Institute Of Management & Career Courses (IMCC) ,PUNE

A

Project Report

On

“Pneumonia Detection System Using CNN”

Submitted by

Raj Bhatted
Anuj Gaikwad
Utkarsh Bhangle
Suraj Biyani

MCA First Year

Under the Guidance of

Prof. Ravindra Vaidya

Institute of Management & Career Courses (IMCC), PUNE
For Academic year 2020-2021

TABLE OF CONTENTS

1. Introduction.

Pneumonia accounts for around 16% of all deaths of children under five years worldwide, being the world's leading cause of death among young children. The main reason behind it is that many people don't get to know about it in early stages and when they realize it's too late. But, if this disease is detected at the early stages, the probability of it being cured increases. Main aim of the system is to build a Diagnostic System for prediction of Pneumonia. By using the concept of image processing with machine learning. Proposed system will detect the disease from CT scan and Chest X-ray images. CNN algorithm is then applied to classify the images for detection of respiratory diseases. After detection of disease, a report will be generated.

PROBLEM DEFINITION

Detection of Pneumonia and Hanta virus from Chest CT Scanned Images using CNN (Convolutional Neural Network).

2. Literature Surveys

- **Pneumonia Detection Using Deep Learning Based on Convolutional Neural Network : Luka Račić, Tomo Popović, Senior Member, IEEE, Stevan Čakić, Stevan Šandi**

As per the paper, Artificial intelligence and machine learning are increasingly finding their application in medicine. This paper describes the use of MLA to process chest X-ray images. The model uses the classification problem which detects whether the chest X-Ray shows changes consistent with pneumonia or not, and classifying the X-ray images in two groups depending on the detection results. CNN based algorithm was used to process chest X-ray images which helped in detecting pneumonia.

The dataset used for this research was taken from Guangzhou Women and Children‘s Medical Centre, Guangzhou which is available on Kaggle. The dataset contained 5856 images of chest X-rays in JPEG format. For the purpose of the experiment, the dataset was divided into three parts (80/10/10) split. That means 80% of the images were used as training data, 10% as validation data and 10% as test data.

The original images were pre-processed into grayscale from RGB and resized to 200x200 pixels. All of this can be performed with Kera’s pre-processing tools. The tools used to create the model were: Numpy, Pandas, Kera’s, Jyupter notebook, matplotlib and seaborn. The image classification was done using CNN which is a Machine Learning Algorithm. Where CNN represent a huge breakthrough in image recognition and classification, where they are most commonly used.

ReLu activation function was used in this experiment because ReLU behaves very well in deep neural networks, also the Dropout method was used to boost the performance.

Evaluation of this model is done by analysing the various parameters such as training accuracy and loss, validation accuracy and loss and of course model accuracy. The model can also be represented by building a confusion matrix where Y-axis holds the predicted value, and the X-axis holds the true values where the trained model was accurately predicted as images of X-rays with pneumonia and the accuracy of this model is predicted to be 88.90%.

To conclude this research paper, it describes the use of Deep Learning in order to classify chest X-rays. Even if the accuracy of this model is high there is a possibility of overfitting due to the size of the dataset. It is important for this model to gather as much data as possible.

- **Deep Learning for Automatic Pneumonia Detection : Alexandr A. Kalinin University of Michigan Ann Arbor, MI 48109 USA, and Shenzhen Research Institute of Big Data, Shenzhen 518172, Guangdong, China**

This research paper about Deep Learning for Automatic Pneumonia Detection states that it is the leading cause of death among young children and one of the top causes worldwide. More than 1 million adults are hospitalized with pneumonia and around 50,000 die from the disease every year in the US alone. The pneumonia complicating recent coronavirus disease 2019 (COVID-19) is a life-threatening condition claiming thousands of lives in 2020

The dataset of pneumonia detection includes more than 26000 chest x-rays publicly provided to the Institutes of health clinical center. Where each image is labelled with names like “Normal”, “No lung Opacity”, “lung opacity”. “Normal” class contains the dataset of healthy patients.” Lung opacity” class contains a dataset of patients with fuzzy white clouds in the lungs.” No Lung opacity” class includes the dataset of unhealthy Lungs. The models without pre-train on the ImageNet performed well on classification, but worse on regression tasks.

Models were evaluated using the mean average precision which was calculated for a single image. Formula used for this calculation is as following

AP = 1thresholds Σ TPtTPt+FPt +FNt

This detection model proposes a solution based on a single model, ensembled over several checkpoints and 4 folds. The model utilises an SSD RetinaNet with SE-ResNext101 encoder pre-trained on ImageNet.

The model is based on RetinaNet implementation on Pytorch with the modifications like calculation/modification, small anchors to boxes, global image classification and dropouts.

This model was trained the RetinaNet model and fixed augmentations with and without the global classification output. The output of global classification was not used directly to classify the images, however making the model predict the other related function improved the result compared to training the regression only output of the model.

To examine experimentally the effect of image augmentations, this model is using augmentation sets like No augmentation, light, Heavy, Heavy without rotation, Heavy with custom rotation.

The results of this detection model can be changed significantly between epochs and depend largely on thresholds. The postprocessing with re-scaling predictions was applied to compensate for the difference between the train and test sets labelling processes.

To conclude this research paper, this detection model proposes a simple and effective algorithm for the localization of lung opacities regions. This model is based on single-shot detector RetinaNet with Se-ResNext101 encoders, pre-trained on ImageNet dataset.

- **CheXNet: Radiologist-Level Pneumonia Detection on Chest X-Rays with Deep Learning : Pranav Rajpurkar Jeremy Irvin Kaylie Zhu Brandon Yang Hershel Mehta Tony Duan Daisy Ding Aarti Bagul Robyn L. Ball Curtis Langlotz Katie Shpanskaya Matthew P. Lungren 3 Andrew Y. Ng**

This research paper about pneumonia detection uses chest X-rays with Deep Learning. This Detection Model contains a data set of 100,000 frontal-view X-ray images with 14 diseases. This Model claim to detect 14 diseases in ChestX-ray14 and achieve

This model 121-layer CNN which on chest X-rays input image gives us output of probability of Pneumonia with a hit map indicating the Area of Pneumonia. The Model is trained using a mini batch size of 16.
The Architecture of this model uses DenseNets to improve flow of information and gradients through the network, making the optimization of very deep networks tractable.

The data set name chest X-ray 14 wang-et-al more than one Lakh frontal view x-ray images where the image size was down sided to 224x224
The training data was augmented with random horizontal flipping.

There was a comparison test for performance done between cheXnet and radiologist where results concluded that cheXnet is Better than radiologist also after performing statistical result cheXnet perform better than radiologist

This model faces limitations of frontal radiographs that were presented to the radiologists and model during diagnosis, but it has been shown that up to 15% of accurate diagnoses require the lateral view, thus the model expects conservative setbacks on performance.

The Algorithm the classified into multiple thoracic pathologies by making some changes

To conclude this research paper, it develops an algorithm which detects pneumonia from chest X-ray images at a level exceeding practicing radiologist. The model shows that a simple extension of our algorithm to detect multiple diseases outperforms previous state of the art on ChestX-ray14, the largest publicly available chest X-ray dataset.

