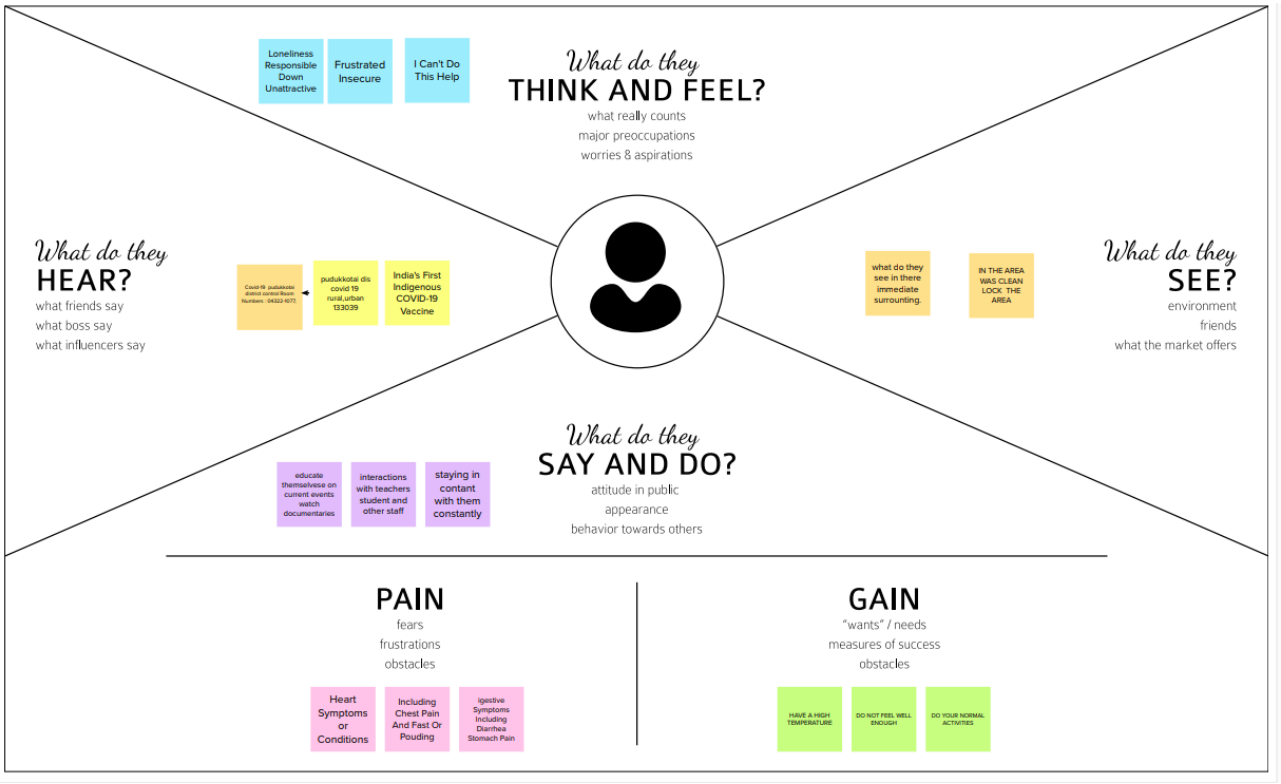
2. Samira Davalbhakta1 & Shailesh Advani, A Systematic Review of Smartphone Applications Available for Corona Virus Disease 2019 (COVID19) and the Assessment of their Quality Using the Mobile Application Rating Scale (MARS), 2020.
3. Debabrata Bej1, · Shubhranil Chakraborty,A cost-efective alertness-rating tool to enable situational awareness among on-duty static security guards in Covid-19 pandemic, 2020.
4. Shibli Nisar1 · Muhammad Ali Zuhaib, A robust tracking system for COVID-19 like pandemic using advanced hybrid technologies, 2021.
5. Erik Vavrinsky , Tomas Zavodnik , Tomas Debnar,

2.3 PROBLEM STATEMENT DEFINITION

In this existing system is fully manual system. News and media play a major role in creating this awareness and thus informing the public about preventive measures that can prevent infection. Creating awareness among the people to take all preventive measures would go a long way in reducing the spread of the virus.

3. IDEATION & PROPOSED SOLUTION

3.1 EMPATHY MAP CANVAS



3.2 IDEATION & BRAINSTORMING

3.3 PROPOSED SOLUTION

The web application shows the location of the containment zones to the users. It also notifies the user when he or she trespasses the boundary of a containment zone or stays in the containment zones. The application further tracks the user's location and provides notification alert if the user has entered a containment zone. The application also provides daily Covid-19 case statistics to the users to keep them updated. The application is developed on python to store the location data. This application around the containment zones and notification manager is used to provide notifications.

3.4 PROBLEM SOLUTION FIT

Existing system is a manual process so it required lots of man power and paper work. And also its have more possibilities to lost the data. It also have the possibilities to misuse the doctors details. The Proposed system requires less time for completion of any work. This is contrast to the sequential search being done in the existing system have been taken care of in the system. The proposed system have emergency alert feature when the user can view the doctor details without registrat

4. REQUIREMENT ANALYSIS

4.1 FUNCTIONAL REQUIREMENT

- Admin module
  In this module, the admin can login the system using his/ her user name and password.
- Update the danger zone details
  In this module, the admin can update the danger zone details to this application.
- User module
  There is registration form available where new user can create their account by providing required information to the system. The registration form details are like name, email, gender, mobile number, address, and etc. These details are stored in the database. And then can getting to the username and password in the system. After the registration process the user can login in the system using username and password.
- Tracking location

  The application uses the network to load the maps, to retrieve containment zone data from the database and to get Covid-19 statistics from this application. . After the login process the user can track the location. He/she can enter location details and then the corresponding covid affected area zone location ns are displayed.

- Alert system
  In this module, send the alert to the user. It can be helpful to predict the locations of user.

4.2 NON FUNCTIONAL REQUIREMENTS

Usability

The system shall allow the users to access the system with pc using web application. The system uses a web application as an interface. The system is user friendly which makes the system easy

Availability

The system is available 100% for the user and is used 24 hrs a day and 365 days a year. The system shall be operational 24 hours a day and 7 days a week.

Scalability

Scalability is the measure of a system's ability to increase or decrease in performance and cost in response to changes in application and system processing demands.

Security

A security requirement is a statement of needed security functionality that ensures one of many different security properties of software is being satisfied.

Performance

The information is refreshed depending upon whether some updates have occurred or not in the application. The system shall respond to the member in not less than two seconds from the time of the request submittal. The system shall be allowed to take more time when doing large processing jobs. Responses to view information shall take no longer than 5 seconds to appear on the screen.

Reliability

The system has to be 100% reliable due to the importance of data and the damages that can be caused by incorrect or incomplete data. The system will run 7 days a week. 24 hours a day.

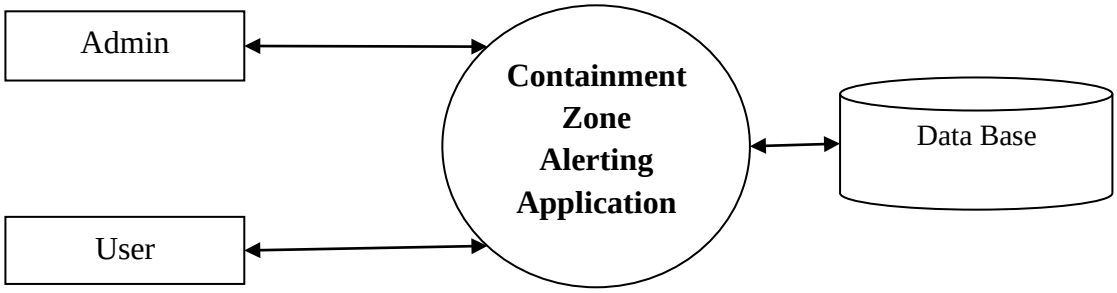
5. PROJECT DESIGN

5.1 DATA FLOW DIAGRAMS

A data-flow diagram is a visual representation of how data moves through a system or a process (usually an information system). The DFD additionally gives details about each entity's inputs and outputs as well as the process itself. A data-flow diagram lacks control flow, loops, and decision-making processes. Using a flowchart, certain operations depending on the data may be depicted.

