INTRODUCTION

Corona virus is a big family of viruses, from the December 2019 Covid-19 has been spread all over the world. Corona virus was first found in wuhan city of china. By the global case count from the Johns Hopkins University, 160,825,974 identified patients of COVID-19 have been reported so far, including 3,340,152 deaths and impacting more than 192 countries. Corona virus affects the humans and animals they causes respiratory infections that can be mild to severe. This virus spreads through the droplets of infected person when they exhale, coughs, sneeze. These droplets may exist in the atmosphere or may takes place in any surfaces or floor. Person can get covid positive by breath in the infected air if person is within close to someone who is covid positive, also infection can be spread by touching nose, eyes, mouth, contaminated surface.

For detection of lung infection doctors are using the X-Ray scanning technique but there X-Ray scanning techniques are not that much effective to find out the infected area. But the main problem with X-Rays is , in this it may be overlap with other diseases like pneumonia so, it become difficult and time consuming task for doctors to treat that patient. Instead of this X-Rays we can use CT images from which we can find out more detail and accurate result. It will reduce efforts as well as time also. So, it will be beneficial for doctors as well as patients.

The Automatic Covid-19 lung infection segmentation from CT images is system where user can give CT image as input and our system can find the infected areas of lungs from CT images and display those infected regions through different colors. Our system also calculate the percentage of infected area and display the Covid-19 risk according to the percentage obtained. This system is very useful to find the infected regions of lungs in short period of time. It is very useful in medical field and using this system doctors can easily findout infected areas of lung which is affected due to Covid-19 and easily identify the risk of Covid-19.

a. Problem definition:

Manual identification of COVID-19 lung infection is extremely time consuming and requires expertise, so we proposed ‘Automatic COVID-19 lung infection segmentation from CT images’ system which is used to identify infected areas in lung automatically.

b. Aim and objectives of the Project:

1) To perform edge detection on lung CT image.
2) To perform segmentation on lung CT image.
3) To find out infected area by covid-19 virus from input image.
4) To find out the percentage of infected area.

c. Scope and limitation of the Project:

Scope

1. This system has wide scope in the medical field.
2. This system will give the covid-19 lung infected percentage by considering the color intensity.

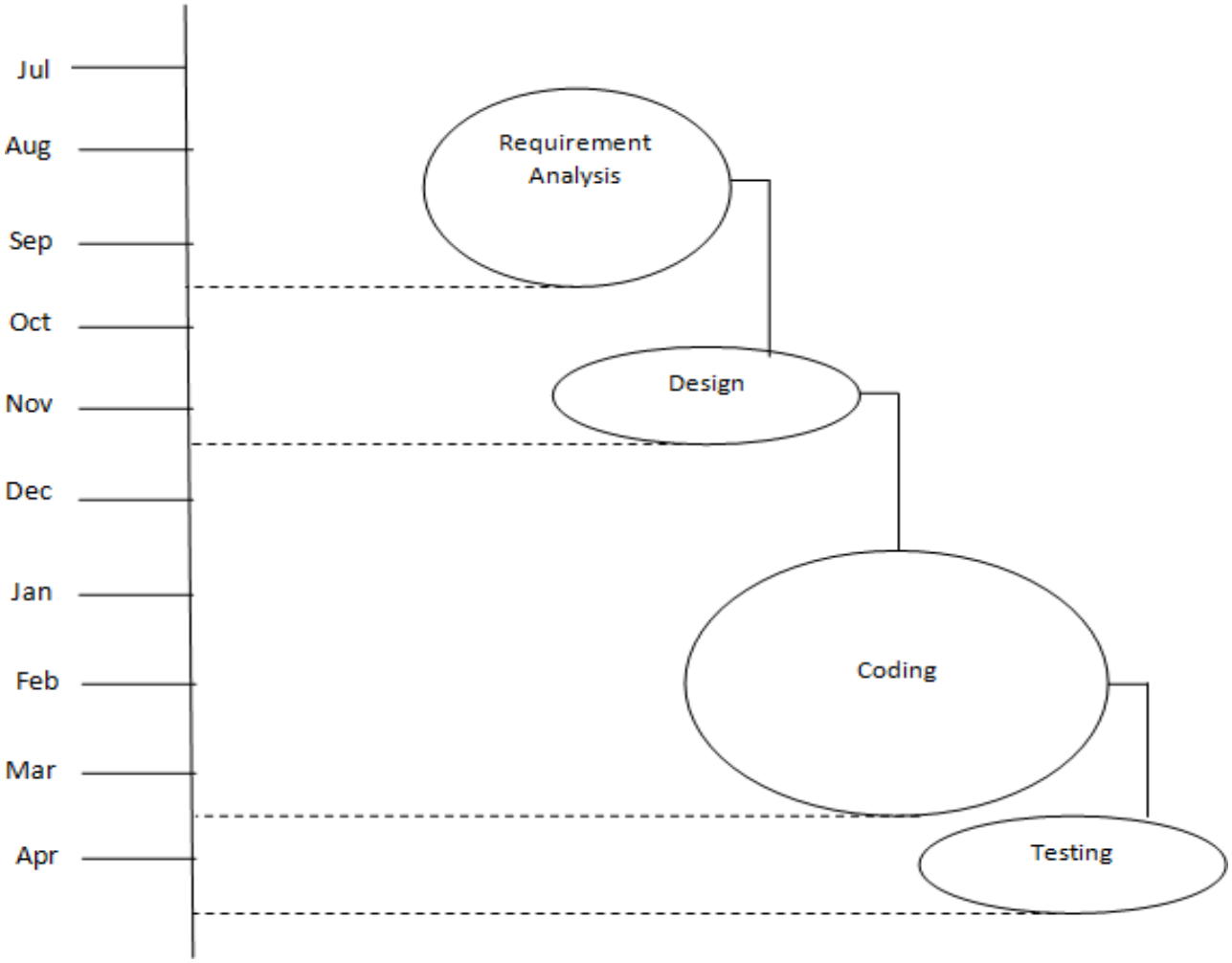
Limitations

1. Required the high resolution CT images for work because system will not work correctly in low resolution CT images.

d. TimeLine of Project

We have used classic life cycle paradigm also called “Water Fall Model”. For software engineering which is sequential approach to software development that begins at the system level and progress through analysis, design, coding, testing and maintenance. We had completed software requirement analysis by the mid of September 2020 which encompasses both system and software requirement gathering. By the end of December 2020 we had completed project planning and design. On the basis of design prepared in the previous stage by the end of Feb 2021 we completed coding stage.

After completion of coding stage the important part in the software development which is testing phase carried out in first week of March 2021.Various criteria of testing were taken into account which includes unit testing, integration testing, validation testing and system testing. First, each and every module of the project was tested under the unit testing. After the unit testing, integration testing was carried out by integrating all module tested in unit testing. After unit testing the module prepared was cross checked with the design.



Literature Overview

a. Literature Overview:

1. Build your covid-19 lung infection segmentation Model using OpenCV and Python:

This paper describes on building your own automatic detection of covid-19 lung infection segmentation from CT images. We'll first understand how to detect covid-19 lung infected before the implementation part. We'll be using OpenCV and Python to build the automatic detection of covid-19 lung infection segmentation from CT images.

2. Converting To Grey Scale Image And Edge Detection:

First, convert the image into greyscale image. After converting to greyscale, we use the canny method for detection of edge of the image. It is a multi-stage algorithm and we will go through each stages.First step is to remove the noise in the image with a 5x5 Gaussian filter. Finding Intensity Gradient of the Image. After getting gradient magnitude and direction, a full scan of image is done to remove any unwanted pixels which may not constitute the edge.

3. Image Segmentation In OpenCv:

We use the marker based watershade algorithm for image segmenation.Thresholding the input image.To remove any noise, we use morphologyEx(). we can deduce, this maximizing operation causes bright regions within an image to grow in dilation.

b.Investigation of current Project and Related work:

Watershed segmentation is a region-based technique that utilizes image morphology. It requires selection of at least one marker ("seed" point) interior to each object of the image, including the background as a separate object. The markers are chosen by an operator or are provided by an automatic procedure that takes into account the application-specific knowledge of the objects. Once the objects are marked, they can be grown using a morphological watershed transformation. A very intuitive description of watersheds can be found in. To understand the watershed, one can think of an image as a surface where the bright pixels represent mountaintops and the dark pixels valleys. The surface is punctured in some of the valleys, and then slowly submerged into a water bath. The water will pour in each puncture and start to fill the valleys. However, the water from different punctures is not allowed to mix, and therefore the dams need to be built at the points of first contact. These dams are the boundaries of the water basins, and also the boundaries of image objects.

Watershed analysis has proven to be a powerful tool for many 2D image-segmentation applications. An example of segmentation of covid-19 lung infection from CT images. Higgins and Ojard applied a 3D extension of the watershed algorithm to volumetric images.

Requirement Analysis

a. System Requirement

1. Software and Hardware requirements:

- **Hardware requirement:**

  a. Processor: Intel dual core i3

  b. Storage:4GB RAM and 500GB Hard Disk Storage.

- **Software requirements:**

  a. Operating System - Microsoft Windows 7 / 8 / 10.
  b. Python 3.7
  c. Xampp

2. Functional Requirements :

2.1 Registration of patient:

In the start of the project first we have to make registration of a new patient. Registration part include-

Patient name

City

Contact Number

Address

2.2 Detect Infection:

After completing the successful registration in Detect infection tab we have to search for the image to upload the CT image and enter the id of patient. After uploading the image system gives the infected part of the image.

2.3 Search patient:

After getting the infected part patient wants to see the percentage of the infected part then they have to go to search tab and enter the patient id or patient name. After that system will display all the information related to patient and Percentage of infected part.

2.4 Gray scale conversion:

A pixel color in an image is a combination of three colors Red, Green, and Blue (RGB). The conversion of a color image into a grayscale image is converting the RGB values (24 bit) into grayscale value (8 bit). Various image processing techniques and software applications converts color image to grayscale image.

2.5 Edge detection:

Edge detection allows users to observe the features of an image for a significant change in the gray level. It highlight the edges of the CT image.

2.6 Segmentation:

The watershed algorithm is a classic algorithm used for segmentation and is especially useful when extracting touching or overlapping objects in images.