- **Pneumonia Detection Using CNN based Feature Extraction: Dimpy Varshni Kartik Thakral Lucky Agarwal Rahul Nijhawan Ankush Mittal**

Deep Learning Algorithms in analysing medical images were found to be very successful, CNN has also gained much attention for disease classification.
To build this model DenseNet, Deep CNN, SVM, Transfer Learning, Random Forest, Naïve Bayes, KNN, Feature Extraction were used.
Chest X-Rays are considered as the most effective method to determine the extent and location of the septic region in the lungs.

Dataset used in this model was downloaded from Kaggle Platform where the range of the dataset consists of 112,120 Chest X-ray where each image is labelled with one or more out of 14 different thoracic diseases. All the images are of 1024 x 1024 resolution.

The Methodology of the system uses Densely Connected Convolutional Neural Network where its architecture is divided into 3 different stages namely the pre-processing stage, the feature extraction stage and the classification stage.

Here the pre-processing stage is used to reduce the computational complexity of the model whereas feature extraction deals with the description of DenseNet-169 model architecture and its contribution in feature extraction.
Whereas in the classification stage, After Feature Extraction different type of classifiers were used to for the classification task but best results were found from SVM and was also used as a classifier for this model

A Support Vector machine used for binary classification is able to find the best hyperplane for the above training data. Here in this model the performance of SVM highly depends on selection of Kernel and parameters.

To further improve the model hyper-parameter tuning with the SVM classifier was used. And the observation was that it affected the statistical results.
Here the combination of CNN which is based on Feature Extraction and supervised classifier algorithm is resulted in optimal solution for classifying Pneumonia and normal Chest X-Ray images primarily due to the substantive features provided by DenseNet

The best results achieved with DenseNet169 architecture as feature extractors can be explained due to its capability of accessing feature-maps from all of its preceding layers. The proposed model achieved the AUC of 0.8002.

Apart from this model there are still some limitations of this as there is no history of the associated patient in this model. Secondly, only frontal chest X-rays were used

3. SOFTWARE REQUIREMENTS AND SPECIFICATION

INTRODUCTION

Software requirements specification is a detailed description of a system to be developed with its functional and non-functional requirements. SRS may include the use cases of how the user is going to interact with the system. The SRS document consists of all necessary requirements required for system development. To develop the system we should have a clear understanding of Software systems. To achieve this we need to have continuous communication with customers to gather all requirements. This Software Requirements Specification provides a complete description of all the functions and constraints .The document describes the issues related to the system and what actions are to be performed by the development team in order to come up with a better solution.

Project Scope

The scope thus far has been the completion of the basic interfaces that will be used to build the system. The system detects various Pneumonia diseases by handling CT scan image datasets thus saving the time required for the procedure which could be lengthy otherwise. It also reduces the workload of doctors further increasing the accuracy of the detection of the Pneumonia diseases.

User Classes and Characteristics

Class 1: The Admin of the System is Responsible for handling the system and overall process. Giving the exact image input and after generation of report sending it to the right patient through mail Id.

Assumption and Dependencies

1. Assumptions: Admin must have some Scientific Background. It is assumed that the administrator is well known for the recovery of the system. System is totally dependent on admin and his usage. Admin will be responsible for all operations performed in system
2. Dependencies: System is completely depends on the administrator and its usage. Admin will be responsible for all the operations of the system

FUNCTIONAL REQUIREMENTS

The practical necessities for our system describe what the system does.

- The system ought to acknowledge the tissue from the CT scan image.
- System should classify whether the tissue is affected or not.
- The system should provide high accuracy in classification and identification of diseases.

Image Pre-processing:

Description and Priority

- Input- image data
- Priority –High

Stimulus/Response Sequences

Response sequence: image grey scaling

Functional Requirements

- Pre-processed Data

EXTERNAL INTERFACE REQUIREMENTS

User Interfaces

Our system interacts with user on the following occasions:

1. By the user while register, login and for grayscaling , thresholding and for detection of the disease

Hardware Interfaces

The minimum processor required is Pentium IV 2.4 Gz with 40GB Hard Disk and 8GB RAM.

System Requirements:

- Operating System (Any)

Hardware Requirements:

- Pentium IV 2.4 GHz
- 4 GB RAM
- 256 GB HDD

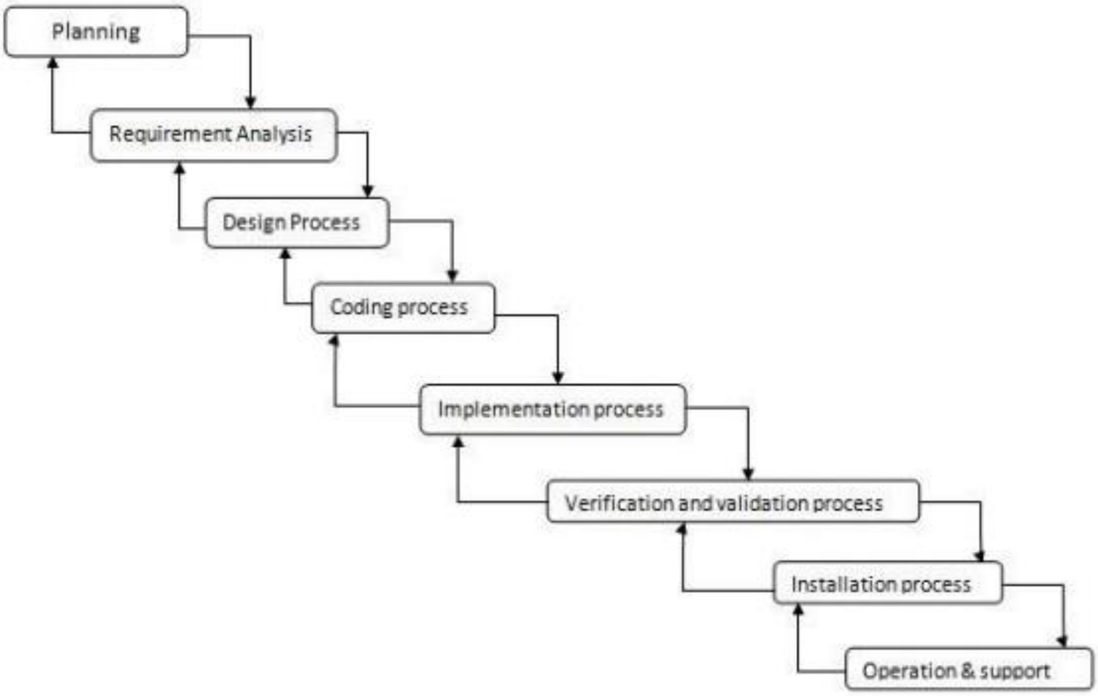
Software Requirements:

- Programming Language: Python 3
- IDE : Python IDE

Database Requirements:

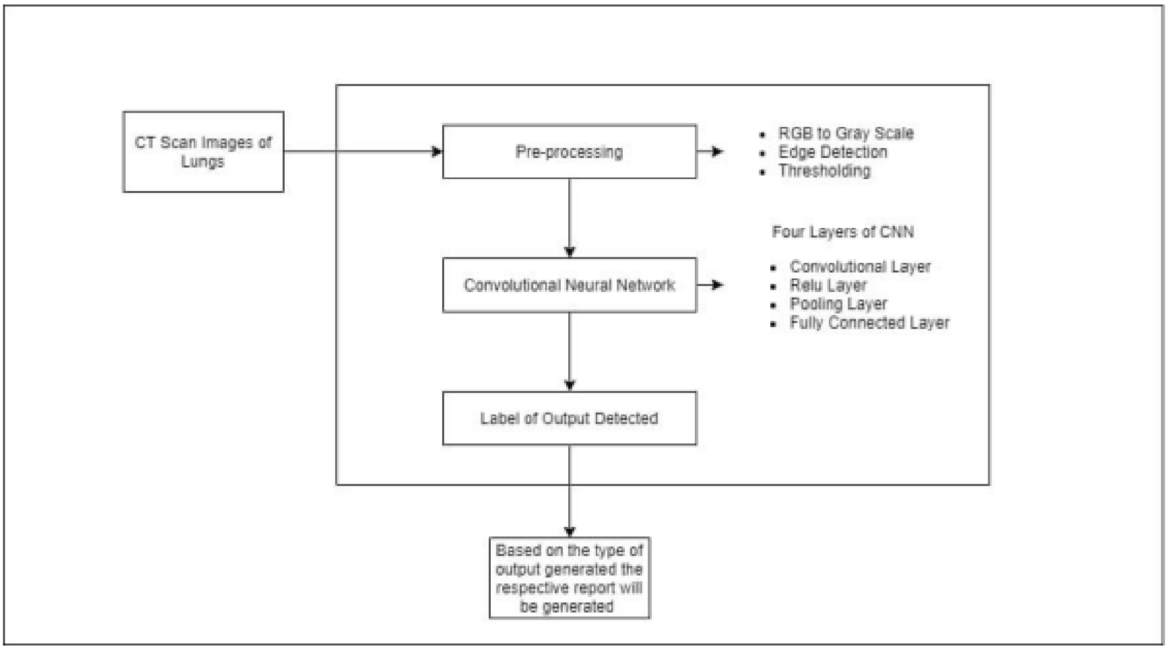
- SQLite

SYSTEM IMPLEMENTATION PLAN:

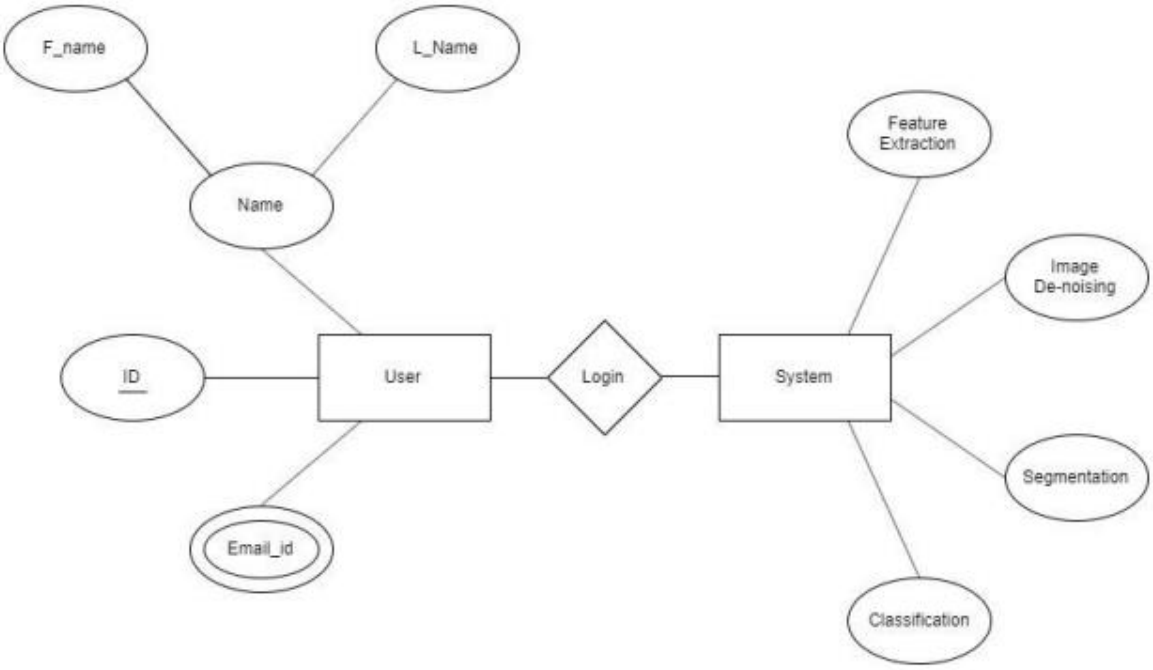


4. SYSTEM DESIGN

SYSTEM ARCHITECTURE:

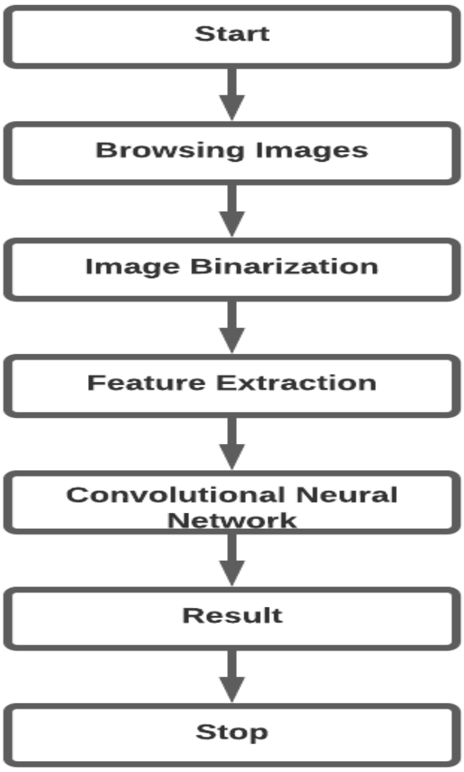


ENTITY RELATIONSHIP DIAGRAMS :



UML DIAGRAMS

Activity Diagram :



5. Other Specifications

ADVANTAGES

- The process of getting a report of CT scan and analysis of the report is automated.
- Workload of pathologists will be reduced by our system.

LIMITATIONS

- Limited to Admin.
- Deployed in an offline environment.

APPLICATION

- The model can be used in Hospitals.
- If we change or take more parameters then the model can also be used to detect other lung diseases.

6. Conclusion and Future Scope

Conclusion:

Pneumonia-screening CNNs achieved better internal than external performance. When models were trained on pooled data from sites with different pneumonia prevalence, they performed better on new pooled data from these sites but not on external data. CNNs robustly identify hospital systems and departments within a hospital, which can have large differences in disease burden and may confound predictions. Pneumonia is a symptom of Hantavirus. Currently there is no cure available to Hantavirus. Oxygen therapy is being used to treat the Hantavirus but it is not a cure. If Hantavirus is detected in early stages then there may be a chance to cure it. We are developing a model that can detect Hantavirus which will reduce the workload of Pathologists and Doctors. The Model will automate the process of Detection of hanta virus and Hantavirus.