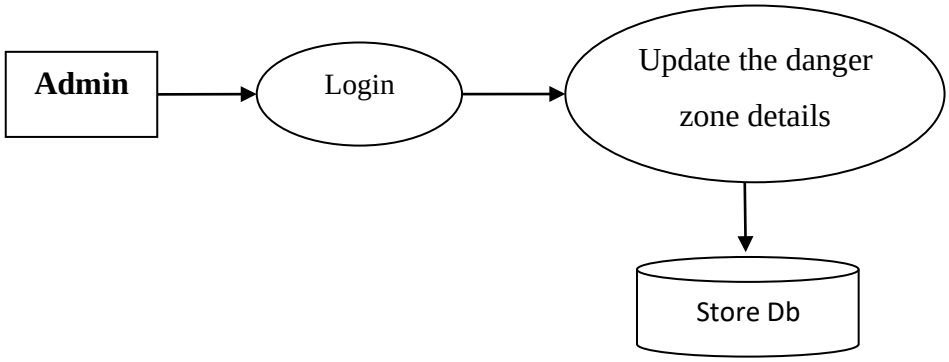
LEVEL 0

The Level 0 DFD shows how the system is divided into 'sub-systems' (processes), each of which deals with one or more of the data flows to or from an external agent, and which together provide all of the functionality of the system as a whole. It also identifies internal data stores that must be present in order for the system to do its job, and shows the flow of data between the various parts of the system.



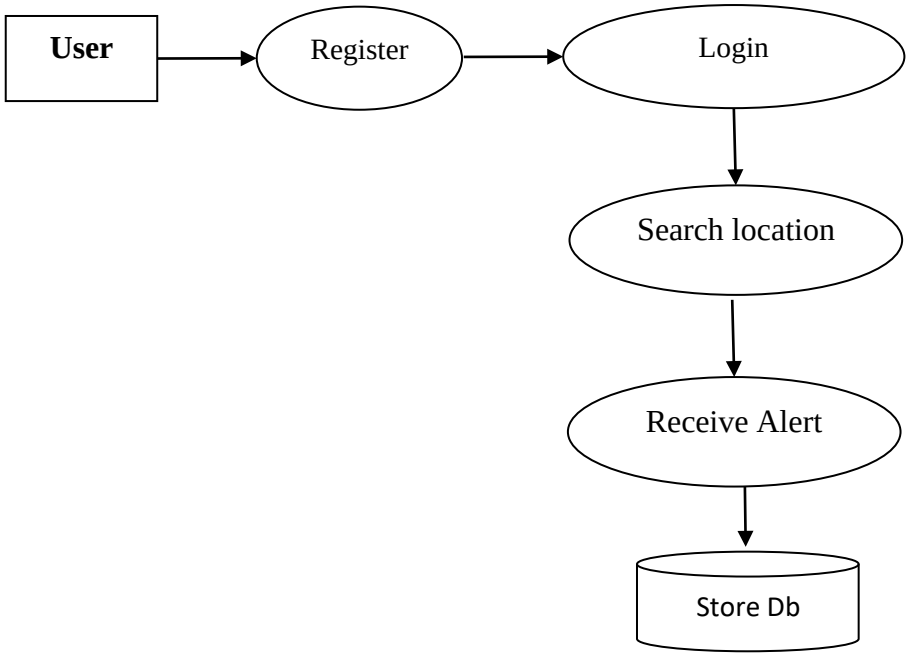
LEVEL 1

The next stage is to create the Level 1 Data Flow Diagram. This highlights the main functions carried out by the system. As a rule, to describe the system was using between two and seven functions - two being a simple system and seven being a complicated system. This enables us to keep the model manageable on screen or paper.

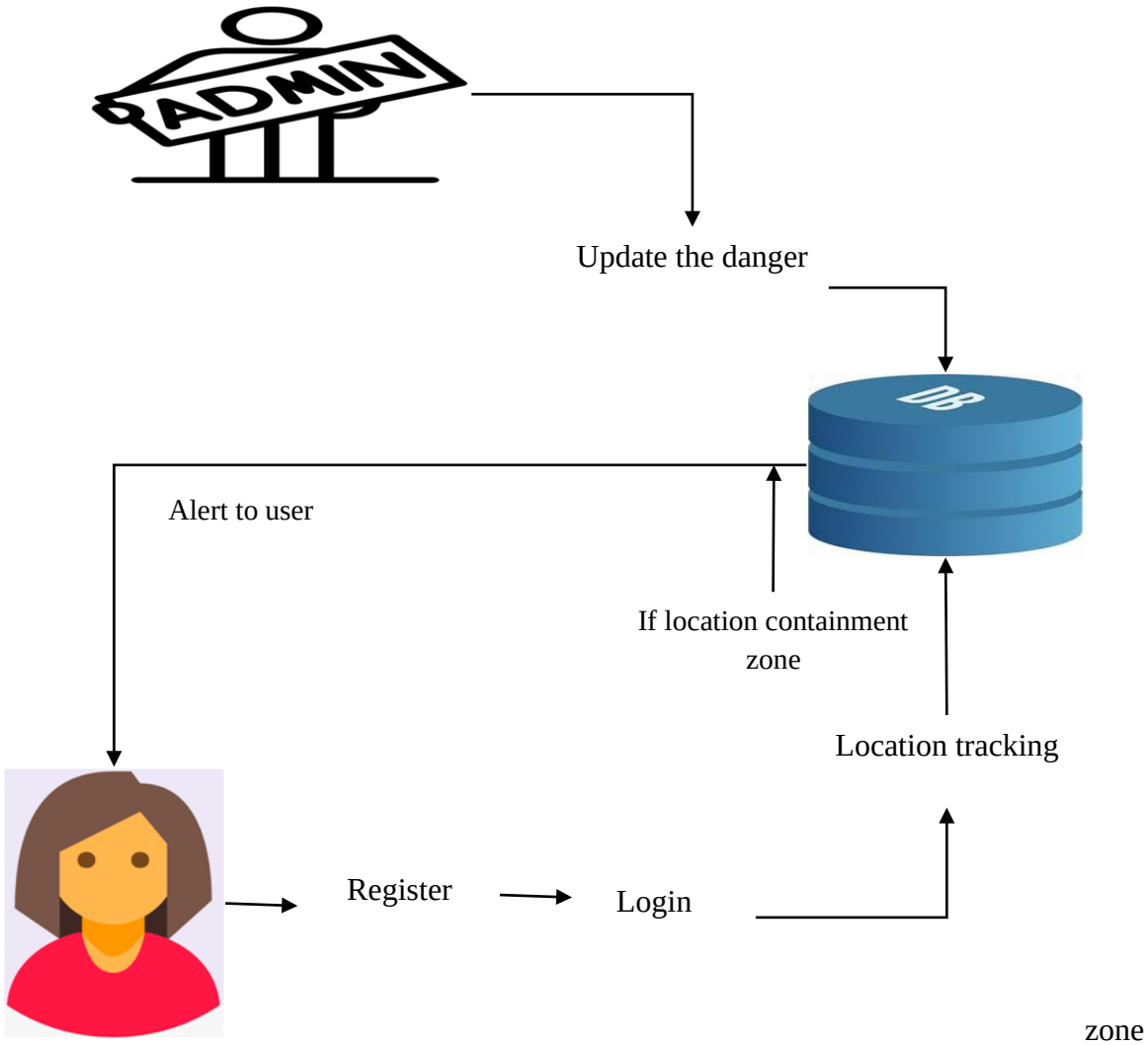


LEVEL-2

A Data Flow Diagram (DFD) tracks processes and their data paths within the business or system boundary under investigation. A DFD defines each domain boundary and illustrates the logical movement and transformation of data within the defined boundary. The diagram shows 'what' input data enters the domain, 'what' logical processes the domain applies to that data, and 'what' output data leaves the domain. Essentially, a DFD is a tool for process modelling and one of the oldest.