3. Performance Requirements:

1. The number of terminals to be supported.

2. The number of the simultaneous users to be supported.

3. Many type of information to be handled.

4. Software system attributes:

1. **Reliability**

   Newly updated version is easily available to all users after adding extra features to it.

2. **Availability**

   System is available to that users who are having laptops and personal computers and who are working in medical field.

3. **Security**

   System is available to only authenticated and authorized users. And system also maintain the privacy of patient details.

4. **Maintainability**

   Maintenance of system easily available to users.

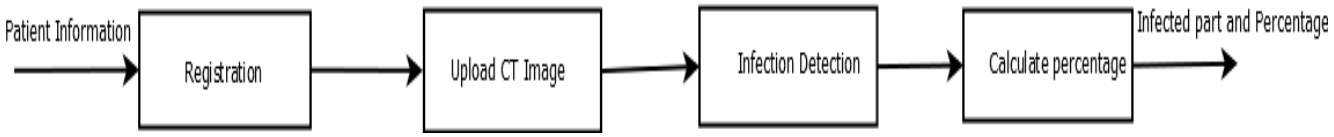
5. **Portability**

   System is portable to any Operating system you just need to install python on machine.

System Design

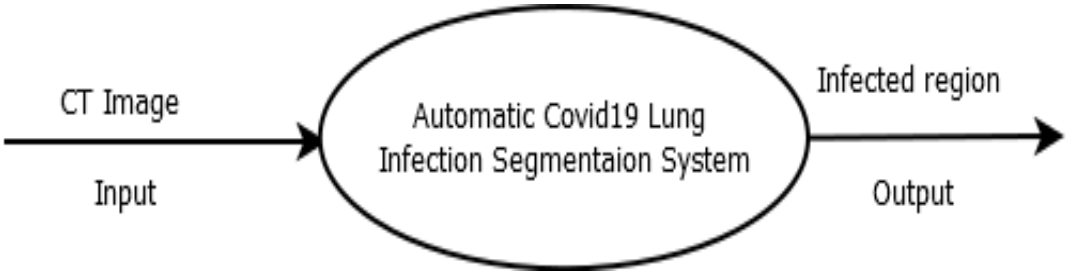
Architectural Design of System :

a. Architectural Design of Automatic COVID-19 Lung Infection Segmentation from CT Images

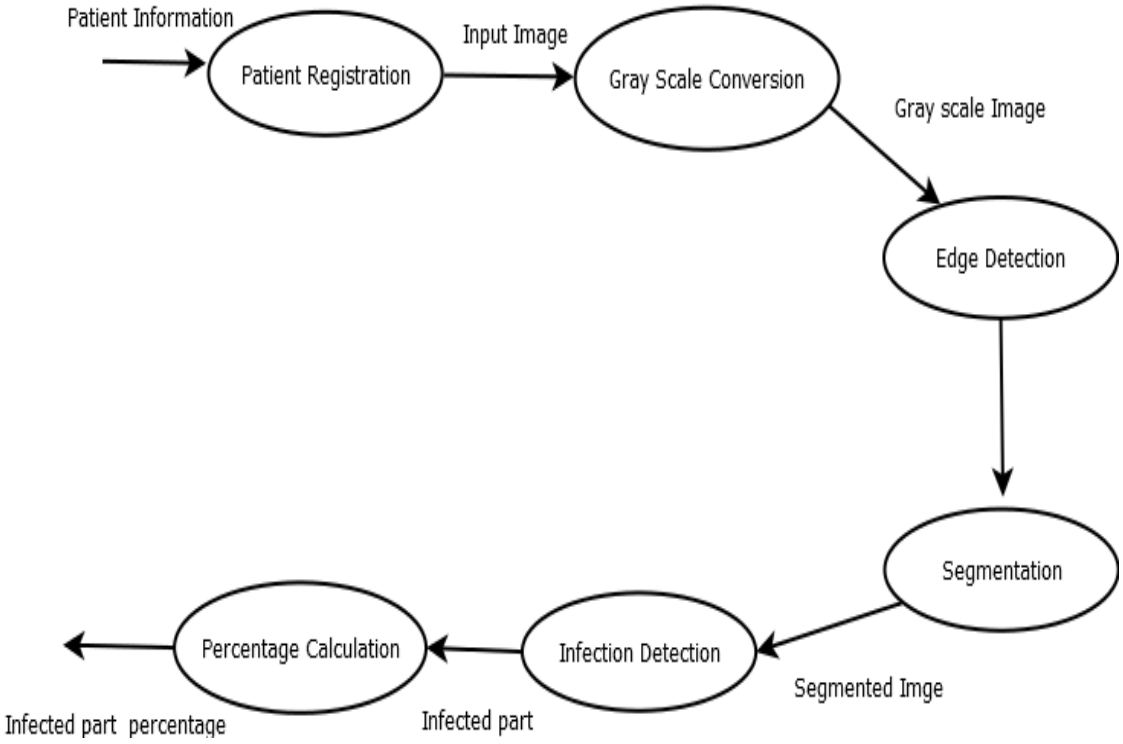


System Modelling:

1) Data Flow Diagram

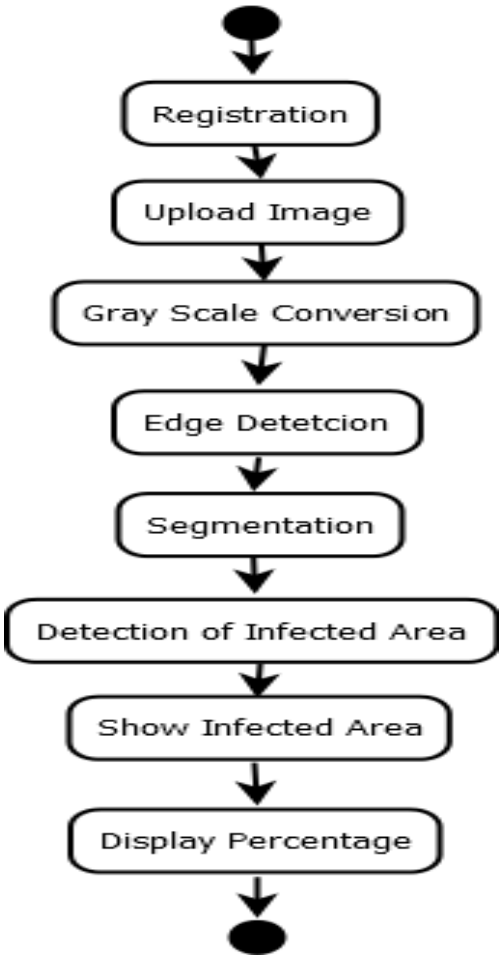


Level-0 Data Flow Diagram

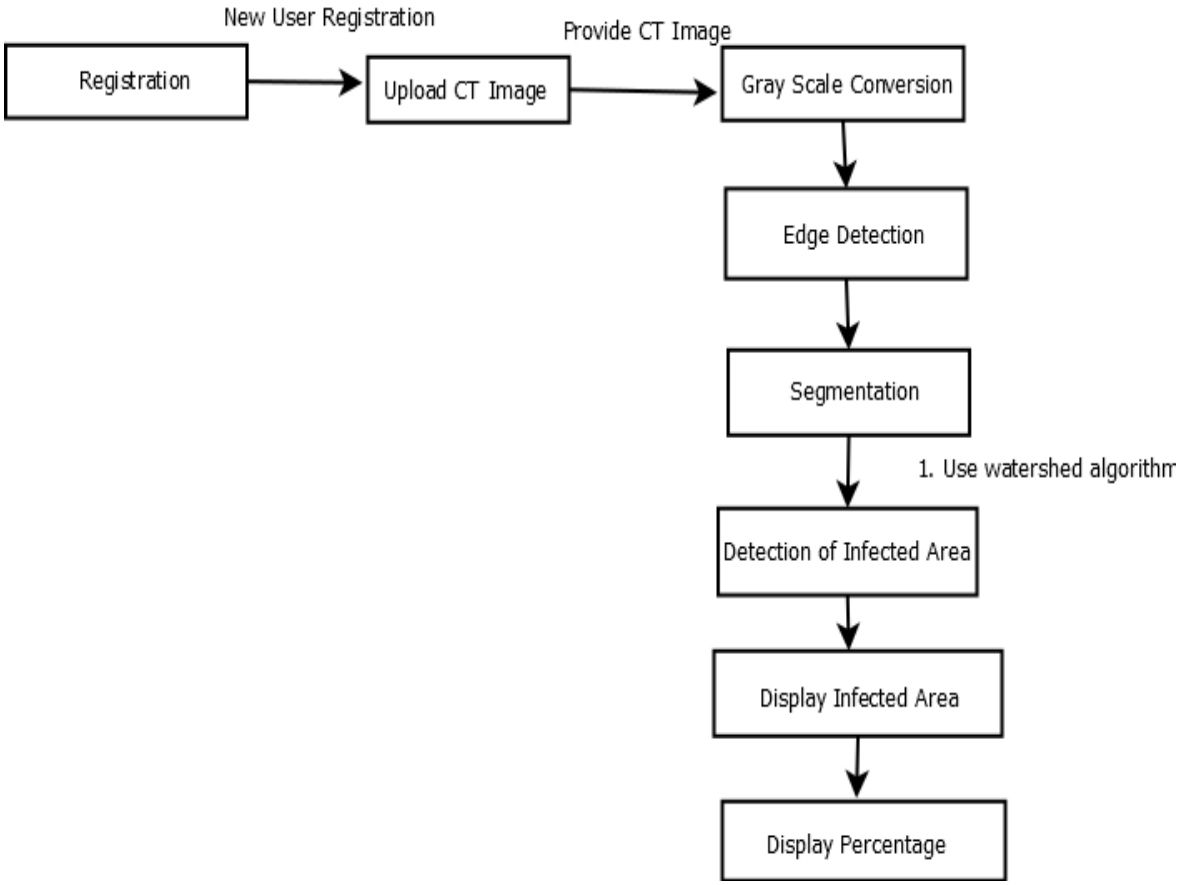


Level-1 Data Flow Diagram

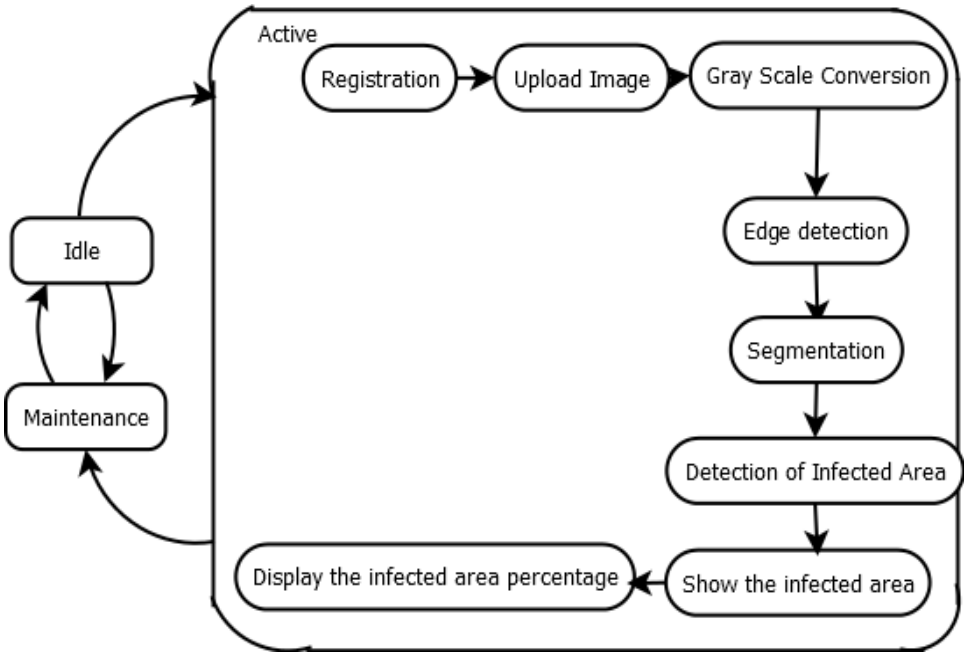
2) Activity Diagram



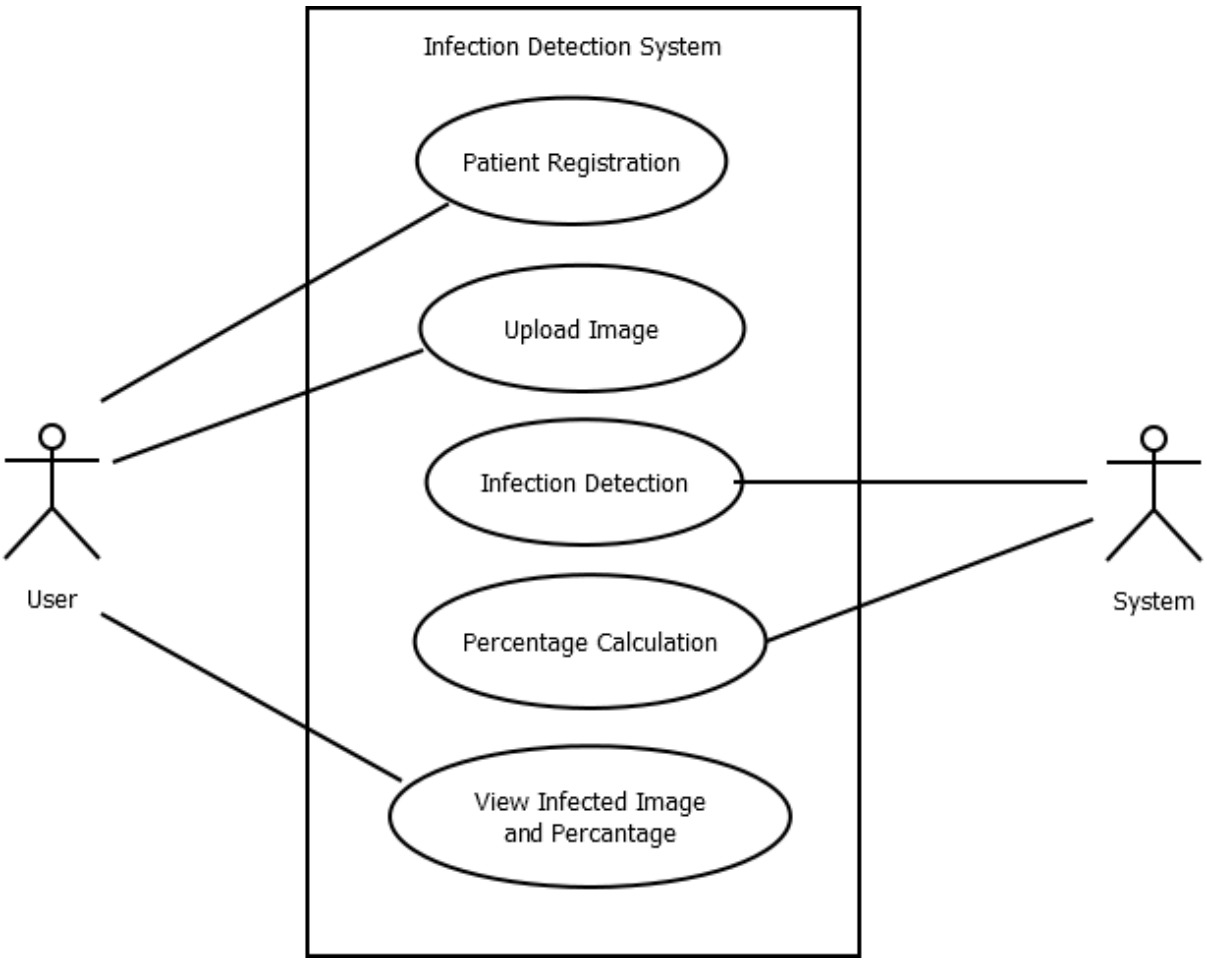
3) Collaboration Diagram



4) State Diagram



5) Use Case Diagram



Implementation

A. Detection of COVID-19 Lung Infection from CT Images

Our objective is to find the lung infection by highlighting infected part of lung in CT image and calculating percentage of covid-19 risk.

1. Get the image by using File Explorer in Python using Tkinter

Select the CT image to detect the lung infection and its percentage.

We have to import the filedialog module from Tkinter. The File dialog module will help you open, save files or directories.
In order to open a file explorer, we have to use the method, askopenfilename().

#input1 = filedialog.askopenfile(initialdir="/")

input1 =filedialog.askopenfilename(initialdir = "/",title = "Select a File",filetypes = (("[PNG]","*.png*"),("[JPG]","*.jpg*"),("all files","*.*")))

2. Gray scale image conversion and thresholding

Grayscaling is the process of converting an image from other color spaces e.g RGB, CMYK, HSV, etc. to shades of gray. It varies between complete black and complete white.

In this method, the pixel values of a grayscale image are assigned one of the two values representing black and white colors based on a threshold. So, if the value of a pixel is greater than a threshold value, it is assigned one value, else it is assigned the other value. We will apply image thresholding on the output image of the previous step:

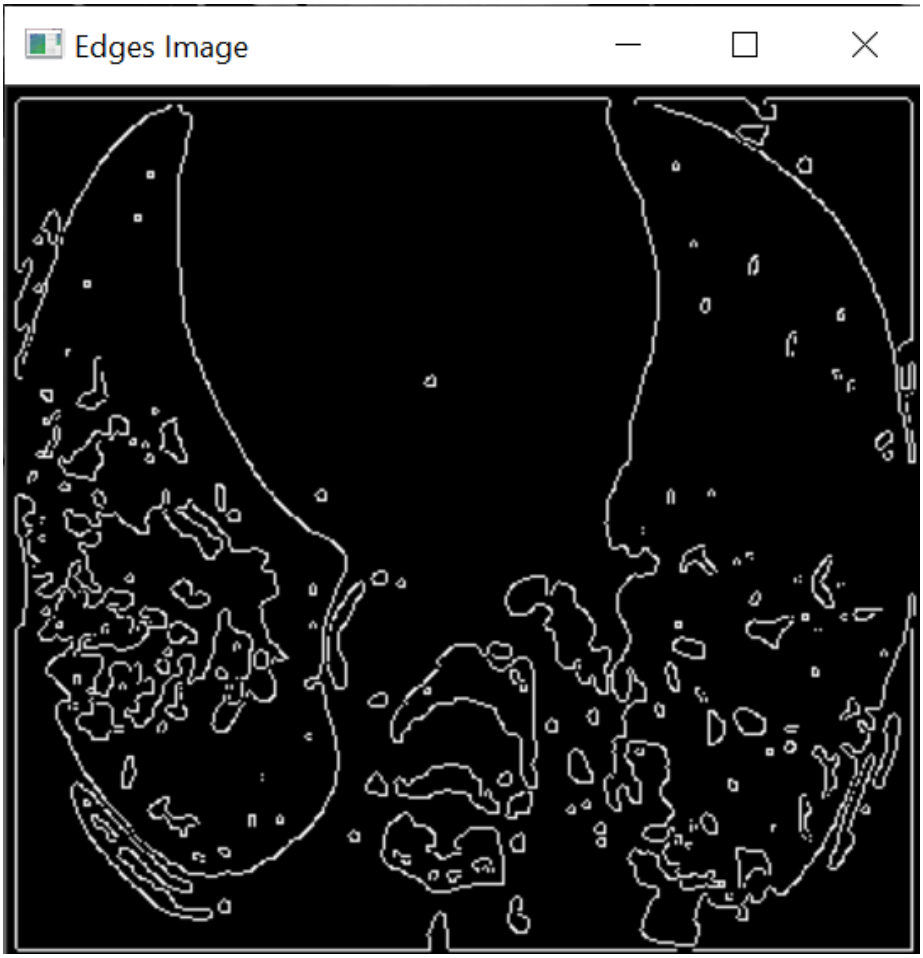
```
# Image operation using thresholding
  img = cv2.imread(aa1.get())
  gray1 = cv2.cvtColor(img, cv2.COLOR_BGR2RGB)

  test =ImageTk.PhotoImage(Image.fromarray(gray1))
  label1 = Label(newWindow,image=test)
  label1.image = test
  label1.grid(row = 3,column = 0,columnspan=3)
```

3. Perform Canny Edge detection

The canny edge detector is an edge detection operator that uses a multi-stage algorithm to detect a wide range of edges in images.

```
#To Get Accuracy Highlight Dark and Light area
    light_val = (150, 150, 150)
    dark_val = (200, 200, 200)
    mask = cv2.inRange(gray1, light_val, dark_val)
    result = cv2.bitwise_and(gray1, gray1, mask=mask)
#Edge Detection
    edges = cv2.Canny(mask,130,300)
```

