MULTIPLE DISEASE PREDICATION
USING ML AND IP

A Project Report Submitted by

YASH NAWANI - 91800103168
PRIYANKKUMAR GONDHA - 91800103145

in partial fulfilment for the award of the degree of

Bachelor of Technology
in
Computer Engineering

Faculty of Technology
Marwadi University, Rajkot
2021-22

Faculty of Technology

Marwadi University

Computer Engineering Department

2021-22

CERTIFICATE

This is to certify that the project entitled **Multiple Disease Predication Using ML and IP** has been carried out by **Yash Nawani - 91800103168** under my guidance in partial fulfilment of the degree of Bachelor of Technology in Computer Engineering of Marwadi University, Rajkot during the academic year 2021-22.

Date: _____

Internal Guide **Head of the Department**

Prof. Mahender Reddy Chilukala
Assistant Professor

Faculty of Technology

Marwadi University

Computer Engineering Department

2021-22

CERTIFICATE

This is to certify that the project entitled **Multiple Disease Predication Using ML and IP** has been carried out by **Priyankkumar Gondha - 91800103145** under my guidance in partial fulfilment of the degree of Bachelor of Technology in Computer Engineering of Marwadi University, Rajkot during the academic year 2021-22.

Date: _____

Internal Guide **Head of the Department**

Prof. Mahender Reddy Chilukala
Assistant Professor

Acknowledgments

I would like to express my special thanks of gratitude to my teachers as well as our dean who gave me the golden opportunity to do this wonderful project on this topic, which also helped me in doing a lot of research and we came to know about so many new things. We are really thankful to them.

Secondly, we would also like to thank my parents and friends who helped me a lot in finalizing this project within the limited time frame. We are making this project not only for marks but to also increase my knowledge.

Thanks again to all those who helped us.

INDEX

Institute's Vision and Mission

Institute's Vision

Our vision is to address challenges facing our society and planet through sterile education that builds capacity of our students and empower them through their innovative thinking practice and character building that will ultimately manifest to boost creativity and responsibility utilizing the limited natural resources to meet the challenges of the 21st century.

Institute's Mission

- To Produce creative, responsible and informed professionals
- To produce individuals who are digital-age literates, inventive thinkers, effective communicators and highly productive.
- To deliver cost-effective quality education
- To offer world-class, cross-disciplinary education in strategic sectors of economy though well devised and synchronized delivery structure and system, designed to tackle the creative intelligence and enhance the productivity of individuals.
- To provide a conducive environment that enables and promotes individuals to creatively interact, coordinate, disseminate and examine change, opinion as well as concept that will enable students to experience higher level of learning acquired through ceaseless effort that lead to the development of character, confidence, values and technical skills.

Department's Vision and Mission

Department's Vision

To impart quality technical education through research, innovation and teamwork for creating professionally superior and ethically strong manpower that meet the global challenges of engineering industries and research organization in the area of Computer Engineering.

Department's Mission

- Maintain a vital, state-of-the art ICT enabled teaching and learning methodologies, which provides its students and faculty with opportunities to create, interpret, apply and disseminate knowledge.
- Enable graduates in becoming digital age literates, innovators, efficient communicators and result oriented professionals.
- Dedicate itself to providing its students with the skills, knowledge and attitudes that will allow its graduates to succeed as engineers, leaders, professionals and entrepreneurs.
- Prepare its graduates for life-long learning to meet intellectual, ethical and career challenges.
- Inspire graduates for competitive exam higher education as well as research and development.

PEO, PO and PSO

Program Educational Objectives (PEO):

Our graduated students are expected to fulfill the following Program Educational Objectives (PEOs):

1. **Core Competency**: Successfully apply fundamental mathematical, scientific, and engineering principles in formulating and solving engineering and real life problems for betterment of society.

2. **Breadth**: Will apply current industry accepted practices, new and emerging technologies to analyse, design, implement and maintain state of art solutions.

3. **Professionalism**: Work effectively and ethically in ever changing global professional environment and multi-disciplinary environment.

4. **Learning Environment**: Demonstrate excellent communication and soft skills to fulfil their commitment towards social responsibilities and foster life-long learning.

5. **Preparation**: Promote research and patenting to enhance technical and entrepreneurship skills within them.