5.2 SOLUTION & TECHNICAL ARCHITECTURE



5.3 USER STORIES

6.1

6. PROJECT PLANNING & SCHEDULING

SPRINT PLANNING & ESTIMATION

6.2 SPRINT DELIVERY SCHEDULE

6.3 REPORTS FROM JIRA

7. CODING & SOLUTIONING

7.1
FEATURE 1

7.2 FEATURE 2

7.3 DATABASE SCHEMA

8. TESTING

8.1

TEST CASES

A test case has components that describe input, action and an expected response, in order to determine if a feature of an application is working correctly. A test case is a set of instructions on “HOW” to validate a particular test objective/target, which when followed will tell us if the expected behavior of the system is satisfied or not.

Characteristics of a good test case:

- Accurate: Exacts the purpose.
- Economical: No unnecessary steps or words.
- Traceable: Capable of being traced to requirements.
- Repeatable: Can be used to perform the test over and over.
- Reusable: Can be reused if necessary.

S.NO	**Scenario**	**Input**	**Excepted output**	**Actual output**
1	Admin Login Form	User name and password	Login	Login success.
2	Upload containment zone details	containment zone details	Added successfully	containment zone details stored in database.
3	User Login Form	User name and password	Login	Login success.

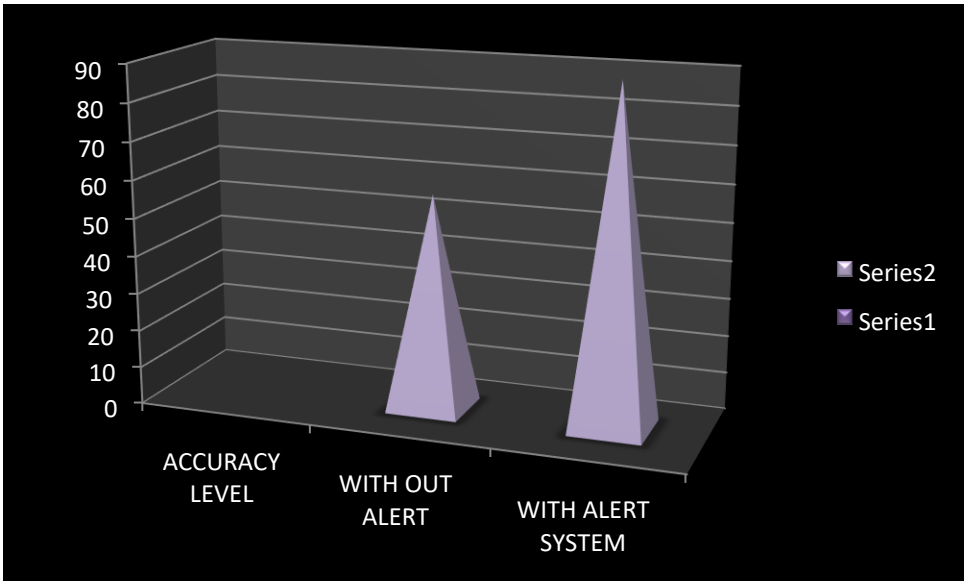
9.1

8.2 USER ACCEPTANCE TESTING

This is a type of testing done by users, customers, or other authorised entities to determine application/software needs and business processes. Acceptance testing is the most important phase of testing as this decides whether the client approves the application/software or not. It may involve functionality, usability, performance, and U.I of the application. It is also known as user acceptance testing (UAT), operational acceptance testing (OAT), and end-user testing.

9. RESULTS

9.1 PERFORMANCE METRICS



10. ADVANTAGES & DISADVANTAGES

ADVANTAGES

- Reduce the time.
- User can search the information very fast.
- Workload and manpower is reduced.
- Report generating is very fast.

DISADVANTAGES

- The existing system is manual.
- The manual system is more error prone.
- Immediate response to the queries is difficult and time consuming.
- Difficult to maintain record and more paper work is required.
- It is not comfortable for user.

11. CONCLUSION

The programme offers customers an effective way to view the designated Covid-19 containment zones on a system. This developed application can be used as a tool to increase social awareness among the public in light of the worrisome rise of Covid-19 impacted patients around the world. The location of the user is also tracked by this application, which determines if it is among the designated containment zones. On entry, it notifies the user separately via alerts. The created Web application also extracts Number of the intruder in the containment zones, which might help the local police trace and identify persons who repeatedly enter the containment zones without permission. By doing so, this programme identifies the containment zones and emphasises the importance of taking further precautions to resist COVID-19. The application has undergone extensive testing and has shown to produce reliable results.

12. FUTURE SCOPE

The scope of the project is the system on which the software is installed, i.e. the project is developed as a desktop application, and it will work for a particular institute. But later on the project can be modified to operate it Containment Zone Application. In future we can develop this project in android application.