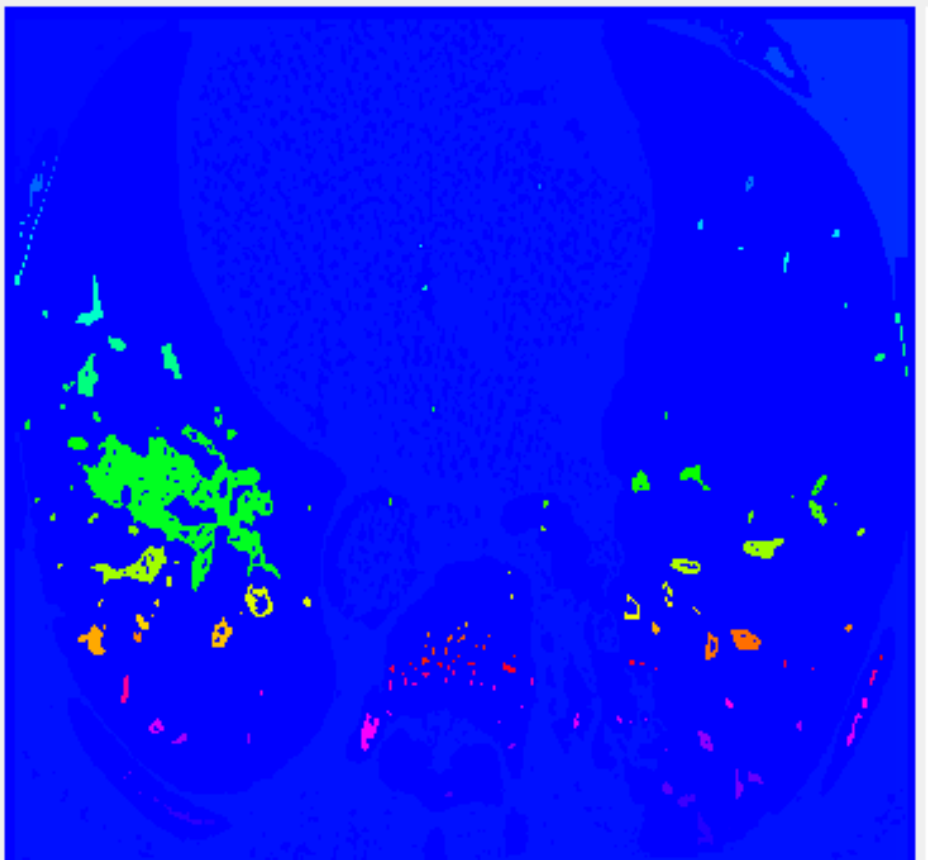


4. Masking of Gray Image and Edge Image create new image object1

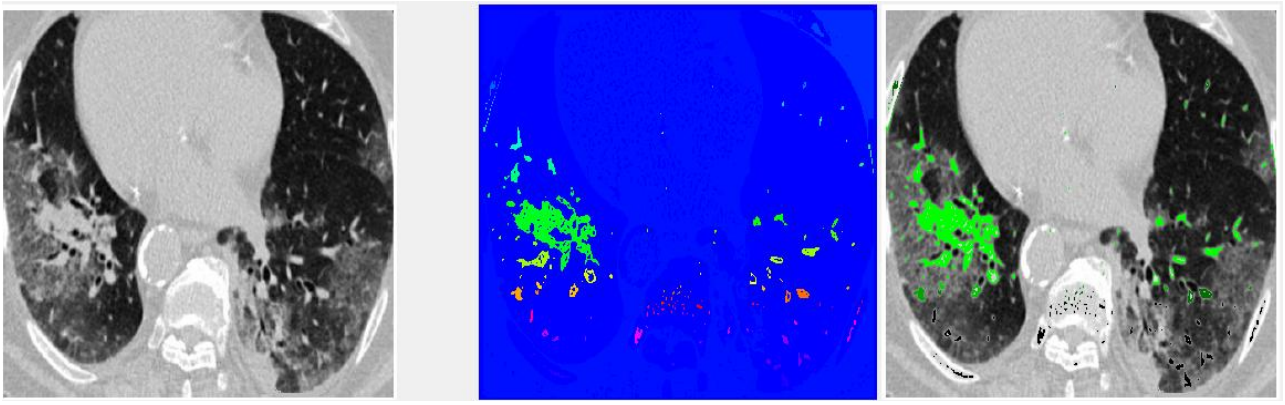
object1 image Remove Noise unwanted part of image and Select COVID infection based on Gray scale color intensity and create new image object2.

object2 image colorize with different Water shade to different edge part of image and create new image object3.

```
a = Label(newWindow ,width=25,text = "Select Image").grid(row = 1,column = 0,pady=(10, 10),padx=(5, 0))
aa1 = StringVar()
a1 = Entry(newWindow,width=30,textvariable=aa1).grid(row = 1,column = 1,pady=(10, 10))
```

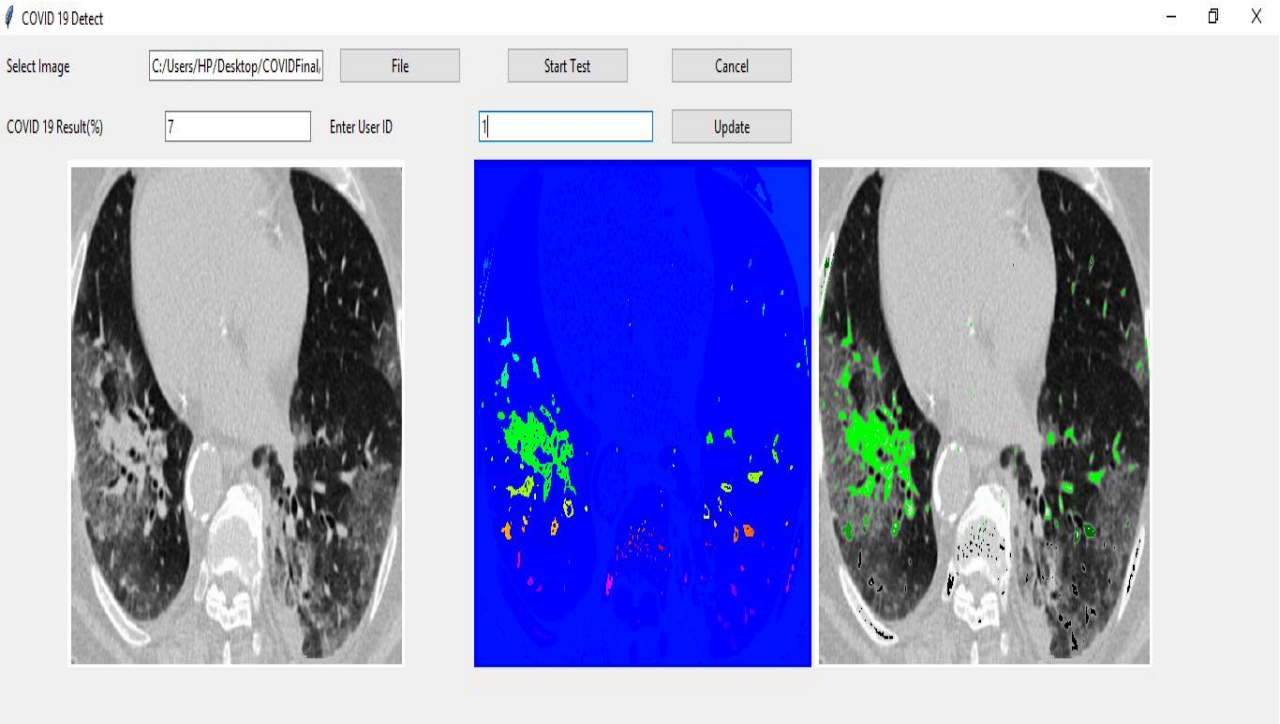


5. Masking of object3 with Original image to get COVID infection part on Original image



```
res2 = cv2.bitwise_and(img, img, mask = mask1)
        final_output = cv2.addWeighted(res1,1,res2,1,0)
        test = ImageTk.PhotoImage(Image.fromarray(final_output))
        label3 = Label(newWindow,image=test)
        label3.image = test
        label3.grid(row = 3,column = 6,columnspan=3)
        Mess=""
        if perval>=10:
            Mess="High Risk"
        elif perval>=5 and perval<10:
            Mess="Min Risk"
        elif perval>=1 and perval<5:
            Mess="Low Risk"
        else:
Mess="No Risk"
```

messagebox.showinfo(title="COVID Test Result", message=Mess+", "+str(perval)+"% Infection Detected")



So here finally we get the CT image with infected part of lung due to covid-19 and percentage of risk of lung infection.