- Function and communicate effectively to solve technical problems.
- Advance professionally to roles of greater computer engineering responsibilities, and/or by transitioning into leadership position in various industries such as business, government, and/or education.
- Prepare for entrepreneurship skills by demonstrating commitment to community by applying technical skills and knowledge to support various service activities.
- Place themselves in positions of leadership and responsibility within an organization and progress through advanced degree or certificate programs in engineering, business, and other professionally related fields.
- Participate in higher study by the process of life-long learning through the successful completion of advanced degrees, continuing education, and/or engineering certification(s)/licensure or other professional development.

Program Outcomes (POs)

Engineering Graduates will be able to:

PO1: Engineering knowledge: Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.

PO2: Problem analysis: Identify, formulate, review research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.

PO3: Design/development of solutions: Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.

PO4: Conduct investigations of complex problems: Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.

PO5: Modern tool usage: Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations.

PO6: The engineer and society: Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice.

PO7: Environment and sustainability: Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.

PO8: Ethics: Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.

PO9: Individual and team work: Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.

PO10: Communication: Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.

PO11: Project management and finance: Demonstrate knowledge and understanding of the engineering and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.

PO12: Life-long learning: Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.

Program Specific Outcomes (PSOs)

PSO1. Students shall demonstrate skills, the knowledge and competence in the analysis, design and development of computer based systems addressing industrial and social issues.

PSO2. Students shall have competence to take challenges associated with future technological issues associated with security, wearable devices, augmented reality, Internet of Anything etc.

Abstract

The wide adaptation of computer-based technology in the health care industry resulted in the accumulation of electronic data. Due to the substantial amounts of data, medical doctors are facing challenges to analyze symptoms accurately and identify diseases at an early stage. However, supervised machine learning (ML) algorithms have showcased significant potential in surpassing standard systems for disease diagnosis and aiding medical experts in the early detection of high-risk diseases. In this literature, the aim is to recognize trends across various types of supervised ML models in disease detection through the examination of performance metrics. The most prominently discussed supervised ML algorithms were Naïve Bayes (NB), Decision Trees (DT), K-Nearest Neighbor (KNN). As per findings. The Logistic Regression (LR) performed highly at the prediction of heart diseases. Finally, Random Forest (RF), and Convolutional Neural Networks (CNN) predicted in precision of brain tumor diseases and common diseases, respectively.

*We have referred various books regarding different components working. We have taken ideas of the project from different website and I have also seen YouTube videos and used Google search engine for completing my project***.**

List of Figures

List of Tables

1 Introduction

1.1 Problem Summary and Introduction

Machine Learning is a subset of AI that is mainly deal with the study of algorithms which improve with the use of data and experience. Machine Learning has two phases i.e., Training and Testing. Machine Learning provides an efficient platform in medical field to solve various healthcare issues at a much faster rate. There are two kinds of Machine Learning – Supervised Learning and Unsupervised Learning. In supervised learning we frame a model with the help of data that is well labelled. On the other hand, unsupervised learning model learn from unlabeled data. The Earth is going through a purplish patch of technology where the demand of intelligence and accuracy is increasing behind it. Today's people are likely addicted to internet but they are not concerned about their physical health. People ignore the small problem and don't visit to visit hospital which turn into serious disease with time.

'The modern technology can accurately predict a patient's disease and with the advancement in educational facilities, doctors, today are not any less capable. But as an aftermath, the patient has to spend lots of money for basic checkups and disease analysis. Along with that predicament of disease still take long hours to produce a final result.'

1.2 Aim and Objective

1.2.1 Aim

Taking the advantage of this growing technology, our basis aim is to develop such a system that will predict the multiple diseases in accordance with symptoms put down by the patients without visiting the hospitals / physicians.

1.2.2 Objective

The project aims to deliver patients accurate predication of disease, symptoms and its cure. It will be done through medium of web implementation where users can use image and reports to generate results. On top of that the website will have availability to predict multiple diseases in accordance to the data fed.