13. APPENDIX

SOURCE CODE

```
from flask import Flask, render_template, flash, request, session
from flask import render_template, redirect, url_for, request import
datetime

import ibm_db import pandas import
ibm_db_dbi from sqlalchemy import
create_engine

engine = create_engine('sqlite://',
echo = False)

dsn_hostname = "b0aebb68-94fa-46ec-a1fc-
1c999edb6187.c3n41cmd0nqnrk39u98g.databases.appdomain.cloud"
dsn_uid = "qdd33612" dsn_pwd = "mQcscHzl8Qeeqyix"

dsn_driver = "{IBM DB2 ODBC DRIVER}"
dsn_database = "BLUDB" dsn_port =
"31249" dsn_protocol = "TCPIP"
dsn_security = "SSL"

dsn = (
    "DRIVER={0};"
    "DATABASE={1};"
    "HOSTNAME={2};"
    "PORT={3};"
    "PROTOCOL={4};"
    "UID={5};"
    "PWD={6};"
    "SECURITY={7};").format(dsn_driver, dsn_database, dsn_hostname, dsn_port,
dsn_protocol, dsn_uid, dsn_pwd,dsn_security)
```

```
try:
   conn = ibm_db.connect(dsn, "", "")     print ("Connected to database: ",
dsn_database, "as user: ", dsn_uid, "on host: ", dsn_hostname)

except:
   print ("Unable to connect: ", ibm_db.conn_errormsg() ) app =
Flask(__name__) app.config['DEBUG']
app.config['SECRET_KEY'] = '7d441f27d441f27567d441f2b6176a'


@app.route("/") def
homepage():
   return render_template('index.html')
@app.route("/Home") def
Home():
   return render_template('index.html')
@app.route("/AdminLogin") def
AdminLogin():
   return render_template('AdminLogin.html')


@app.route("/UserLogin") def
UserLogin():
   return render_template('UserLogin.html')
@app.route("/NewUser") def
NewUser():
   return render_template('NewUser.html')


@app.route("/NewMedicine") def
NewMedicine():
```

```
    return render_template('NewMedicines.html')

@app.route("/adminlogin", methods=['GET', 'POST'])
def adminlogin():     error = None     if request.method
== 'POST':
      if request.form['uname'] == 'admin' and request.form['password'] == 'admin':

          conn = ibm_db.connect(dsn, "", "")           pd_conn =
ibm_db_dbi.Connection(conn)           selectQuery = "SELECT *
FROM regtb  "           dataframe = pandas.read_sql(selectQuery,
pd_conn)           dataframe.to_sql('booktb1', con=engine,
if_exists='append')           data = engine.execute("SELECT * FROM
booktb1").fetchall()

          return render_template('AdminHome.html', data=data)

else:
      return render_template('index.html', error=error)




@app.route("/newmedi", methods=['GET', 'POST'])
def newmedi():     if request.method == 'POST':


      State = request.form['State']
      City = request.form['City']
      Area = request.form['Area']
Patient = request.form['Patient']
info = request.form['info']

      Containment = request.form['Containment']
```

```
        file = request.files['file']
file.save("static/upload/" + file.filename)



        conn = ibm_db.connect(dsn, "", "")

        insertQuery =  "INSERT INTO meditb VALUES ('"+ State +"','" + City + "','" + Area +
"','" + Patient + "','" + info + "','" + Containment + "','" + file.filename + "')"
insert_table = ibm_db.exec_immediate(conn, insertQuery)

        if(Containment=="Yes"):

            conn = ibm_db.connect(dsn, "", "")            pd_conn
= ibm_db_dbi.Connection(conn)                selectQuery =
"SELECT    *    FROM    regtb"                  dataframe  =
pandas.read_sql(selectQuery, pd_conn)

            dataframe.to_sql('regtb', con=engine, if_exists='append')
data2 = engine.execute("SELECT * FROM regtb").fetchall()

            for item in data2:
email = item[4]

                sendmail(email,"New Zone Added Please Vist Website")



        alert = 'Zone info saved successfully' return
        render_template('goback.html', data=alert)

@app.route("/MedicineInfo") def
MedicineInfo():
```

```
    conn = ibm_db.connect(dsn, "", "")     pd_conn =
ibm_db_dbi.Connection(conn)     selectQuery = "SELECT *
FROM meditb  "     dataframe = pandas.read_sql(selectQuery,
pd_conn)     dataframe.to_sql('booktb1', con=engine,
if_exists='append')     data = engine.execute("SELECT * FROM
booktb1").fetchall()

    return render_template('MedicinesInfo.html', data=data)



@app.route("/newuser", methods=['GET', 'POST'])
def newuser():     if request.method == 'POST':

        name1 = request.form['name']
gender1 = request.form['gender']
Age = request.form['age']         email
= request.form['email']         pnumber
= request.form['phone']         address
= request.form['address']

        uname = request.form['uname']
password = request.form['psw']

        conn = ibm_db.connect(dsn, "", "")

        insertQuery = "INSERT INTO regtb VALUES ('" + name1 + "','" + gender1 + "','" + Age
+ "','" + email + "','" + pnumber + "','" + address + "','" + uname + "','" + password + "')"
insert_table = ibm_db.exec_immediate(conn, insertQuery)
        # return 'file register successfully'



    return render_template('UserLogin.html')
```

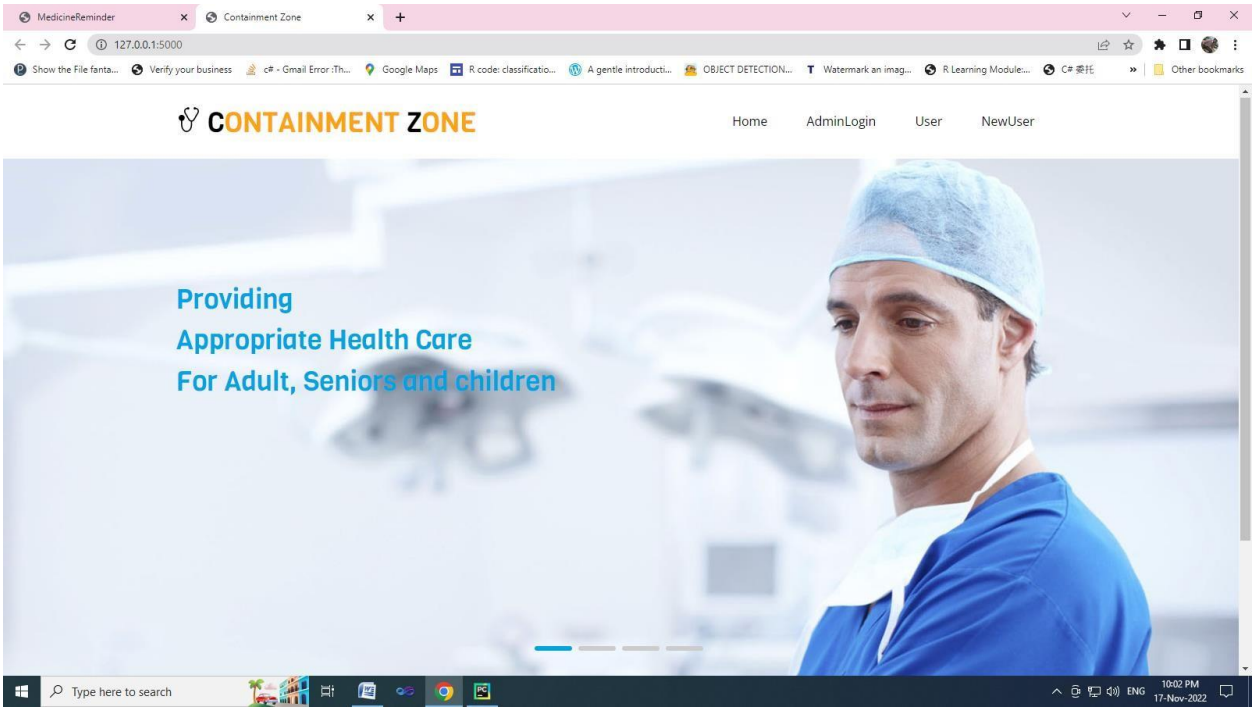
```
@app.route("/userlogin", methods=['GET', 'POST'])
def userlogin():     error = None     if request.method
== 'POST':
      username = request.form['uname']
password = request.form['password']
session['uname'] = request.form['uname']



      conn = ibm_db.connect(dsn, "", "")
pd_conn = ibm_db_dbi.Connection(conn)

      selectQuery = "SELECT * from regtb where UserName='" + username + "' and
password='" + password + "'"         dataframe = pandas.read_sql(selectQuery,
pd_conn)

      if dataframe.empty:
         data1 = 'Username or Password is wrong'
return render_template('goback.html', data=data1)
else:
         print("Login")            selectQuery = "SELECT * from regtb where
UserName='" + username + "' and password='" + password + "'"            dataframe =
pandas.read_sql(selectQuery, pd_conn)

         dataframe.to_sql('Employee_Data',
con=engine,                        if_exists='append')

         # run a sql query            print(engine.execute("SELECT * FROM
Employee_Data").fetchall())

         return render_template('UserHome.html', data=engine.execute("SELECT * FROM
Employee_Data").fetchall())
```