Integration And Testing

Integration and Testing-

Testing Performed:-

Test Plan:

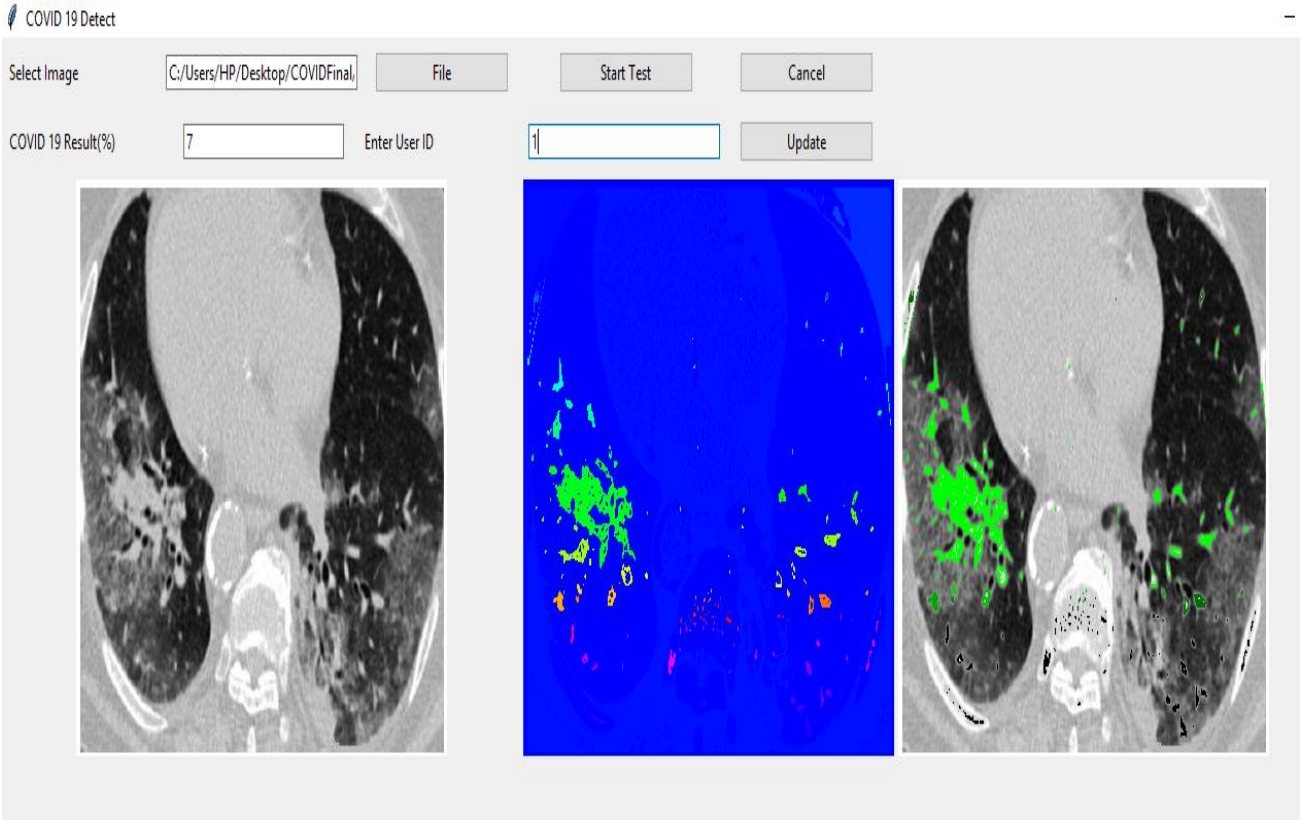
a. Time:

This project should detect the infected areas in lung automatically and gives the percentage of infected area by using the lung CT images. There are different methods implemented in this project, time taken by these methods are as follows:

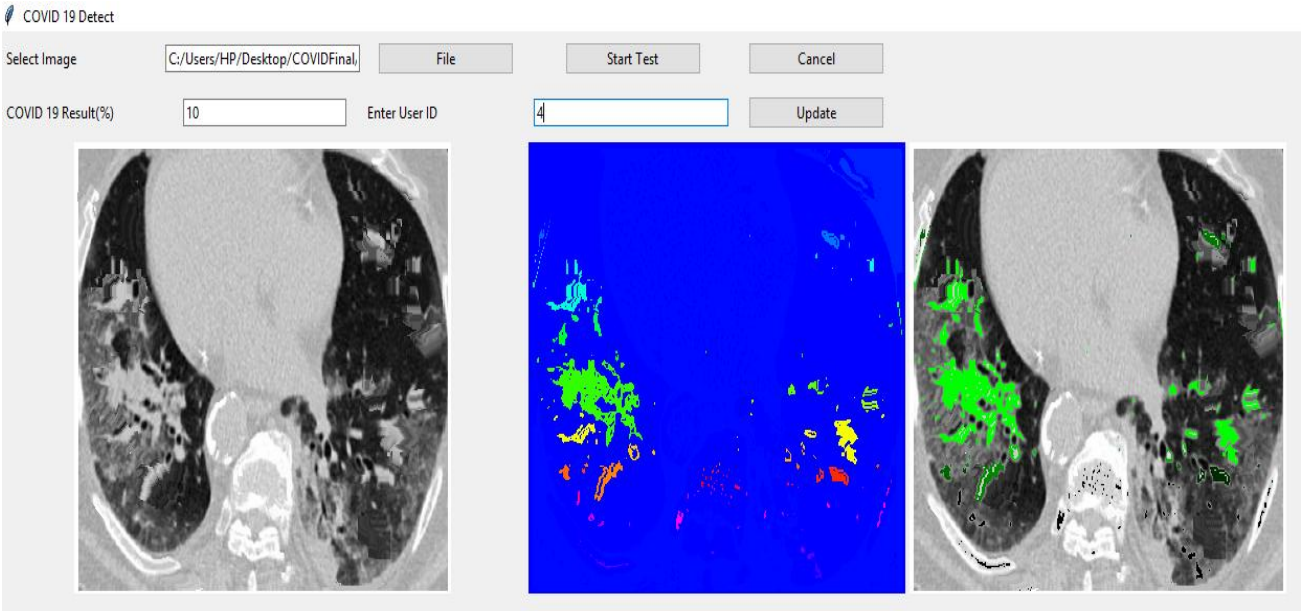
Methods	**Time Required(Avg)**
System Booting	20 second
Grey scale Conversion	4 second
Edge Detection	5 second
segmentation	10 second
Display infection %	6 second

b. Output:

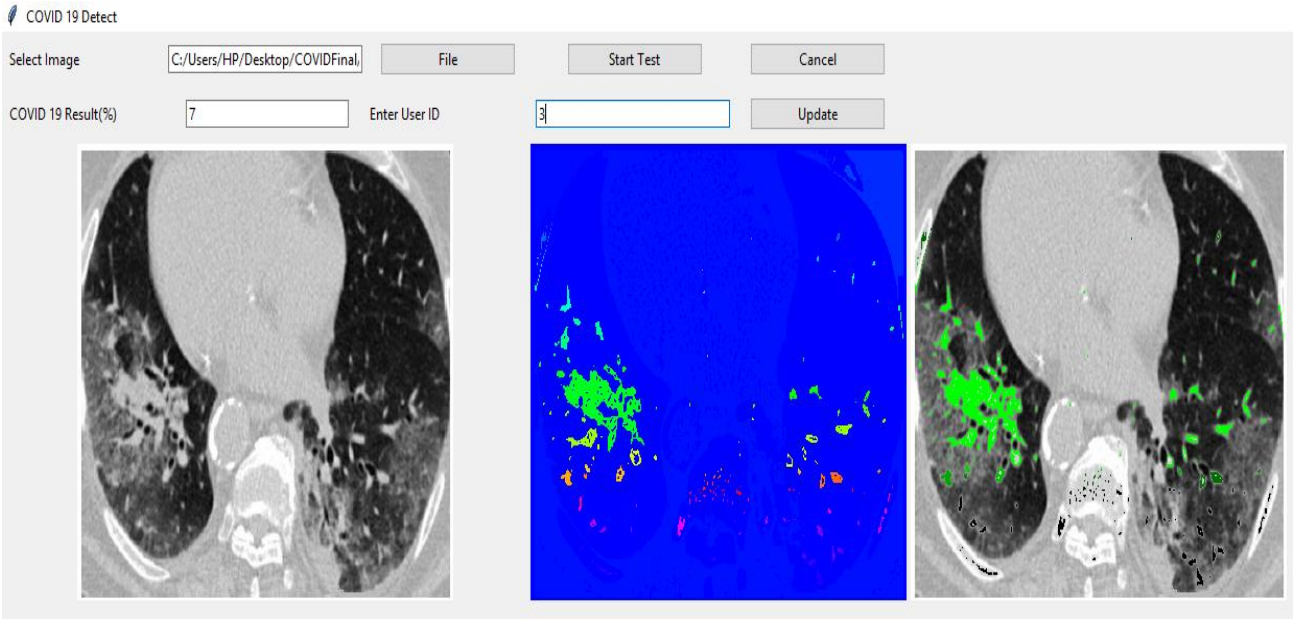
Sample 1:



Sample 2:



Sample 3:


COVID 19 Detect
Select Image
C:/Users/HP/Desktop/COVIDFinal/
File
Start Test
Cancel
COVID 19 Result(%)
7
Enter User ID
3
Update

Performance Analysis

Performance Analysis-

The following table shows the performance of each method:

Methods	**Time Required(Avg)**
System Booting	20 second
Grey scale Conversion	4 second
Edge Detection	5 second
segmentation	10 second
Display infection %	6 second

Conclusion: We have proposed a deep learning based model to detect and classify COVID-19 cases from CT images. Our model is fully automated with an end-to-end structure without the need for manual feature extraction.

The performance of the developed model is assessed by expert radiologists. This system can be used in remote places in countries affected by COVID-19 to overcome a shortage of radiologists. Also, such models can be used to diagnose other chest-related diseases including tuberculosis and pneumonia.

Applications

- **Supporting system for medical diagnostic:**

  Trained system to look for abnormalities and classify areas of the lung scan as either infected or not infected – this is called 'segmentation'. The tool can judge the severity of the disease by comparing the extent of infected area with healthy area," the release explained.

- **Automated Detection:**

  System can 'read' the chest CT scans of COVID-19 patients, and using a segmentation technique , estimate how much damage has been caused in the lungs by searching for specific abnormal features. Such a tool can provide automated assistance to doctors and therefore help in faster diagnosis and better management of COVID-19, it says.