1.3 Problem Specification

The main problem that this project aims to solve is to identify the disease of the patient with the need physically go to any hospital/physician and ultimately spend less money on such checkups. This problem is a major issue in many developing countries like India, where people are not financially able to afford such regular check ups. So, a free website that will help them any particular disease they might have caught will surely help people in many ways. For instance, people will be spending less money of physicians meeting fee hence more saving. In addition to this, the website will have more trust and reliability as many doctors might cheat their patients by giving false information in order to get their money. On top of that, the website is available to users 24/7, which means they don't have to worry about any particular timing on which doctor will be available. And lastly, the website can easily handle multiple users which means no more standing in long queues at doctor's clinic.

1.4 Literature Review and Prior Art Search

The major problem with using AI for the diagnosis of disease is the lack of data for training predictive models. Though there is vast amount of data including mammograms, genetic tests, and medical records, they are not open to the people who can make use of them for research. Some initiatives like "100,000 Genomes Project" in the UK, the U.S. Department of Veteran Affairs' "Million Veteran Program", and the NIH's "The Cancer Genome Atlas" will hopefully provide data to researchers and data scientists. Disease Prediction Using Machine Learning Dept. of CSE, CMRIT 2019-2020 Page 3 In many countries' health records are being digitized. The adoption of EMR is also increasing. According to a data brief by The Office of National Coordinator for Health Information Technology (ONC), 3 out of 4 private or not-for-profit hospitals adopted at least a Basic EHR system in the US. In many other countries, different EHR systems exist. The Stockholm EPR corpus is a great

example of such systems which consists of data from 512 clinical units with over 2 million patient records. India is thinking about setting up a National eHealth Authority (NeHA) during the Digital India program. These types of electronic health documents provide a huge amount of data for intelligence data analysis. Many research has been conducted on predicting various diseases like Liver Disease, Heart Disease, Diabetes etc., detecting tumours, leukaemia etc. using computer vision, assisting doctors in making efficient decisions.

1.5 Plan of Work

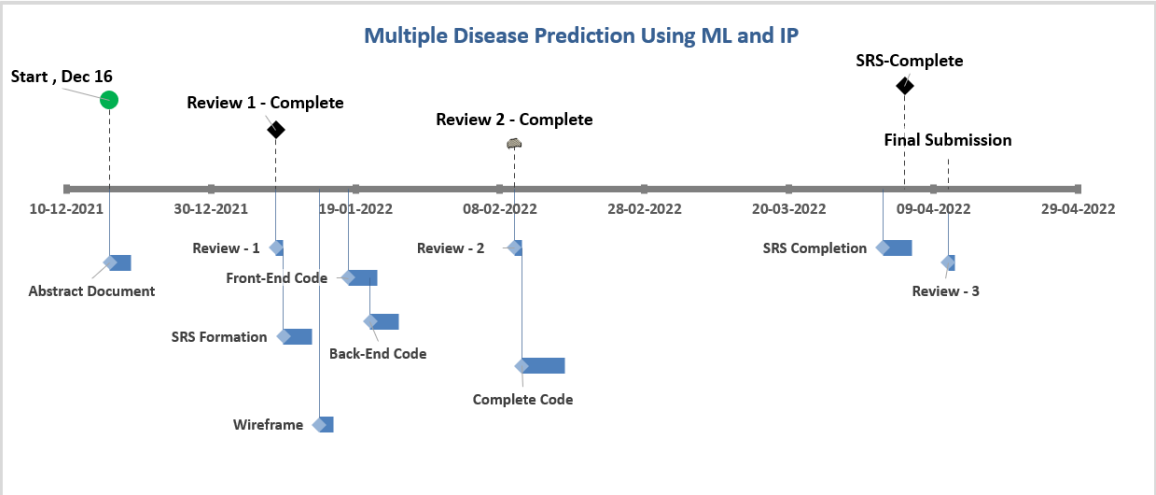

Figure 1-1: Project Time Frame

The above figure depicts the basic timeline of our project. Starting from 16th of December, we have marked every task's date and days it took to complete. We have final review on 11th of April 2022 and hence it marks to final submission of the project. The basic tasks that we have marked as our milestone are in the table below:

Tasks

Start	End	Duration	Label	Vert. Position	Vert. Line
16-12-2021	18-12-2021	3	Abstract Document	-25	-25
08-01-2022	08-01-2022	1	Review - 1	-20	-20
09-01-2022	12-01-2022	4	SRS Formation	-50	-30
14-01-2022	15-01-2022	2	Wireframe	-80	-80
18-01-2022	21-01-2022	4	Front-End Code	-30	-30
21-01-2022	24-01-2022	4	Back-End Code	-45	-15
10-02-2022	10-02-2022	1	Review - 2	-20	-20
11-02-2022	16-02-2022	6	Complete Code	-60	-40
02-04-2022	05-04-2022	4	SRS Completion	-20	-20
11-04-2022	11-04-2022	1	Review - 3	-25	-20
			Insert new rows above this one		

Milestones

Date	Label	Position
16-12-2021	Start, Dec 16	30
05-04-2022	SRS-Complete	35
08-01-2022	Review 1 - Complete	20
10-02-2022	Review 2 - Complete	15
11-04-2022	Final Submission	10
	Insert new rows above this one	

Table 1: Tasks and Milestones

1.6 Materials / Tools Required

1.6.1 Software requirements

- Google Collab
- Visual studio
- Python
- CSS, HTML

1.6.2 Hardware requirements

- 64- bit or 32-bit Processor
- 2 GB RAM 32-bit or 64-bit
- 16 GB Hard disk space
- Internal or Discrete Graphics