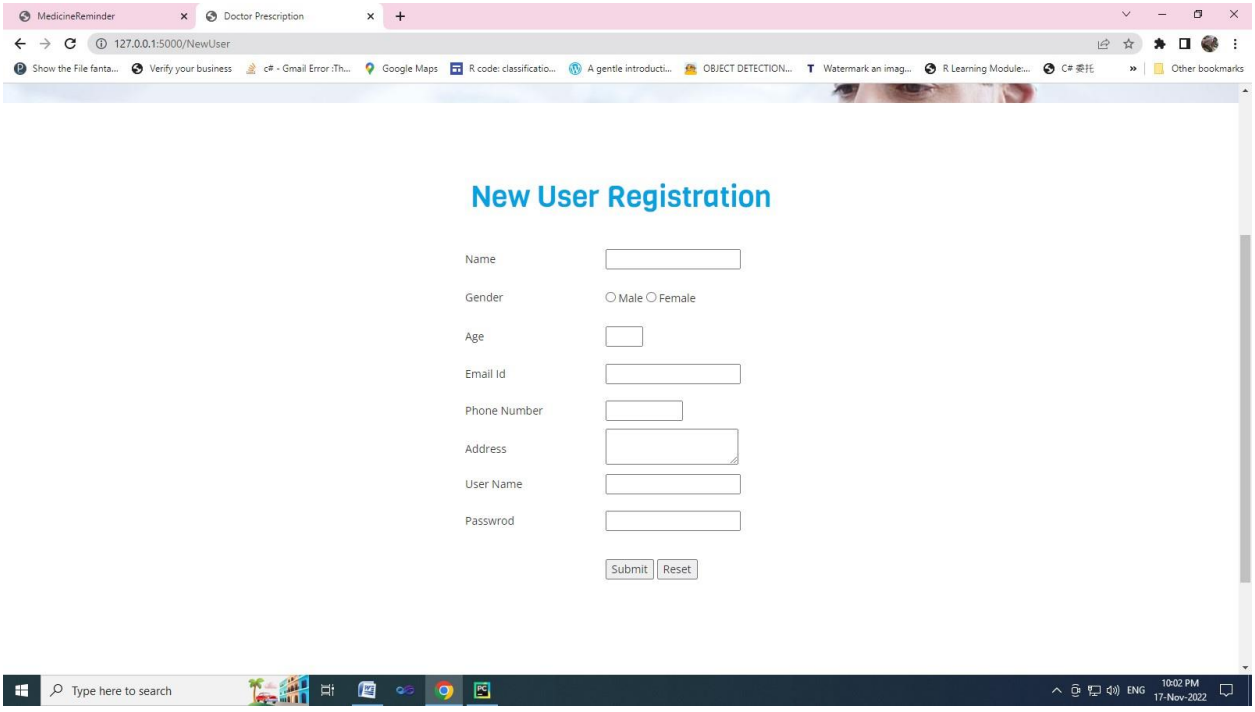
```
@app.route("/UserHome") def
UserHome():

    username = session['uname']


    conn = ibm_db.connect(dsn, "", "")     pd_conn =
ibm_db_dbi.Connection(conn)     selectQuery = "SELECT * FROM regtb
where username='" + username + "'"     dataframe =
pandas.read_sql(selectQuery, pd_conn)     dataframe.to_sql('booktb1',
con=engine, if_exists='append')     data = engine.execute("SELECT * FROM
booktb1").fetchall()

    return render_template('UserHome.html', data=data)


@app.route("/Alert") def
Alert():


    conn = ibm_db.connect(dsn, "", "")     pd_conn =
ibm_db_dbi.Connection(conn)     selectQuery = "SELECT *
FROM meditb "     dataframe = pandas.read_sql(selectQuery,
pd_conn)     dataframe.to_sql('booktb1', con=engine,
if_exists='append')     data = engine.execute("SELECT *
FROM booktb1  ").fetchall()

    return render_template('Alert.html', data=data)

@app.route("/Remove", methods=['GET']) def
Remove():
```

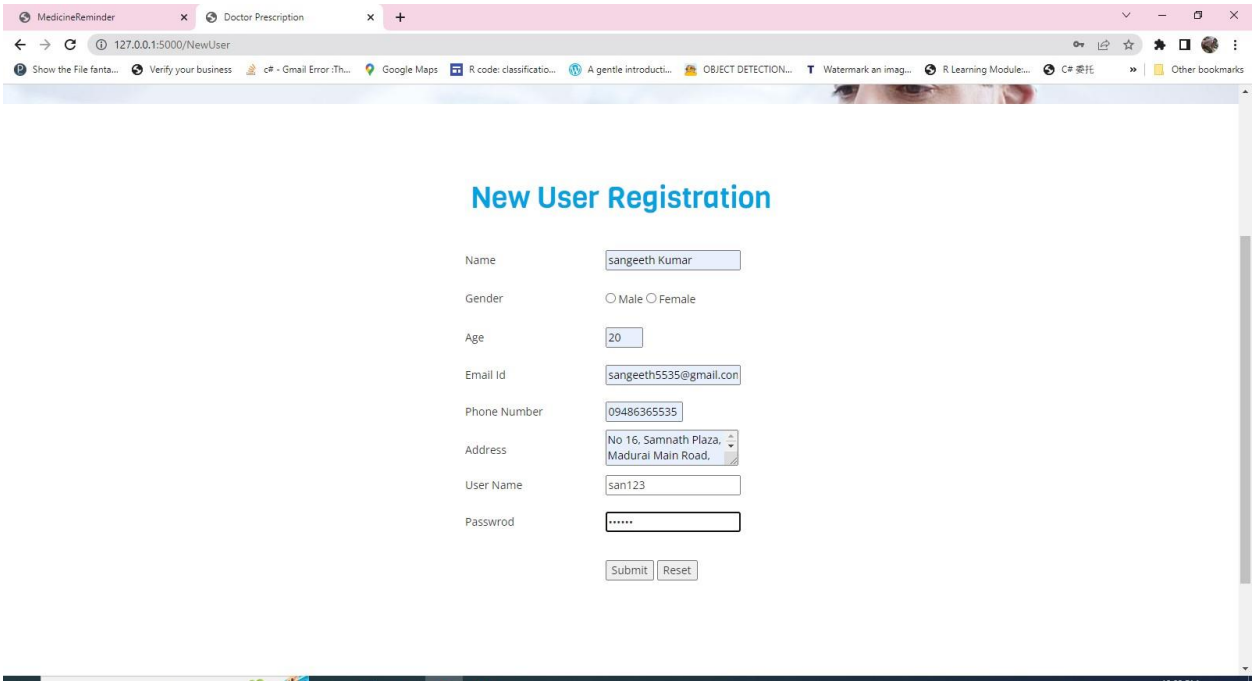
```
    unmae = request.args.get('id')      mname =
request.args.get('mname')      conn =
ibm_db.connect(dsn, "", "")      pd_conn =
ibm_db_dbi.Connection(conn)

    insertQuery = "Delete from meditb  where  username='"+ unmae +"' and
MediName='"+mname +"'"      insert_table =
ibm_db.exec_immediate(conn, insertQuery)



    selectQuery = "SELECT * from meditb "
dataframe = pandas.read_sql(selectQuery, pd_conn)

    dataframe.to_sql('Employee_Data',
                     con=engine,
if_exists='append')

    # run a sql query
    print(engine.execute("SELECT * FROM Employee_Data").fetchall())

    return render_template('MedicinesInfo.html', data=engine.execute("SELECT * FROM
Employee_Data").fetchall())



def sendmsg(targetno,message):
    import requests
requests.post("http://smsserver9.creativepoint.in/api.php?username=fantasy&password=5966
92&to=" + targetno + "&from=FSSMSS&message=Dear user  your msg is " + message + "
Sent By FSMSG
FSSMSS&PEID=1501563800000030506&templateid=1507162882948811640")



def sendmail(Mailid,message):
```