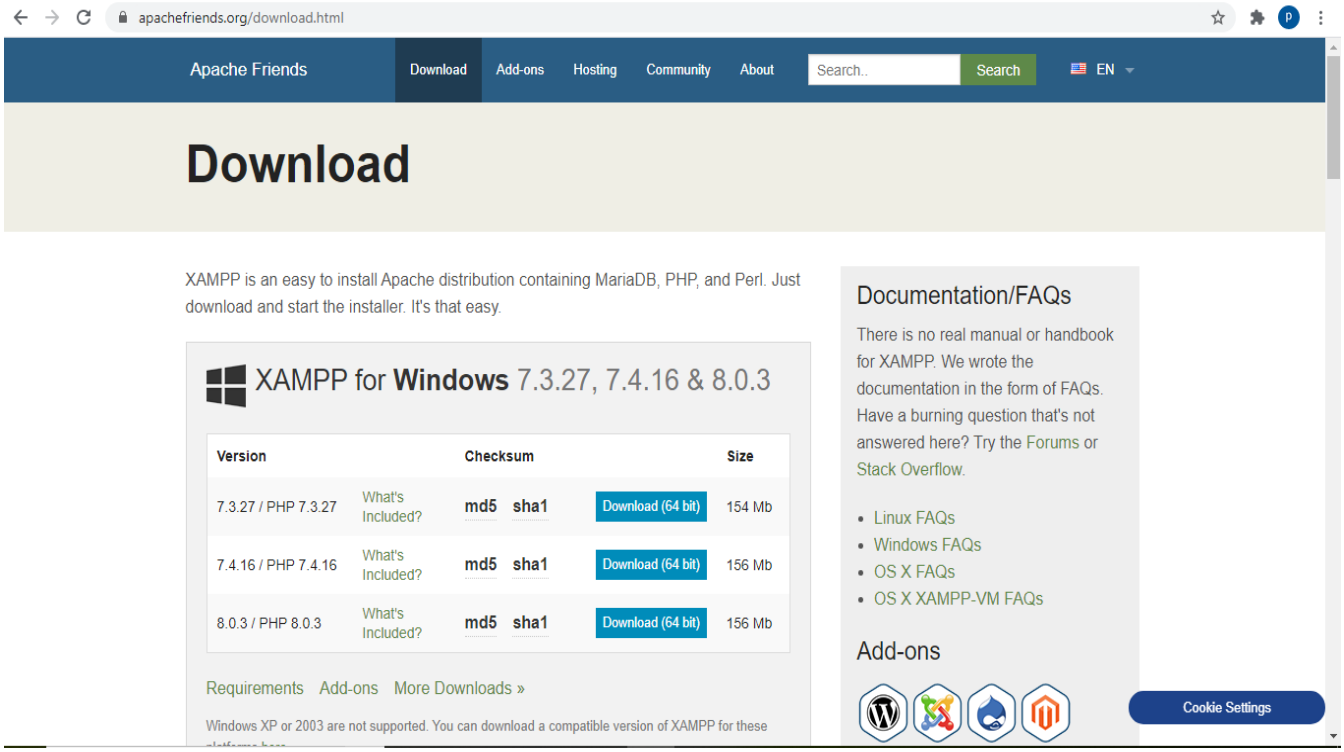
Installation Guide

And

User Manual

Installation Sequence for XAMPP

1. XAMPP setup

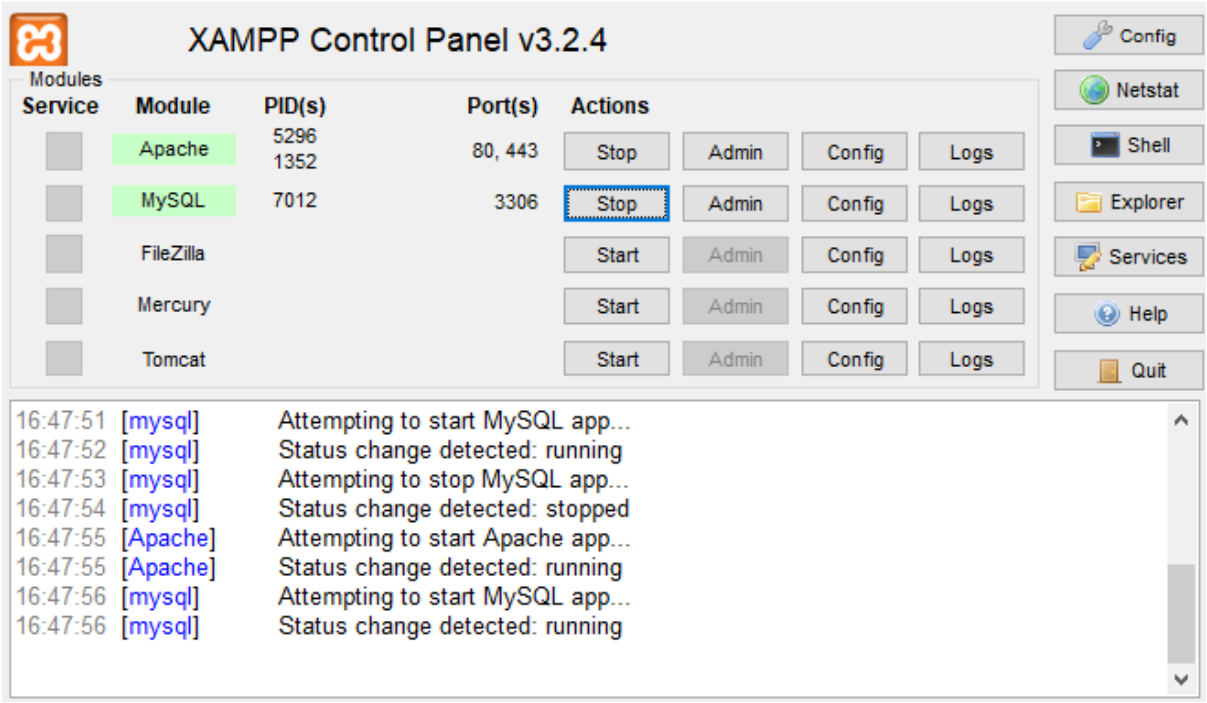


Step 1: Download and install XAMPP

Go to the below link and download the XAMPP for windows version 7.3.27

https://www.apachefriends.org/download.html

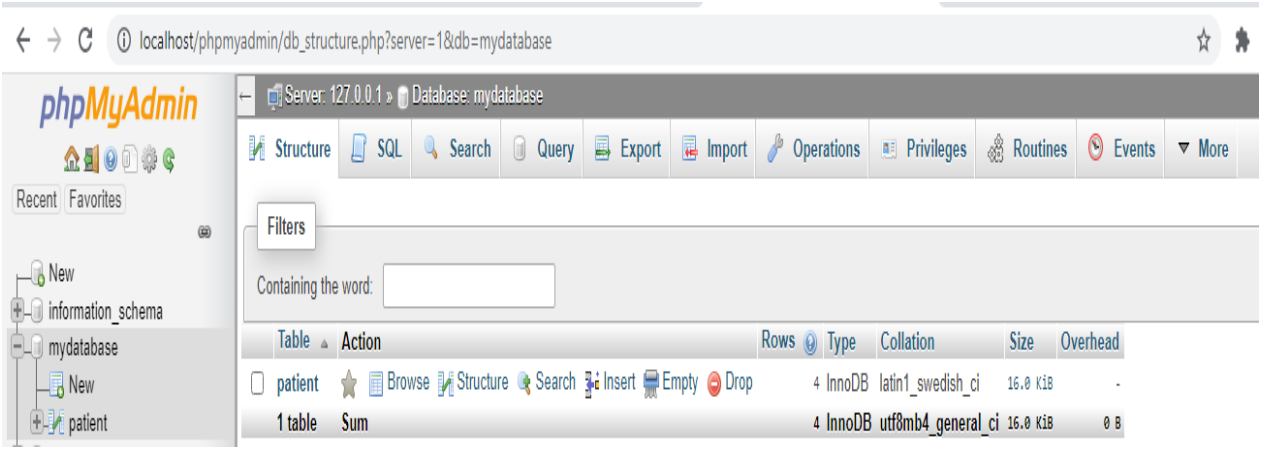
Step 2: Open XAMPP and start Apache and MySQL



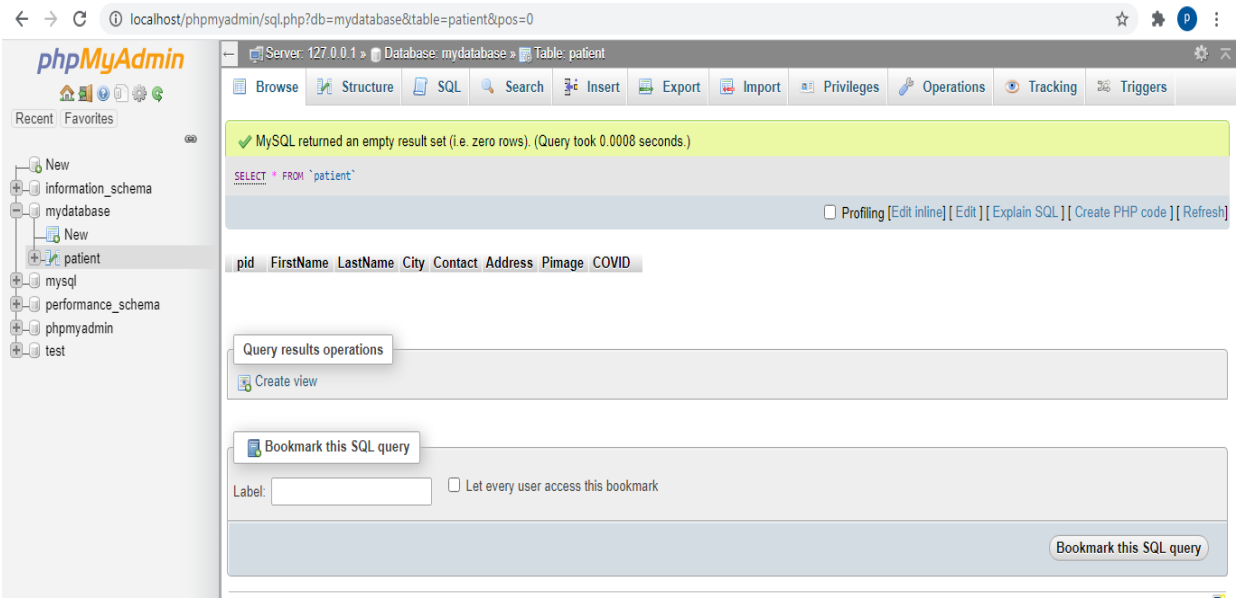
Follow below steps to create database table for storing patient information :

i. Open browser and paste below link on it.
http://localhost/phpmyadmin/

ii. Click on new and create database for storing patient information



iii. Add columns in patient table like Patient id, Name , Address , City, Contact Number, Patient CT image and Percentage of Covid-19