2 Analysis, Design Methodology and Implementation Strategy

2.1 Observation Matrix

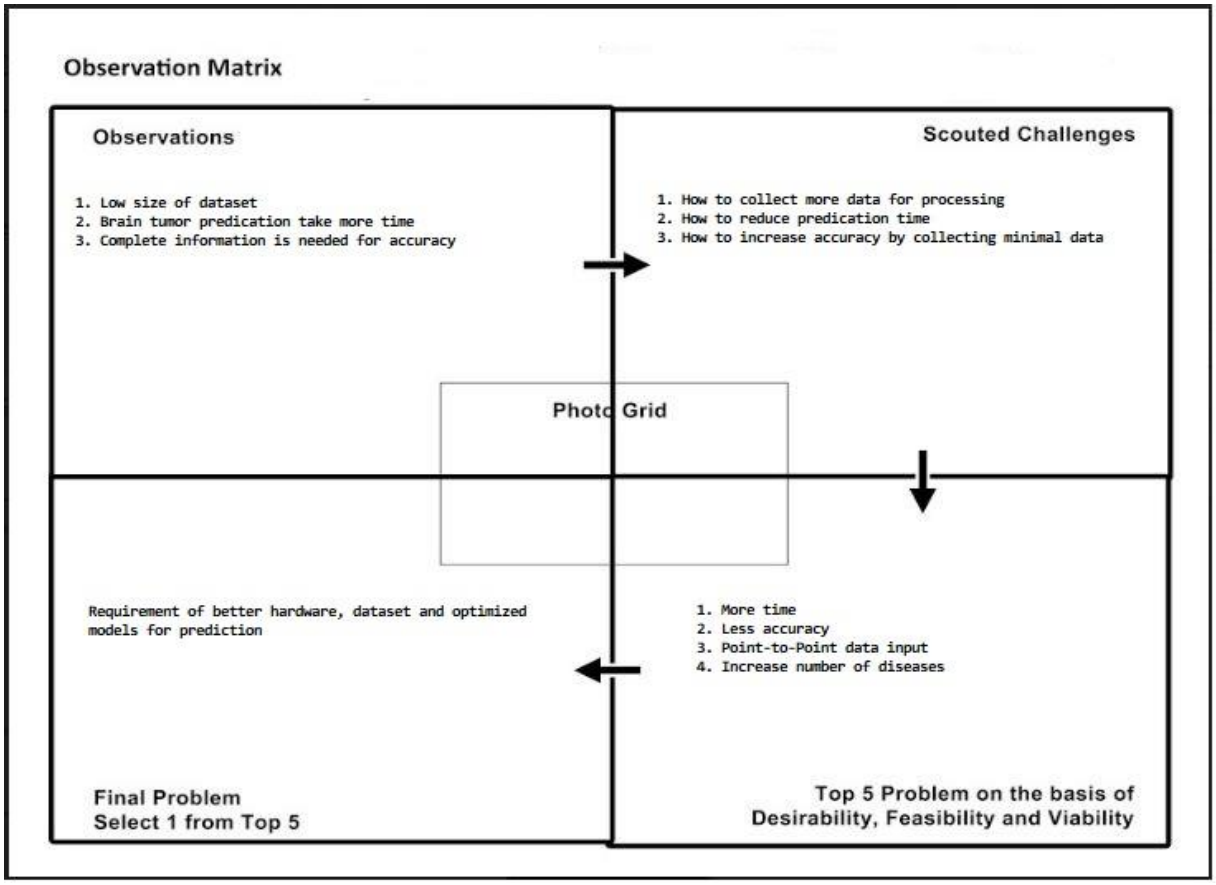

Figure 2-1 Observation Canvas

On observation of the project, it can be noticed that we have used datasets having less amount of data which took toll on accuracy. In addition to this, brain tumor prediction uses image augmentation which ultimately increases the time taken to predict the disease. Also, all the parameters need to be put to get accurate results. Hence all this leads to our top 5 problems that we face that are: Time consumption, Accuracy, point-to-point data fill, and a smaller number of available diseases. Fortunately, better hardware, large dataset and optimized model can solve all these problems.