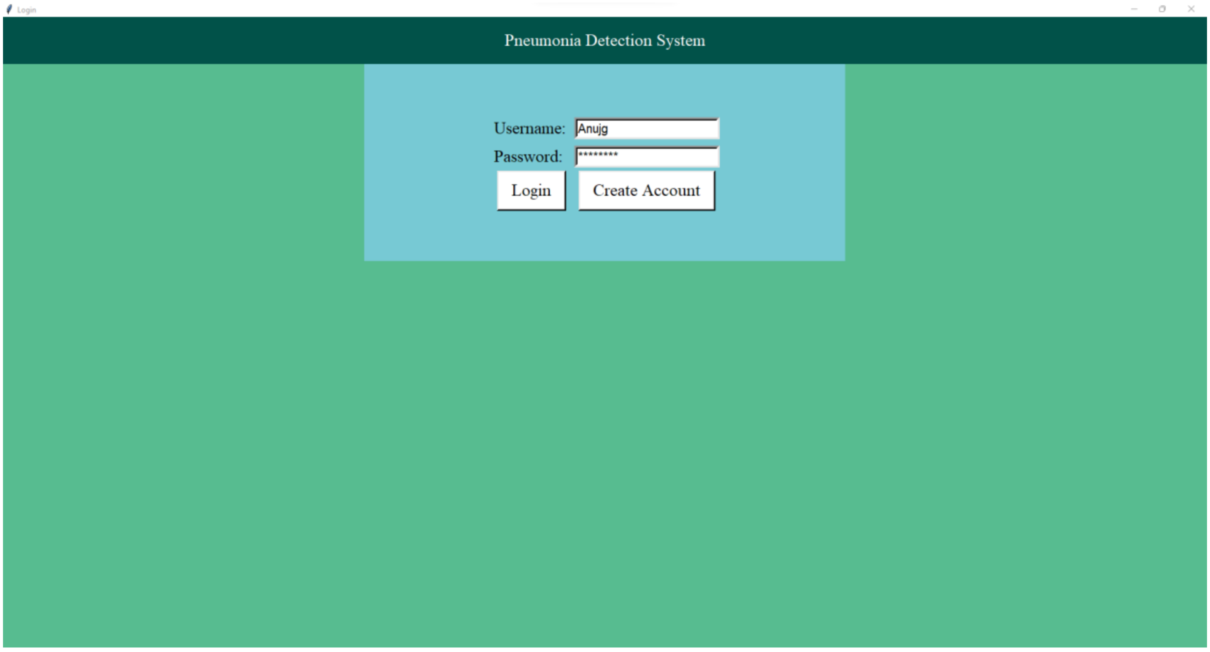
Future Scope:

Can be deployed in real time web model

7 Analysis and Design.

7.1 User Interface Design:

Login



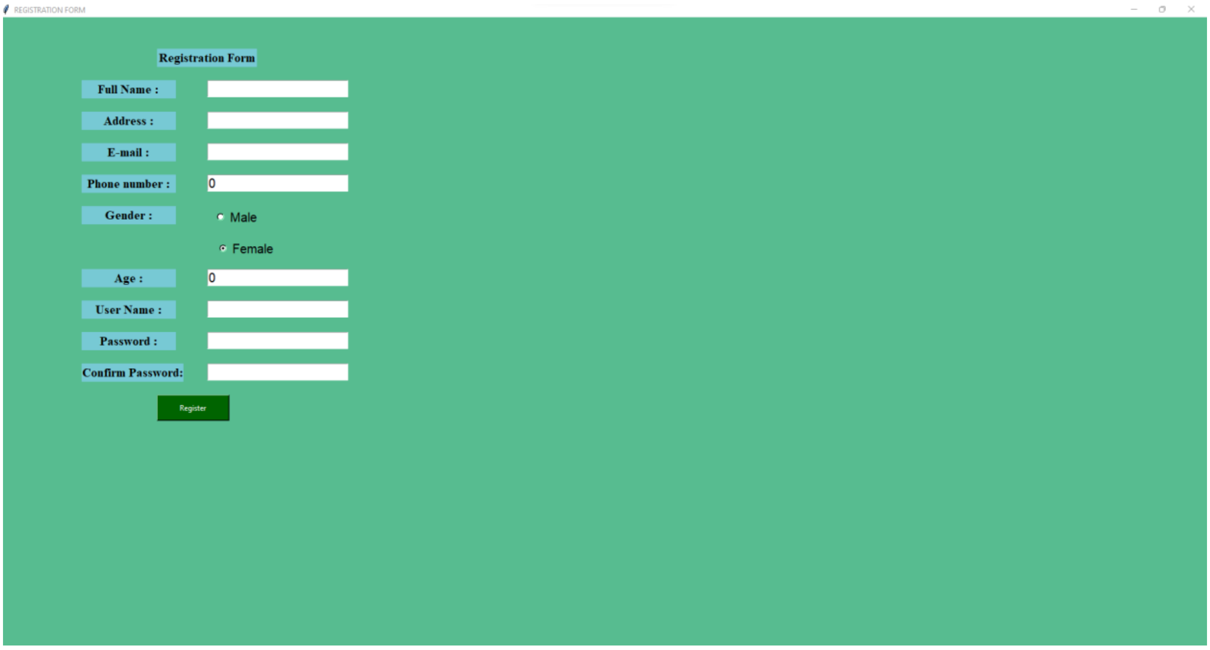
Login

Pneumonia Detection System

Username: Anujg

Password: ********

Registration



REGISTRATION FORM

Registration Form

Full Name :

Address :

E-mail :

Phone number : 0

Gender : ○ Male ● Female

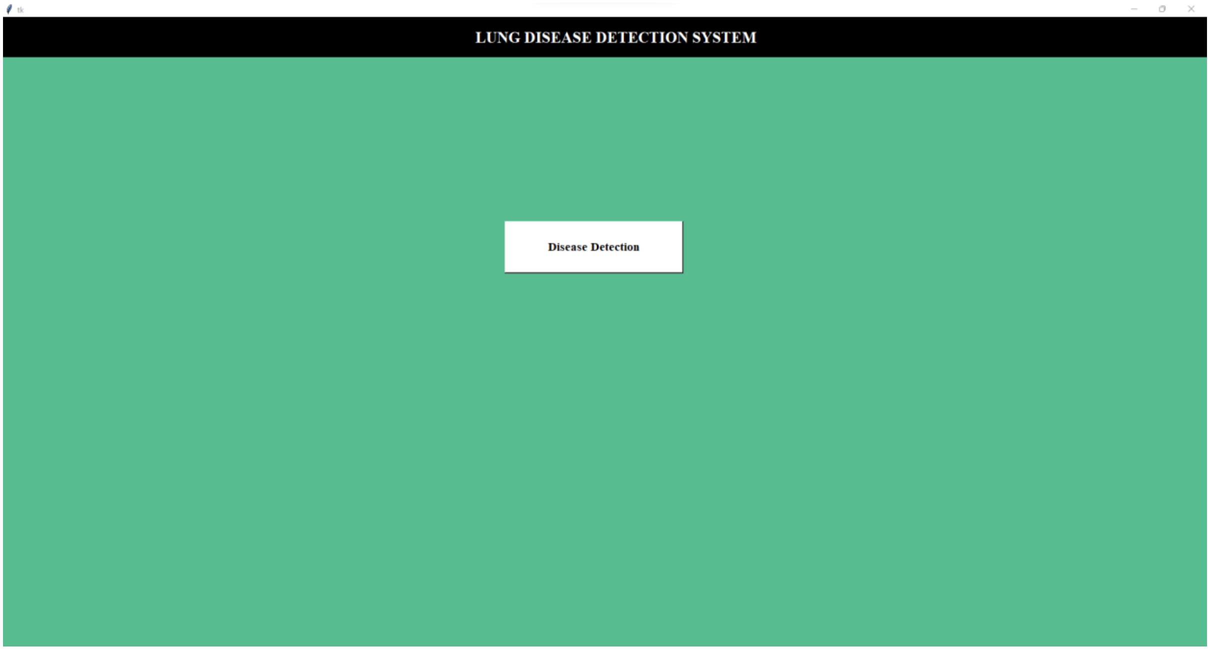
Age : 0

User Name :