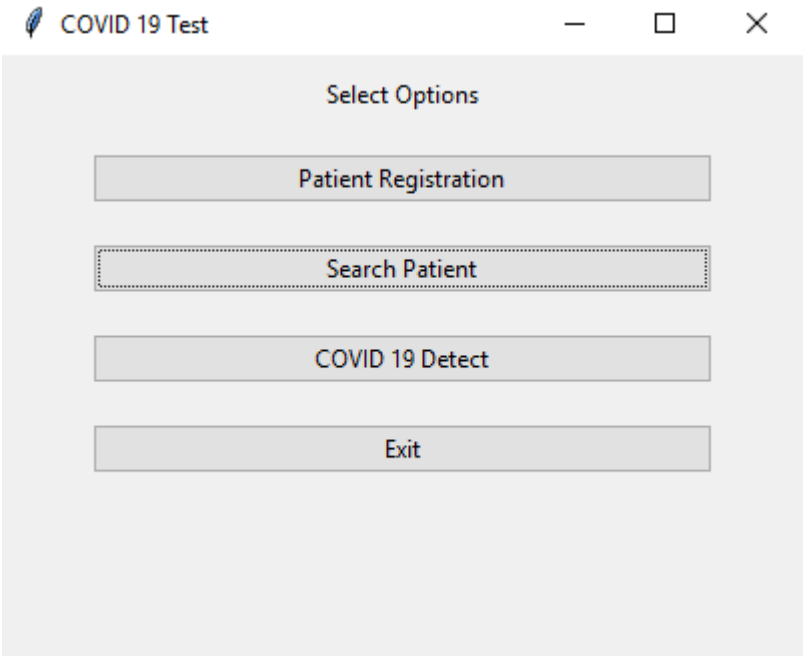
How to check patient is affected by covid-19 using lung CT images:

Step1: Download and install Python 3.7.0

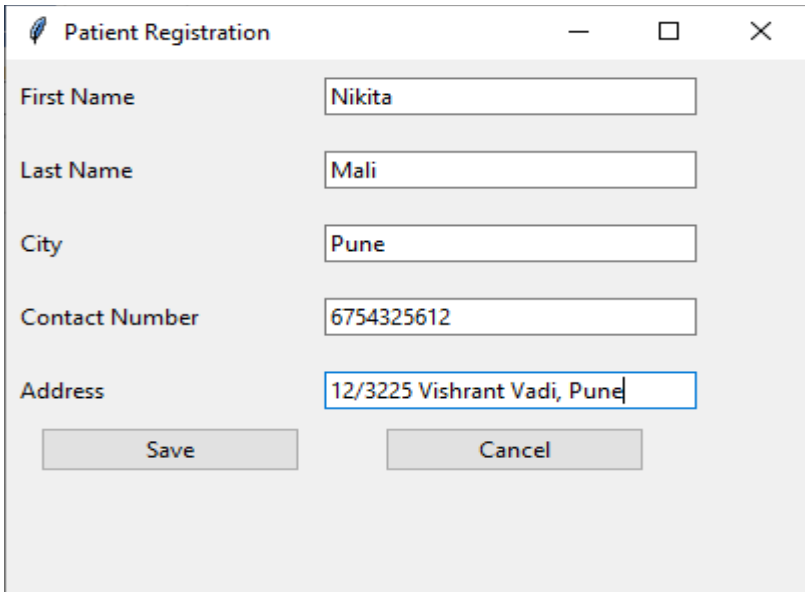
https://www.python.org/downloads/release/python-370/

Step2: Open Python IDE and run Main.py file

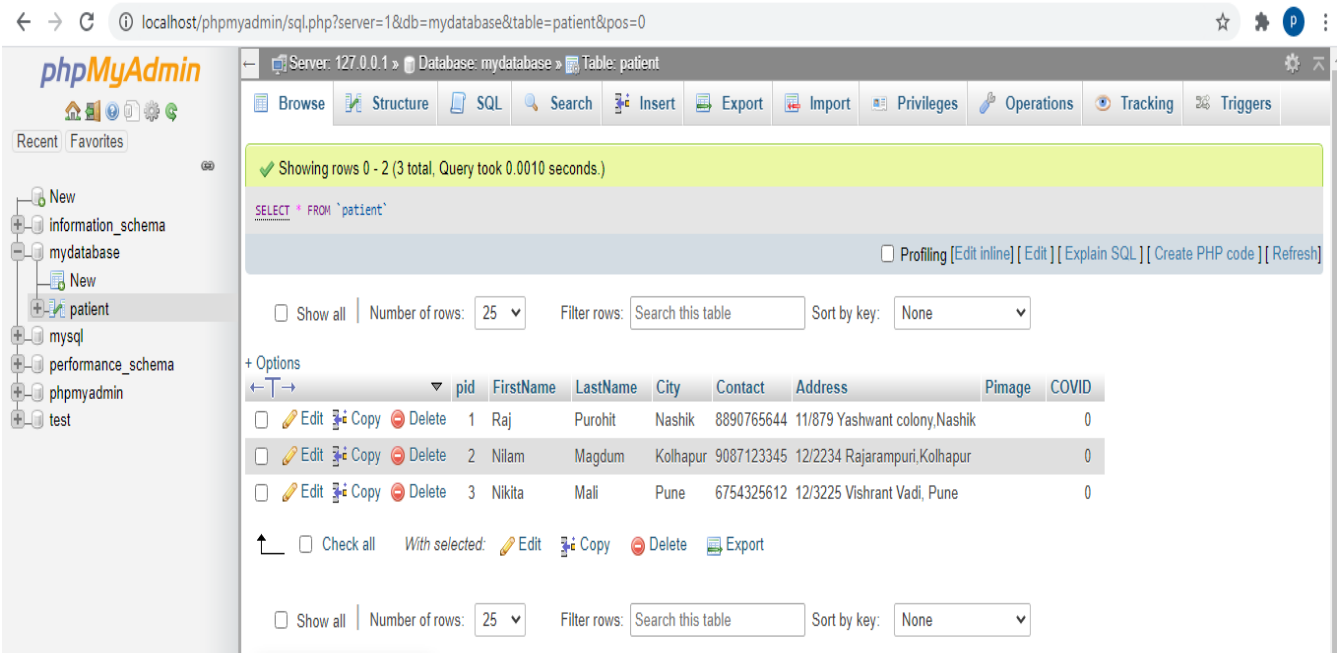
When we run file then it will open following window:



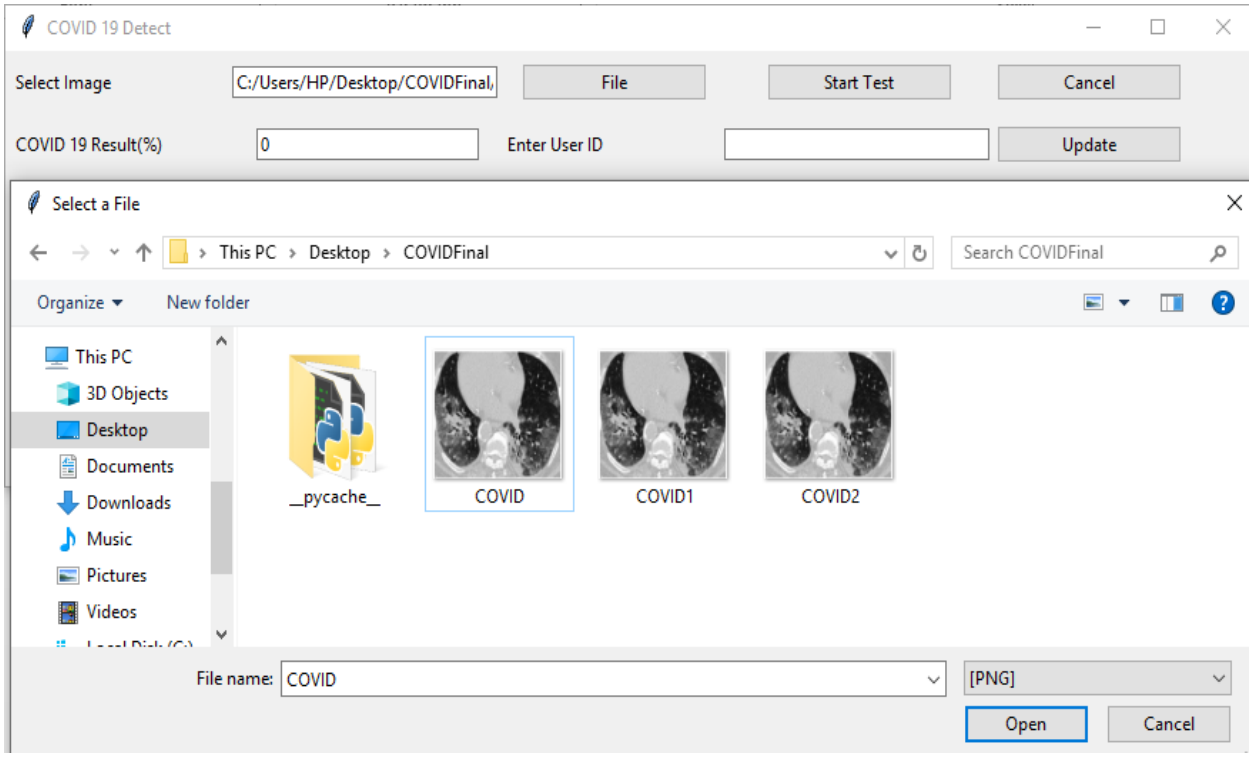
Step3: Click on Patient Registration to store patient information in database



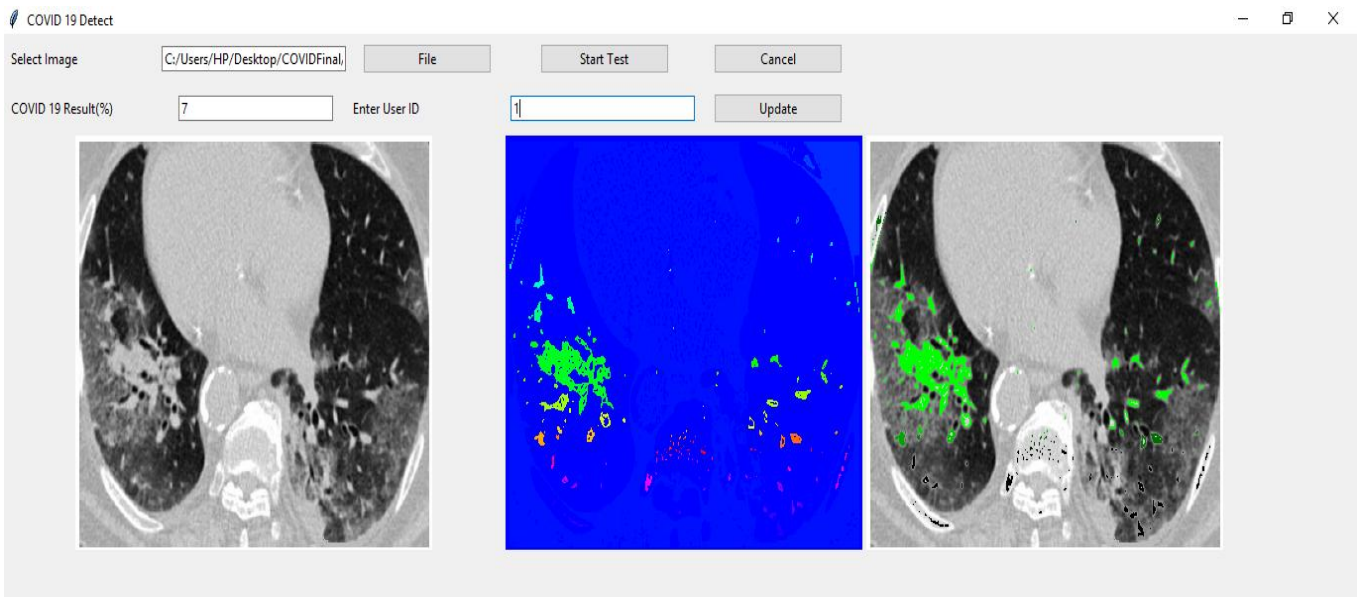
When we click on save button the above information is stored in patient table.