2.2 Ideation Canvas

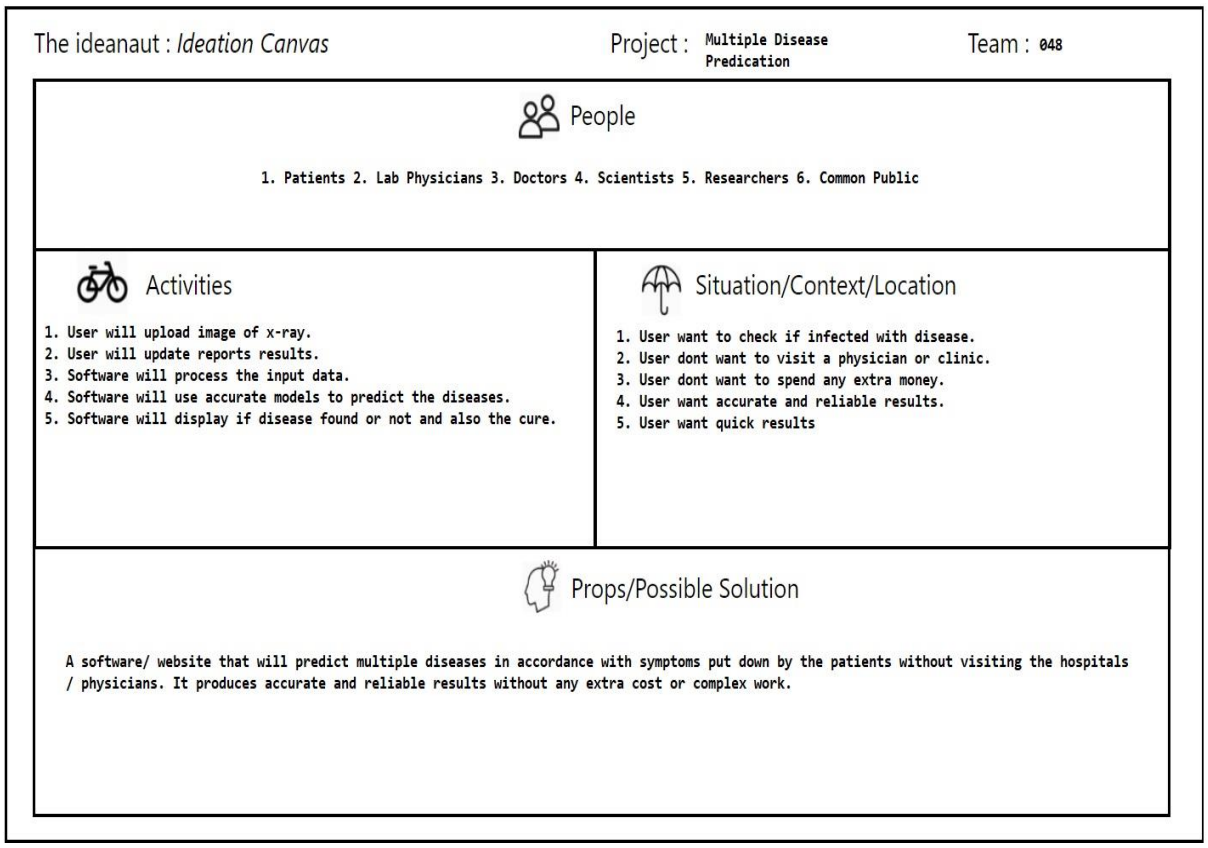

Figure 2-2 Ideation Canvas

We have developed a software/ website that will predict the multiple diseases in accordance with symptoms put down by the patients without visiting the hospitals / physicians. It produces accurate and reliable result without any extra cost or work. The people who interact with the software which are: Patients, Lab Physicians, Doctors, Scientists, Researchers, Common Public. Users and software perform various activities such as User will upload image of x-ray, User will update reports results, Software will process the input data, Software will use accurate models to predict the diseases. This software can be used under many situations but is not limited to User want to check if infected with disease, User don't want to visit a physician or clinic, User don't want to spend any extra money, User want accurate and reliable results, User want quick results. Product Development Canvas

Our main purpose is to develop a software/ website that will predict the multiple diseases in accordance with symptoms put down by the patients without visiting the hospitals /

physicians. It produces accurate and reliable result without any extra cost or work. The people who interact with the software which are: Patients, Lab Physicians, Doctors, Scientists, Researchers, Common Public. Taking in note the user experiences the software sis fully functioning after development. The results generated are accurate and correct. Users can totally rely on this software as per need as it can prove to be very helpful. The project has many components like a browser, clinical data, machine learning models, a good user interface and accurate results. Using product functionalities like upload X-Ray image, input test results data, prediction of disease using models, display results and cure users can take benefit of many features like Less time to predict, very easy to use, Accurate disease prediction, 24/7 Available, Cure for disease provided.