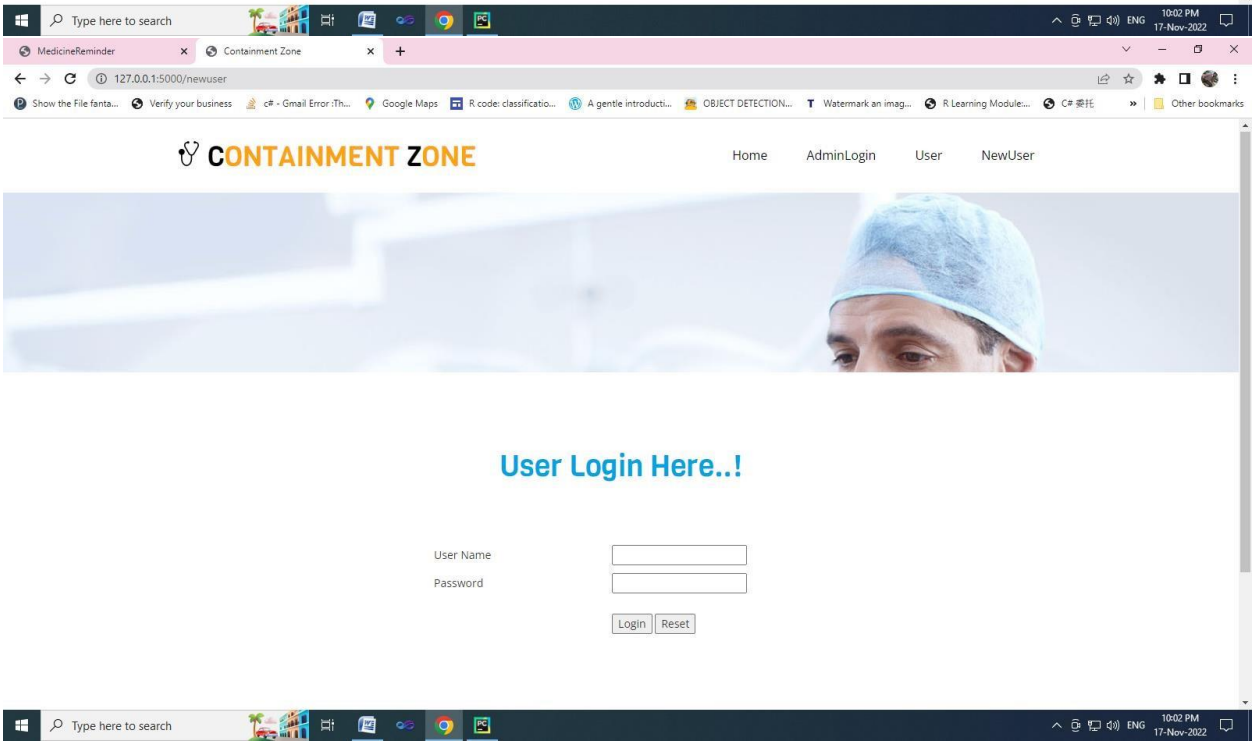
```
    import smtplib     from email.mime.multipart
import MIMEMultipart     from email.mime.text
import MIMEText     from email.mime.base import
MIMEBase     from email import encoders

    fromaddr = "sampletest685@gmail.com"
toaddr = Mailid

    # instance of MIMEMultipart
msg = MIMEMultipart()

    # storing the senders email address
msg['From'] = fromaddr

    # storing the receivers email address
msg['To'] = toaddr

    # storing the subject
msg['Subject'] = "Alert"

    # string to store the body of the mail
body = message

    # attach the body with the msg instance
    msg.attach(MIMEText(body, 'plain'))

    # creates SMTP session     s =
smtplib.SMTP('smtp.gmail.com', 587)

    # start TLS for security
s.starttls()

    # Authentication
    s.login(fromaddr, "hneucvnontsuwgpj")
```

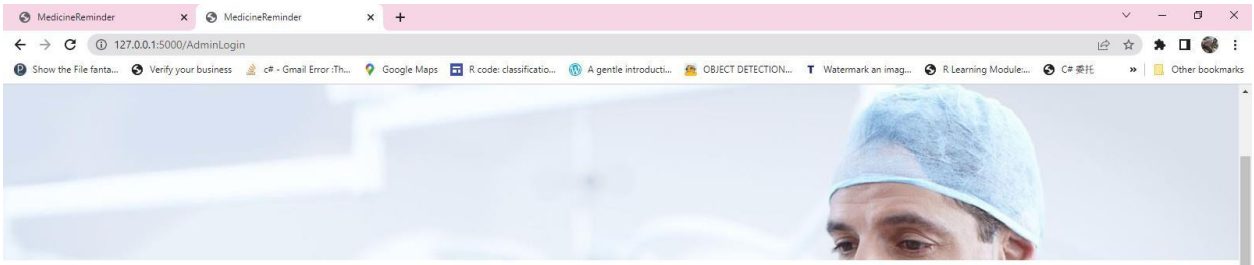
```
    # Converts the Multipart msg into a string
text = msg.as_string()

    # sending the mail
    s.sendmail(fromaddr, toaddr, text)

    # terminating the session


if __name__ == '__main__':
    app.run(debug=True, use_reloader=True)
```
SCREENSHOTS


MedicineReminder
Containment Zone
127.0.0.1:5000
CONTAINMENT ZONE
Home
AdminLogin
User
NewUser
Providing
Appropriate Health Care
For Adult, Seniors and children
Type here to search
ENG
10:02 PM
17-Nov-2022


MedicineReminder
Doctor Prescription
127.0.0.1:5000/NewUser
New User Registration
Name
Gender
Male
Female
Age
Email Id
Phone Number
Address
User Name
Passwrod
Submit
Reset
Type here to search
ENG
10:02 PM
17-Nov-2022


MedicineReminder
Doctor Prescription
127.0.0.1:5000/NewUser
New User Registration
Name
sangeeth Kumar
Gender
Male Female
Age
20
Email Id
sangeeth5535@gmail.con
Phone Number
09486365535
Address
No 16, Samnath Plaza,
Madurai Main Road,
User Name
san123
Passwrod
Submit
Reset


MedicineReminder
Containment Zone
127.0.0.1:5000/newuser
CONTAINMENT ZONE
Home
AdminLogin
User
NewUser
User Login Here..!
User Name
Password
Login
Reset

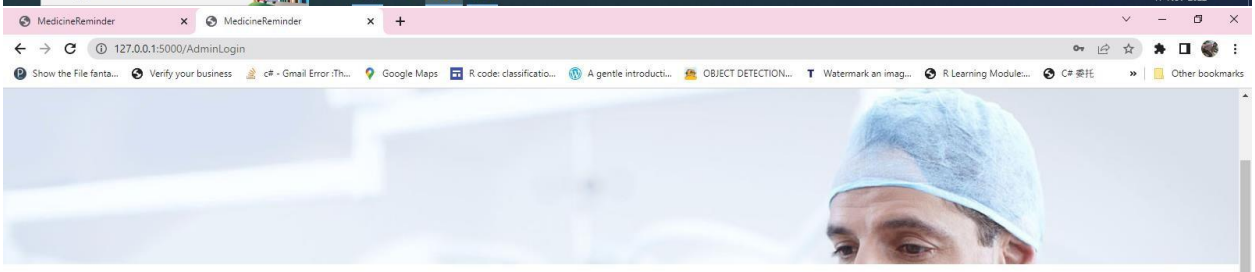


Admin Login Here..!