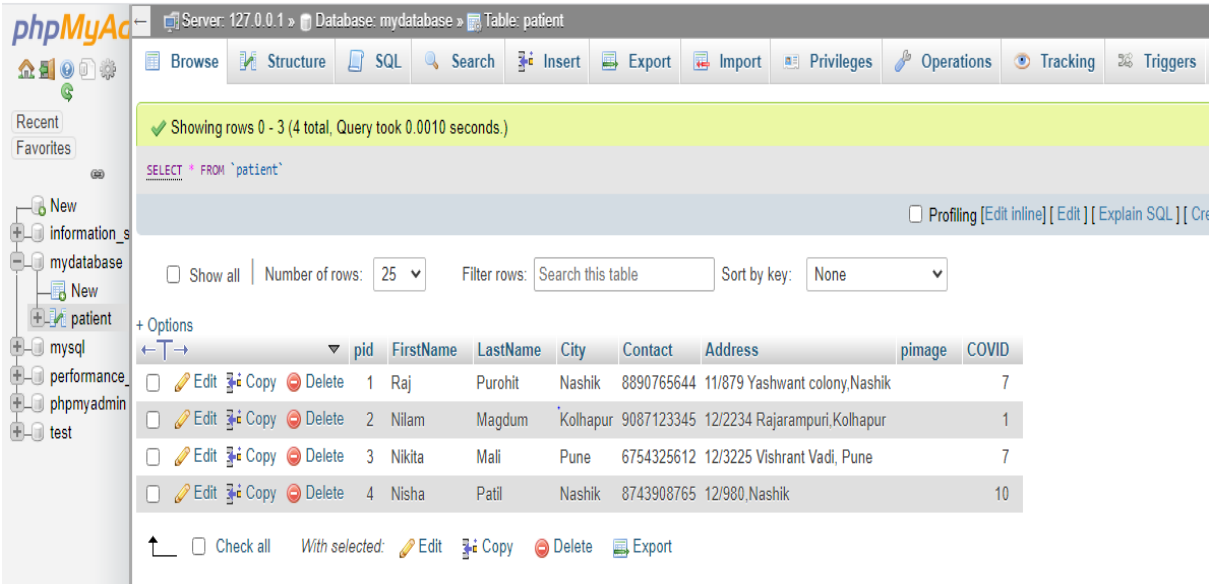
Step4: Click on COVID 19 detect button and choose the lung CT image of patient and start testing to find out percentage of infection in lungs.



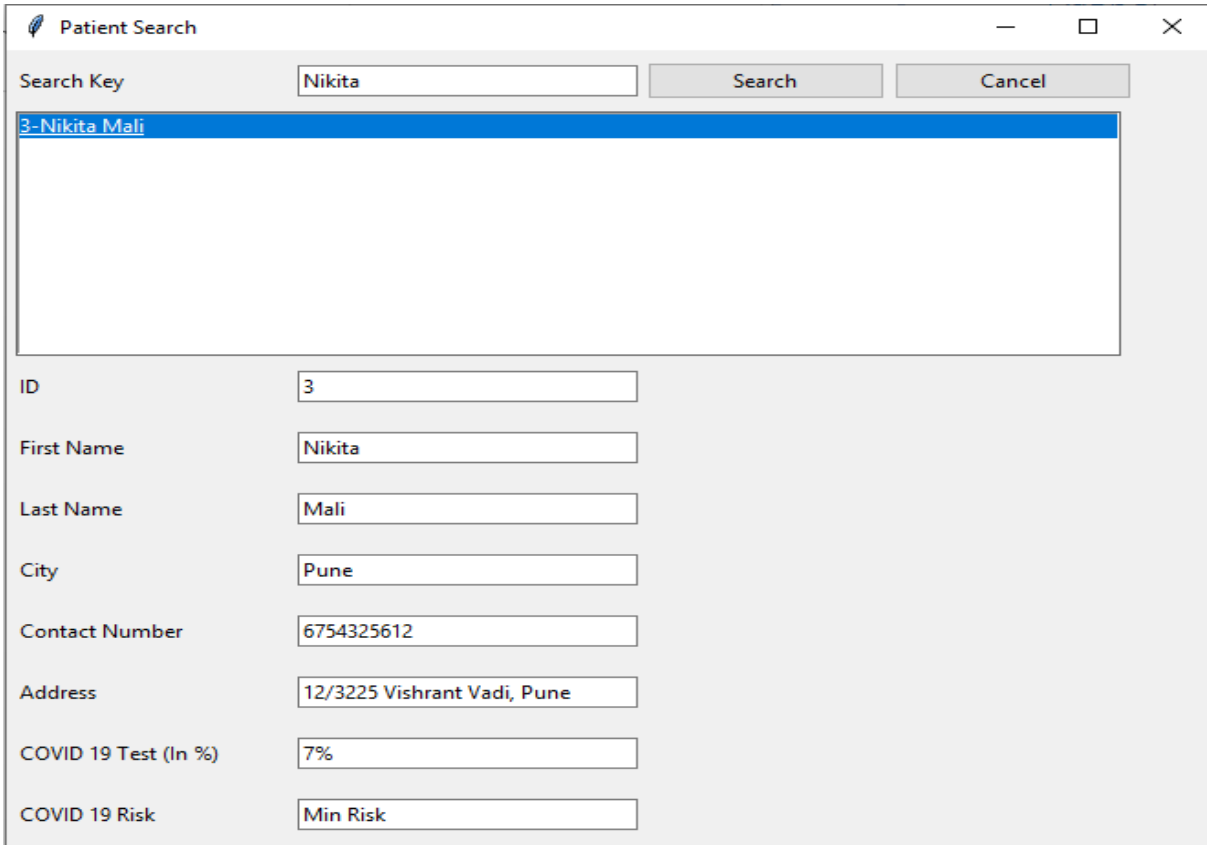
After clicking on start test we get the percentage of covid-19 risk and infected regions in CT image.



By entering User Id that is Patient id which is present in our database we can store percentage of COVID-19 risk of that respected patient.



Step5: We can also search Patient and it will display the information of that person with respect to percentage of covid 19 lung infection and Risk level.



Cost Estimation

- **Hardware Cost:**

Hardware	**Cost**
Computer System	Rs. 40000/-
Internet	Rs. 800/-
Light Source	Rs. 300/-
Total	Rs. 41100/-

In this project the Cost Estimation based on COCOMO (Constructive Cost Model) the formula for the this Model is follows

Effort = Constant × (Size) scale factor× Effort Multiplier

Effort in terms of person-months

Constant: 2.45 in 1998 based on Organic Mode –

Size: Estimated Size in KLOC –

Scale Factor: combined process factors

Effort Multiplier (EM): combined effort factors

The basic COCOMO equations take the form

Effort Applied (E) = ab(KLOC)b b [man-months]

Development Time (D) = cb(Effort Applied)d b [months]

People required (P) = Effort Applied / Development Time [count]

Where, KLOC is the estimated number of delivered lines (expressed in thousands) of code for project. The coefficients ab, bb, cb and db are given in the following table.

Ethics

Declaration of Ethics

As A Computer Science & Engineering Student, I believe it is Unethical To,

1. Surf the internet for personal interest and non-class related purposes during classes
2. Make a copy of software for personal or commercial use
3. Make a copy of software for a friend
4. Loan CDs of software to friends
5. Download pirated software from the internet
6. Distribute pirated software from the internet
7. Buy software with a single user license and then install it on multiple Computers
8. Share a pirated copy of software
9. Install a pirated copy of software

References

[1] Coronavirus COVID-19 Global Cases by the Center for Systems Science and Engineering at Johns Hopkins University. Accessed: Apr. 2, 2020. [Online]. Available: https://coronavirus.jhu.edu/map.html

[2] T. Ai et al., "Correlation of chest CT and RT-PCR testing in coronavirus disease 2019 (COVID-19) in China: A report of 1014 cases," Radiology, vol. 2019, Feb. 2020, Art. no. 200642.

[3] G. D. Rubin et al., "The role of chest imaging in patient management during the COVID-19 pandemic: A multinational consensus statement from the fleischner society," Radiology, Apr. 2020, Art. no. 201365.

[4] F. Shi et al., "Review of artificial intelligence techniques in imaging data acquisition, segmentation and diagnosis for COVID-19," IEEE Rev. Biomed. Eng., early access, Apr. 16, 2020, doi: 10.1109/RBME.2020.2987975.

[5] Y. Fang et al., "Sensitivity of chest CT for COVID-19: Comparison to RT-PCR," Radiology, Feb. 2020, Art. no. 200432, doi: 10.1148/radiol.2020200432.

[6] M.-Y. Ng et al., "Imaging profile of the COVID-19 infection: Radiologic findings and literature review," Radiol., Cardiothoracic Imag., vol. 2, no. 1, Feb. 2020, Art. no. e200034.