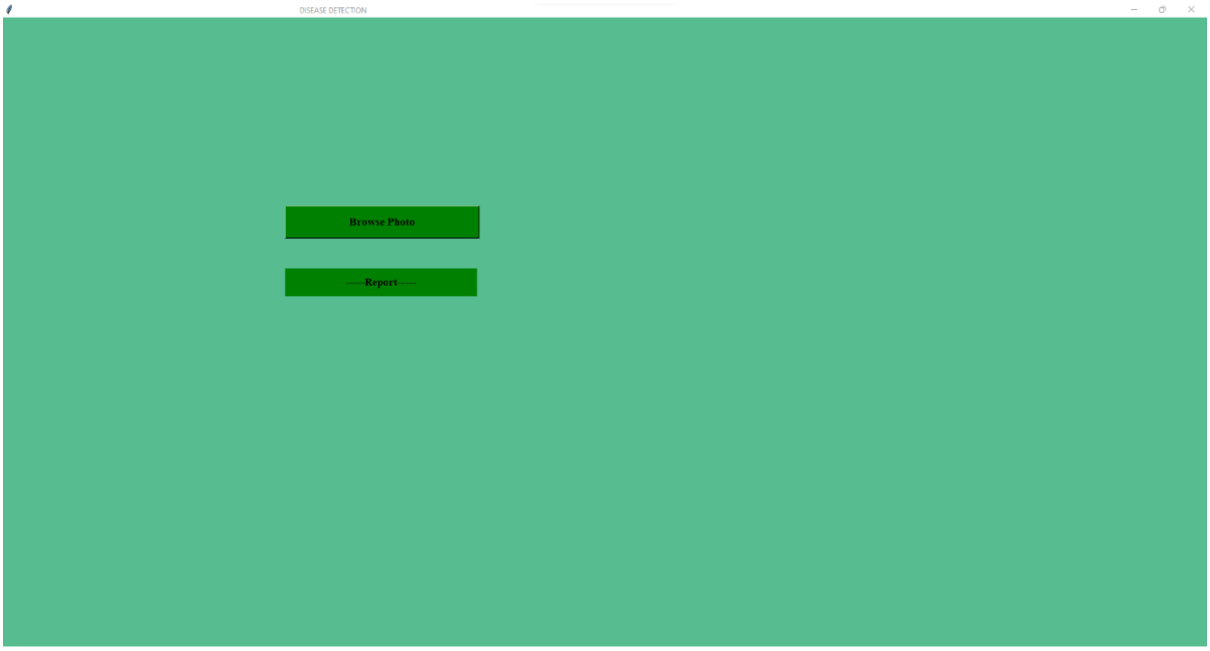
Password :

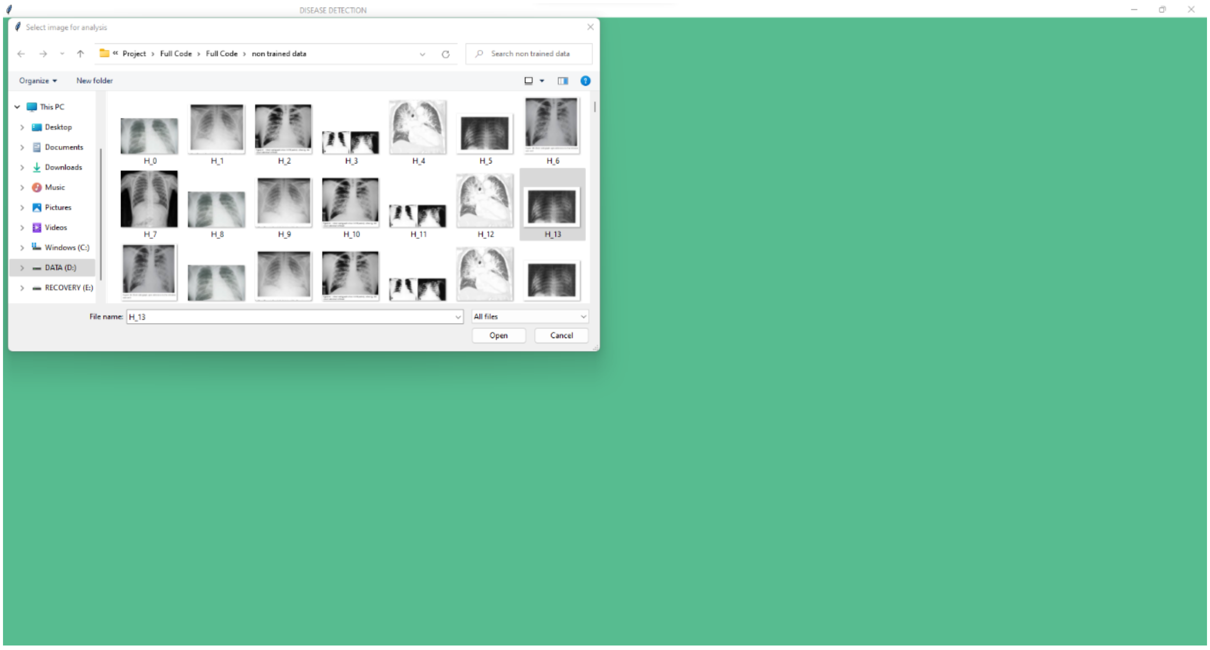
Confirm Password:

Home page



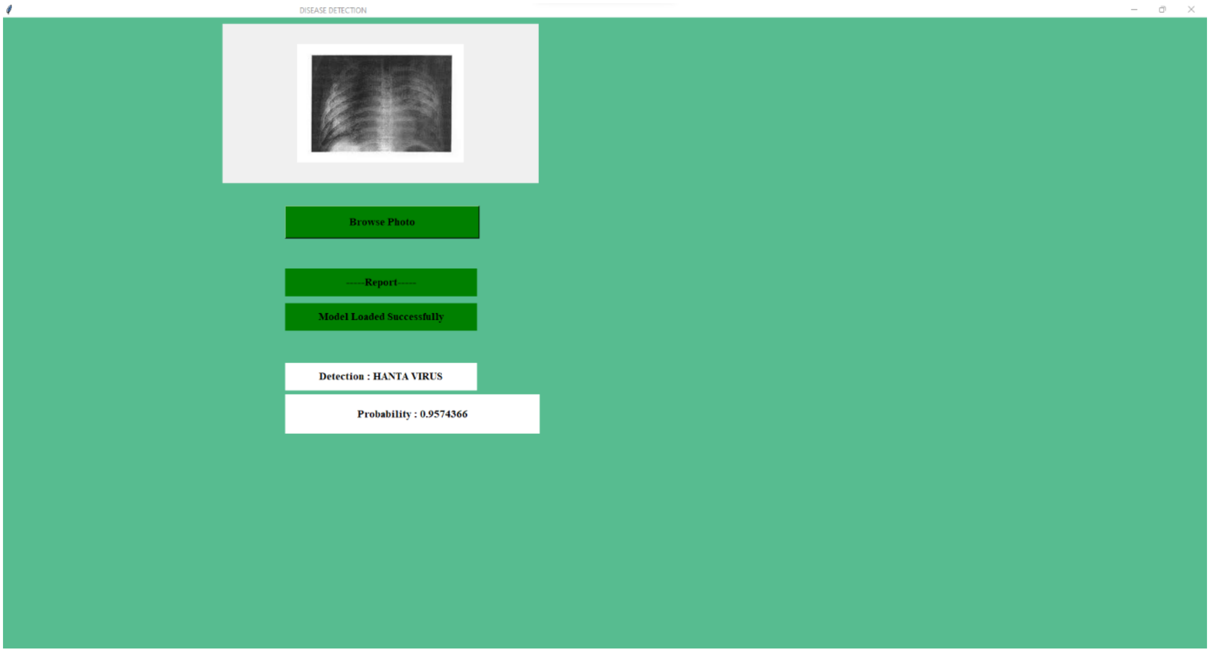
Disease Detection Form





File name: H_13

Output page



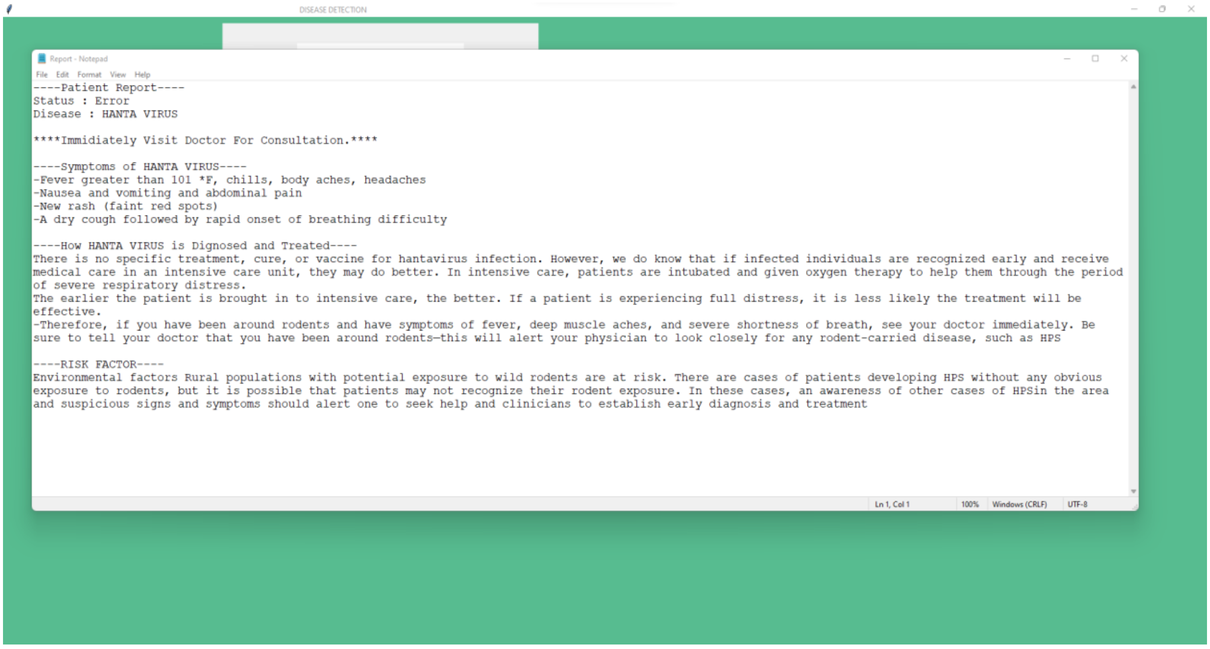
------Report------

Model Loaded Successfully

Detection : HANTA VIRUS

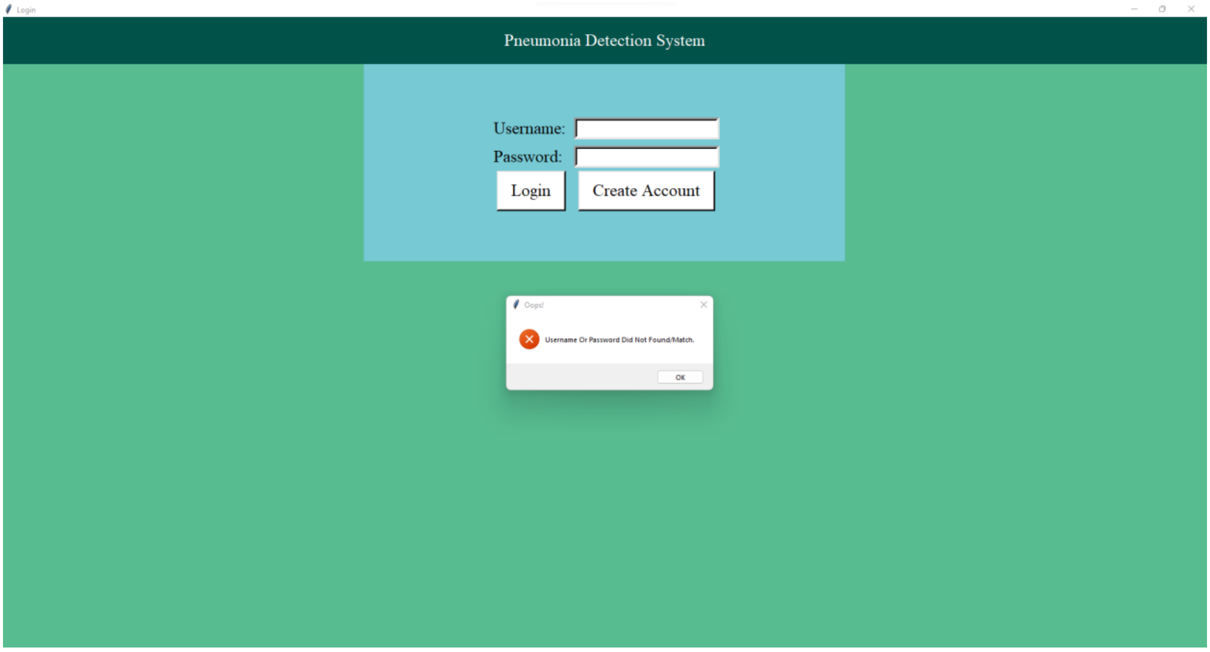
Probability : 0.9574366

Report

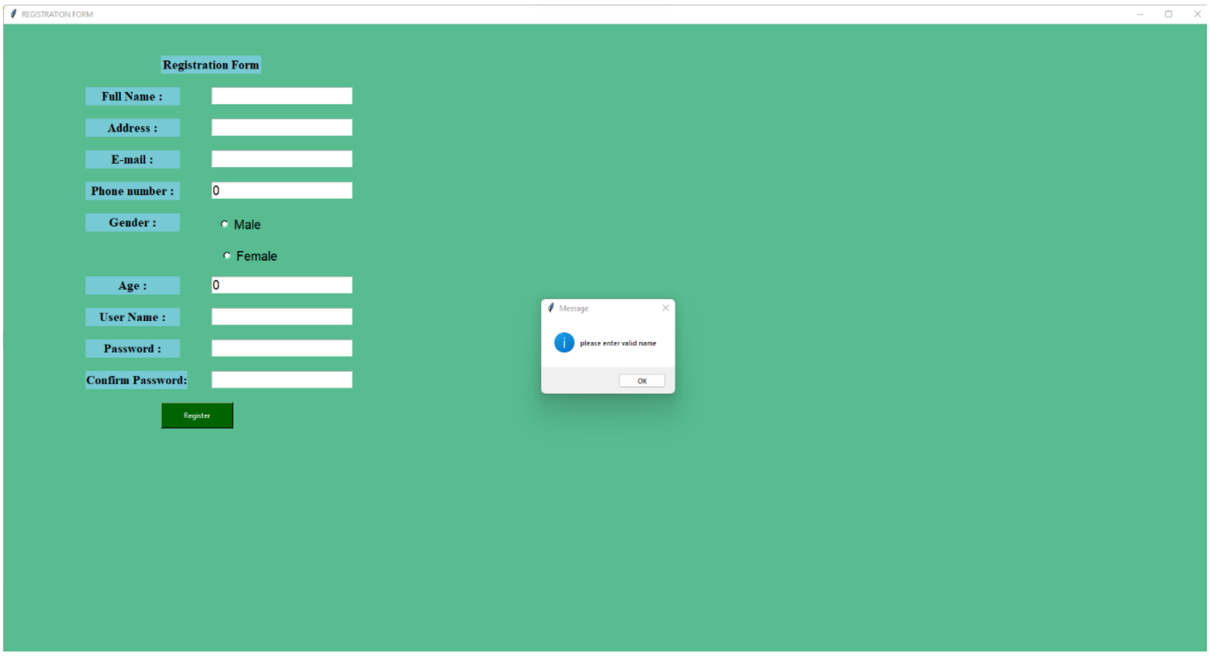


```
----Patient Report----
Status : Error
Disease : HANTA VIRUS

****Immidiately Visit Doctor For Consultation.****

----Symptoms of HANTA VIRUS----
-Fever greater than 101 *F, chills, body aches, headaches
-Nausea and vomiting and abdominal pain
-New rash (faint red spots)
-A dry cough followed by rapid onset of breathing difficulty

----How HANTA VIRUS is Dignosed and Treated----
There is no specific treatment, cure, or vaccine for hantavirus infection. However, we do know that if infected individuals are recognized early and receive
medical care in an intensive care unit, they may do better. In intensive care, patients are intubated and given oxygen therapy to help them through the period
of severe respiratory distress.
The earlier the patient is brought in to intensive care, the better. If a patient is experiencing full distress, it is less likely the treatment will be