User Name

Password

Login Reset

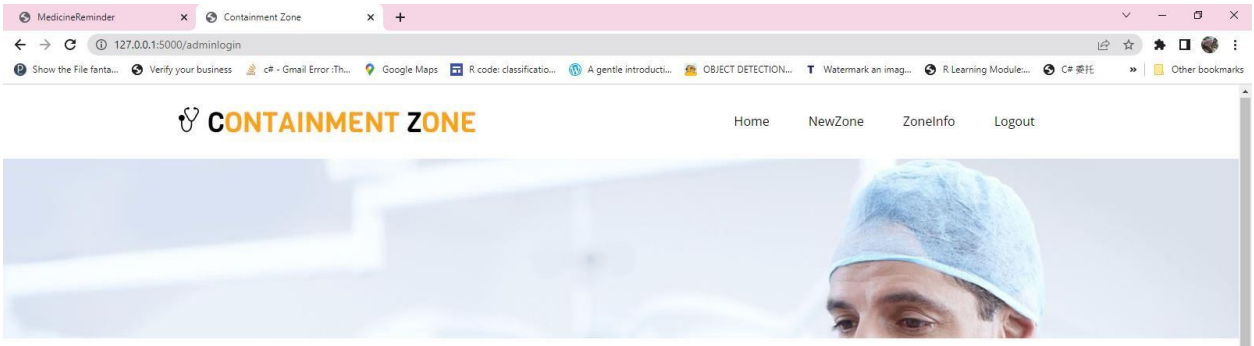


Admin Login Here..!

User Name admin

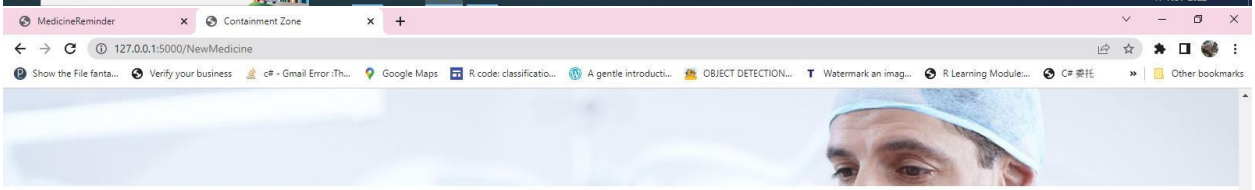
Password

Login Reset



User Information

Name	Gender	Age	EmailId	Phone	UserName
isha	female	20	isharavi172002@gmail.com	9486365535	isha
sangeeth Kumar	male	20	sangeeth5535@gmail.com	09486365535	san123



New Zone Information

State

City

Area Name

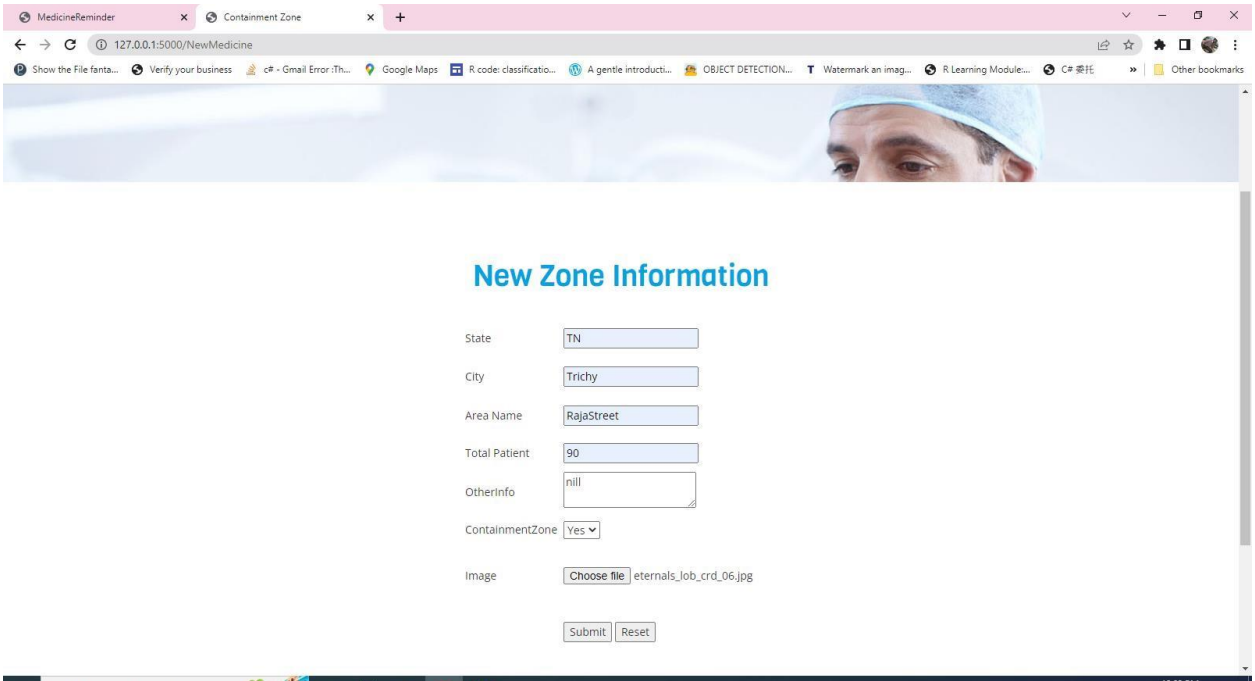
Total Patient

OtherInfo

ContainmentZone Yes

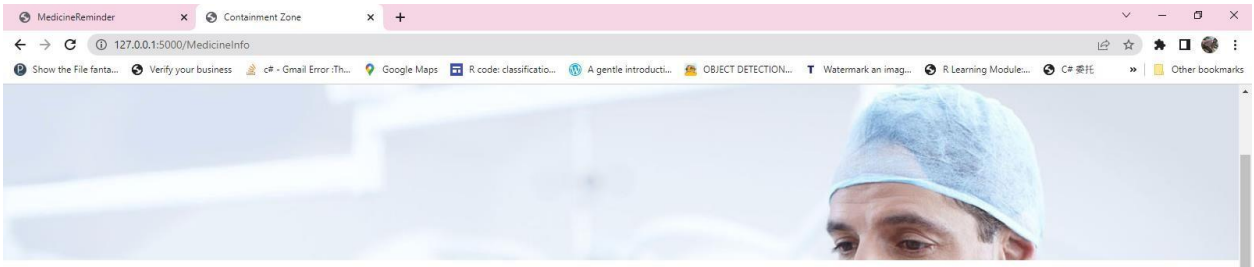
Image Choose file No file chosen

Submit Reset



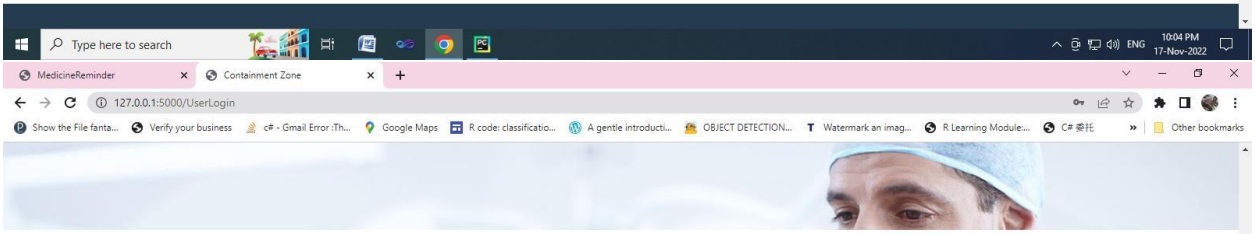
Zone info saved successfully

Go Back



Zone Information

State	City	Area	Patient	OtherInfo	Containment	Image	
TN	Trichy	RajaStreet	90	nill	Yes		Remove
TN	Trichy	RajaStreet	90	nill	Yes		Remove

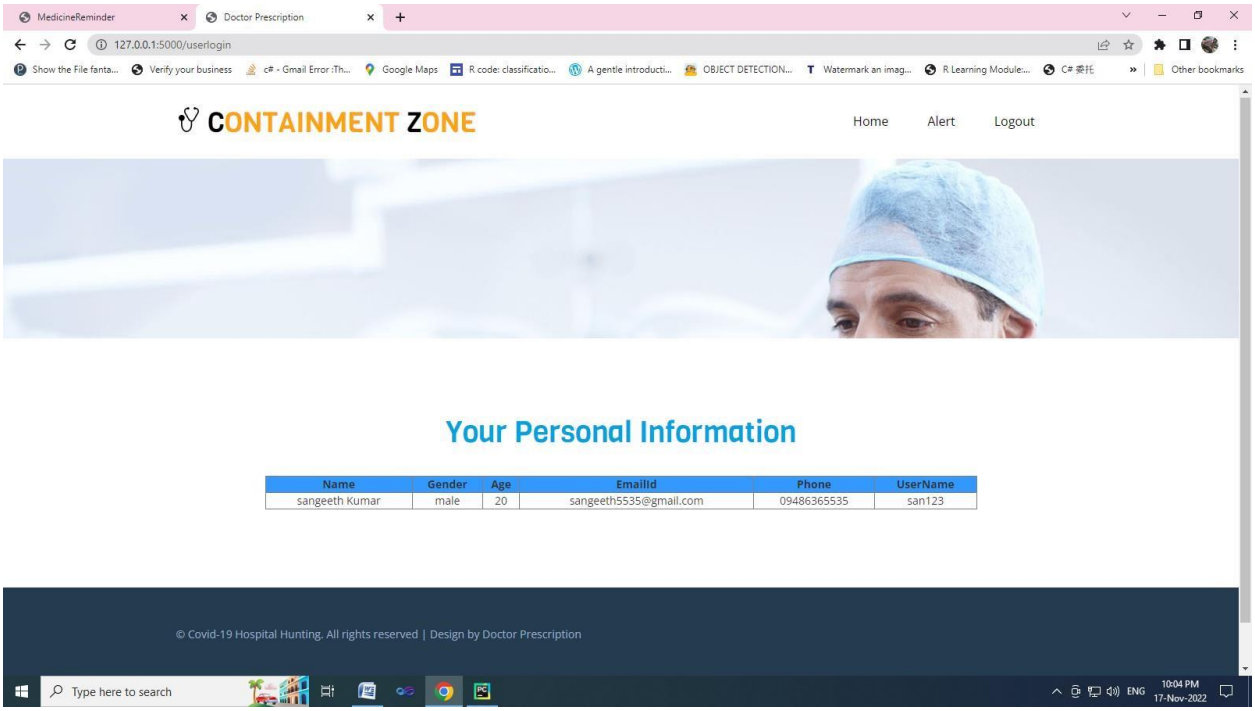


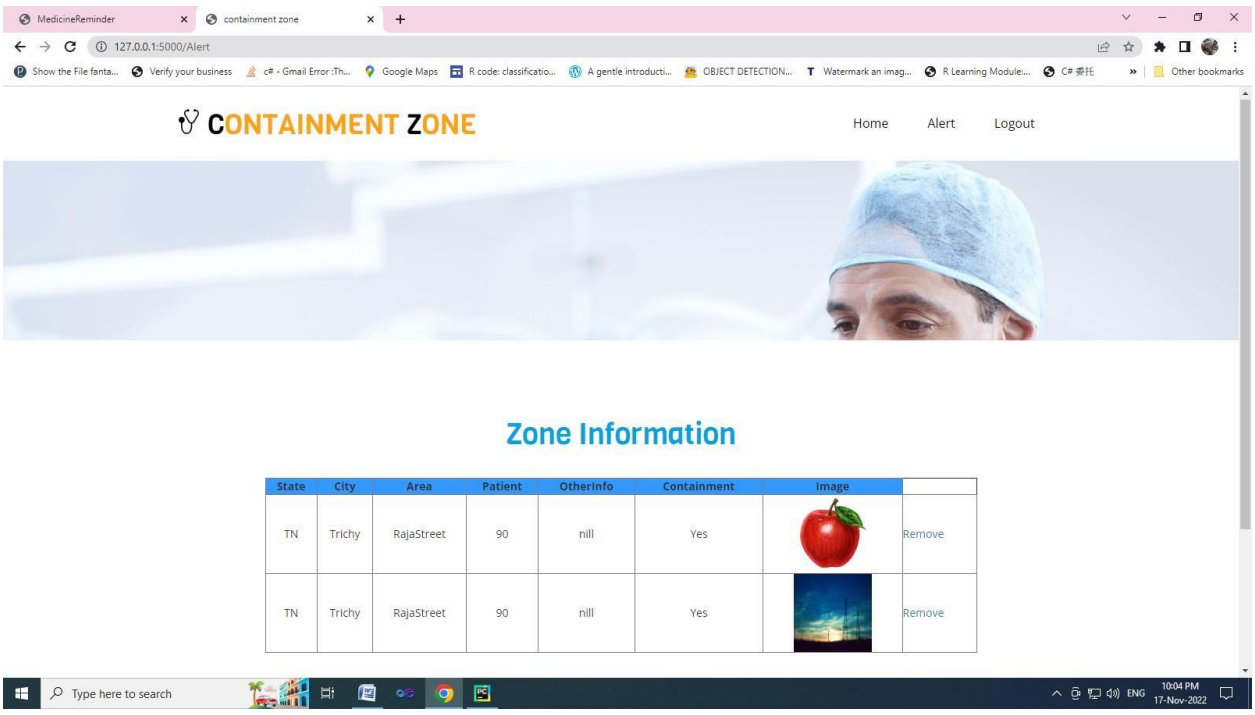
User Login Here..!

User Name: san123

Password: ••••••

Login Reset


MedicineReminder
Doctor Prescription
127.0.0.1:5000/userlogin
CONTAINMENT ZONE
Home
Alert
Logout
Your Personal Information
Name | Gender | Age | EmailId | Phone | UserName
sangeeth Kumar | male | 20 | sangeeth5535@gmail.com | 09486365535 | san123
© Covid-19 Hospital Hunting. All rights reserved | Design by Doctor Prescription
Type here to search
ENG
10:04 PM
17-Nov-2022


MedicineReminder
containment zone
127.0.0.1:5000/Alert
CONTAINMENT ZONE
Home
Alert
Logout
Zone Information
State | City | Area | Patient | OtherInfo | Containment | Image
TN | Trichy | RajaStreet | 90 | nill | Yes | Remove
TN | Trichy | RajaStreet | 90 | nill | Yes | Remove
Type here to search
ENG
10:04 PM
17-Nov-2022

GITHUB & PROJECT DEMO LINK

GITHUB LINK : https://github.com/IBM-EPBL/IBM-Project-15153-1659594452.git

PROJECT DEMO LINK : https://www.youtube.com/watch?v=qM7bQxNoTuE&t=0s