effective.
-Therefore, if you have been around rodents and have symptoms of fever, deep muscle aches, and severe shortness of breath, see your doctor immediately. Be
sure to tell your doctor that you have been around rodents—this will alert your physician to look closely for any rodent-carried disease, such as HPS

----RISK FACTOR----
Environmental factors Rural populations with potential exposure to wild rodents are at risk. There are cases of patients developing HPS without any obvious
exposure to rodents, but it is possible that patients may not recognize their rodent exposure. In these cases, an awareness of other cases of HPSin the area
and suspicious signs and symptoms should alert one to seek help and clinicians to establish early diagnosis and treatment
```

7.2 Validations Applied:



Username Or Password Did Not Found/Match.



Full Name :
Address :
E-mail :
Phone number : 0
Gender : ○ Male ○ Female
Age : 0
User Name :
Password :
Confirm Password:

please enter valid name

7.3 Table Specifications:

Database Name: Pneumonia Detection System
Table Name: Doctor_Data

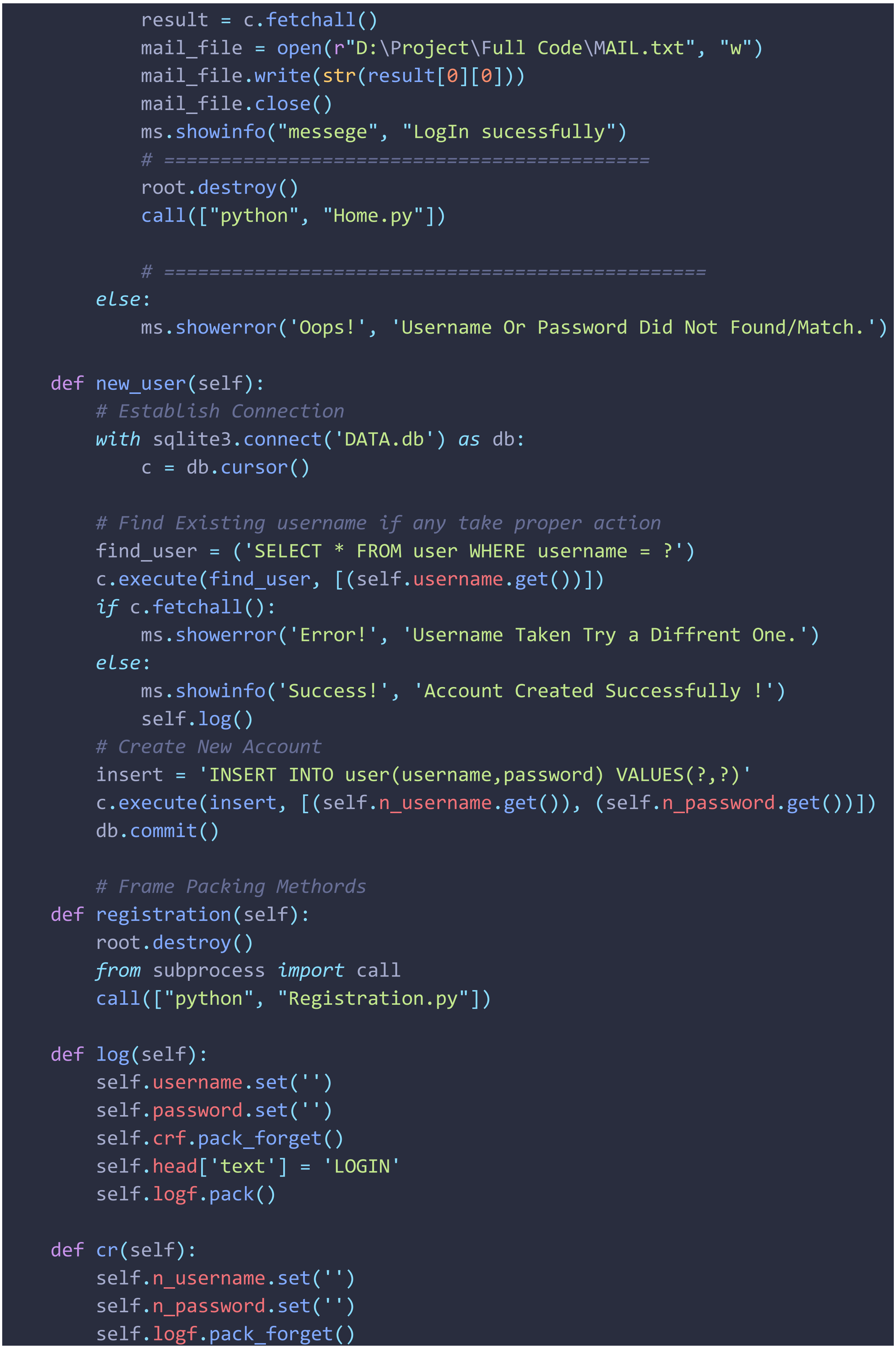
Data Dictionary:

Sr. No.	Field Name	Data Type	Size	Description
1	Id	Int	11	Unique Id is given for every user
2	Full_Name	Varchar	100	Full Name of user at time of registration
3	Address	Varchar	100	Address is taken at the time of registration
4	Email_Id	Varchar	100	Email of user at the time of registration
5	Phone_No	Number	100	Phone No of user at the time of registration
6	Gender	Varchar	10	Gender of user at the time of registration
7	Age	Numver	03	Age of user at the time of registration
8	UserName	Varchar	100	Username of user at the time of registration and login
9	Password	Varchar	100	Password of user at the time of registration and login

10	**Cnf_Pass**	**Varchar**	**100**	**Confirm Password of user at the time of registration and login**

8 Code

```
# Imports
from sqlite3.dbapi2 import Row
from tkinter import messagebox as ms
import sqlite3
import tkinter as tk
from subprocess import call
from tkinter.constants import CENTER
# main Class


class main:
    def __init__(self, master):

        self.master = master
        # Some Usefull variables
        self.username = tk.StringVar()
        self.password = tk.StringVar()
        self.n_username = tk. StringVar()
        self.n_password = tk.StringVar()
        # Create Widgets
        self.widgets()

    # Login Function
    def login(self):
        # Establish Connection

        with sqlite3.connect('DATA.db') as db:
            c = db.cursor()

        # Find user If there is any take proper action
        db = sqlite3.connect('DATA.db')
        cursor = db.cursor()
        cursor.execute("CREATE TABLE IF NOT EXISTS DOCTOR_DATA"
                       "(Fullname TEXT, address TEXT, username TEXT, Email
TEXT, Phoneno TEXT,Gender TEXT,age TEXT , password TEXT)")
        db.commit()
        find_entry = (
            'SELECT * FROM DOCTOR_DATA WHERE username = ? and password = ?')
        c.execute(find_entry, [(self.username.get()), (self.password.get())])
        result = c.fetchall()

        if result:
            msg = ""
            self.logf.pack_forget()
            print(msg)
            find_entry = ('SELECT Email FROM DOCTOR_DATA WHERE username = ?')
            c.execute(find_entry, [(self.username.get())])
```



```
            result = c.fetchall()
            mail_file = open(r"D:\Project\Full Code\MAIL.txt", "w")
            mail_file.write(str(result[0][0]))
            mail_file.close()
            ms.showinfo("messege", "LogIn sucessfully")
            # ===========================================
            root.destroy()
            call(["python", "Home.py"])

            # ================================================
        else:
            ms.showerror('Oops!', 'Username Or Password Did Not Found/Match.')

    def new_user(self):
        # Establish Connection
        with sqlite3.connect('DATA.db') as db:
            c = db.cursor()

        # Find Existing username if any take proper action
        find_user = ('SELECT * FROM user WHERE username = ?')
        c.execute(find_user, [(self.username.get())])
        if c.fetchall():
            ms.showerror('Error!', 'Username Taken Try a Diffrent One.')
        else:
            ms.showinfo('Success!', 'Account Created Successfully !')
            self.log()
        # Create New Account
        insert = 'INSERT INTO user(username,password) VALUES(?,?)'
        c.execute(insert, [(self.n_username.get()), (self.n_password.get())])
        db.commit()

        # Frame Packing Methords
    def registration(self):
        root.destroy()
        from subprocess import call
        call(["python", "Registration.py"])

    def log(self):
        self.username.set('')
        self.password.set('')
        self.crf.pack_forget()
        self.head['text'] = 'LOGIN'
        self.logf.pack()

    def cr(self):
        self.n_username.set('')
        self.n_password.set('')
        self.logf.pack_forget()
        self.head['text'] = 'Create Account'
        self.crf.pack()

    def widgets(self):
        self.head = tk.Label(self.master, text='Pneumonia Detection System',
                             background="#015249", foreground="white",
font=('Times New Roman', 20), padx=800, pady=20)
        self.head.pack()

        self.logf = tk.Frame(self.master, padx=200, pady=80,
                             background="#77C9D4", )

        tk.Label(self.logf, text='Username: ', background="#77C9D4", font=(
            "Times New Roman", 20), pady=5, padx=5).grid(sticky=tk.W)
        tk.Entry(self.logf, textvariable=self.username, bd=5,
                 background="white", font=('', 15)).grid(row=0, column=1)
        tk.Label(self.logf, text='Password: ', background="#77C9D4", font=(
            "Times New Roman", 20), pady=5, padx=5).grid(sticky=tk.W)
        tk.Entry(self.logf, textvariable=self.password, bd=5,
                 background="white", font=('', 15), show='*').grid(row=1,
column=1)
        tk.Button(self.logf, text=' Login ', bd=3, font=("Times New Roman",
20),
                  background="White", foreground="Black", padx=5, pady=5,
command=self.login).grid()
        tk.Button(self.logf, text=' Create Account ', font=("Times New Roman",
20),
                  background="White", foreground="Black", bd=3, padx=5,
pady=5, command=self.registration).grid(row=2, column=1)

        self.logf.pack()
        self.crf = tk.Frame(self.master, padx=200, pady=200)


root = tk.Tk()

root.configure(background="#57BC90")
root.geometry("1920x1080")
root.title("Login")

main(root)

root.mainloop()
```

9: APPENDIX

Problem Statement

Detection of Pneumonia and Hanta virus from Chest CT Scanned Images using CNN (Convolutional Neural Network).

Feasibility Assessment

Feasibility study is the test of a system proposal according to its workability, impact on the organization, ability to meet user needs, and effective use of resources. It focuses on the evaluation of existing system and procedures analysis of alternative candidate system cost estimates. Feasibility analysis was done to determine whether the system would be feasible. The development of a computer-based system or a product is more likely plagued by resources and delivery dates. Feasibility study helps the analyst to decide whether or not to proceed, amend, postpone or cancel the project, particularly important when the project is large, complex and costly.

Once the analysis of the user requirements complements, the system has to check for the compatibility and feasibility of the software package that is aimed at. An important outcome of the preliminary investigations is the determination that the system requested is feasible.

Economic Feasibility

The proposed model is developed using open source IDE and Database Server like PyCharm and SQLite.

Time Feasibility

Time feasibility study will take into account the period in which the project is going to take up to its completion. A project will fail if it takes too long to be completed before it is useful.