2.3 Database Design

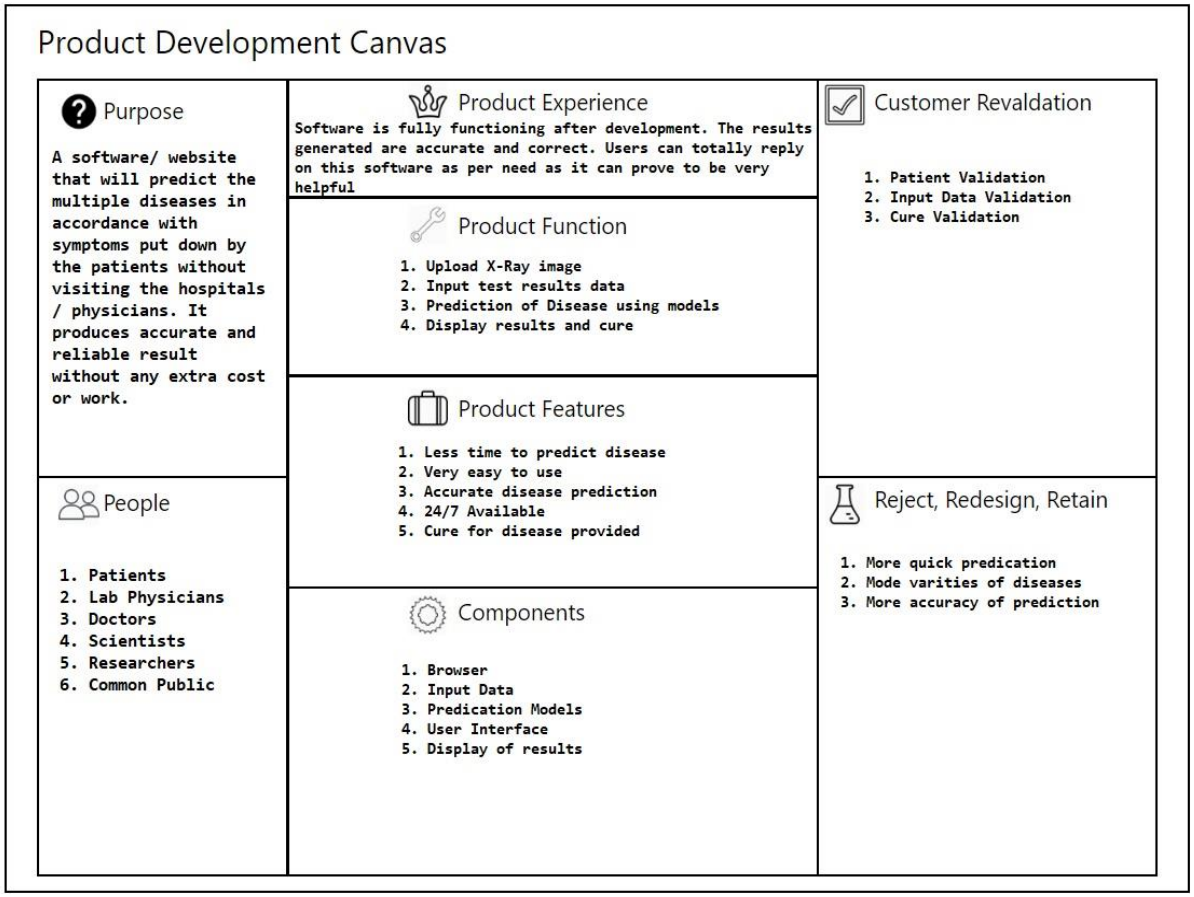

Figure 2-3 Product Development Canvas

2.3.1 Entity Relation Diagram

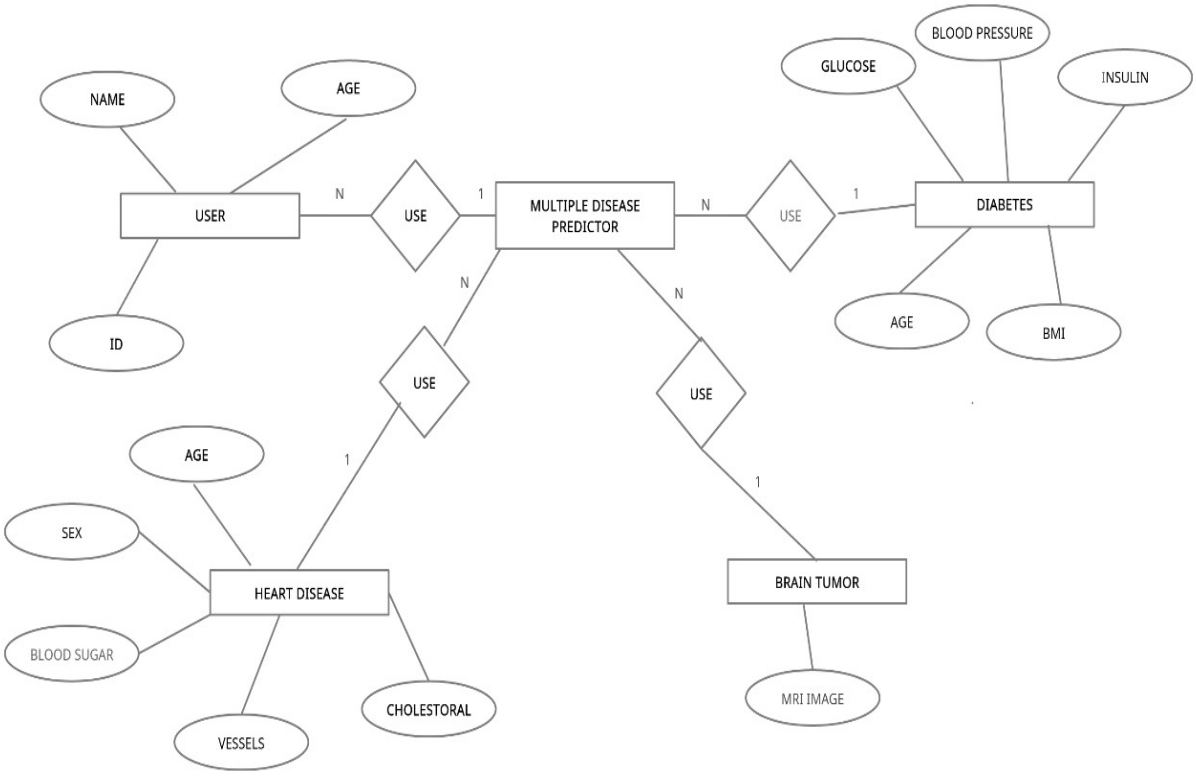

Figure 2-4 Entity Relation Diagram

2.4 System Design

2.4.1 Use Case Diagram

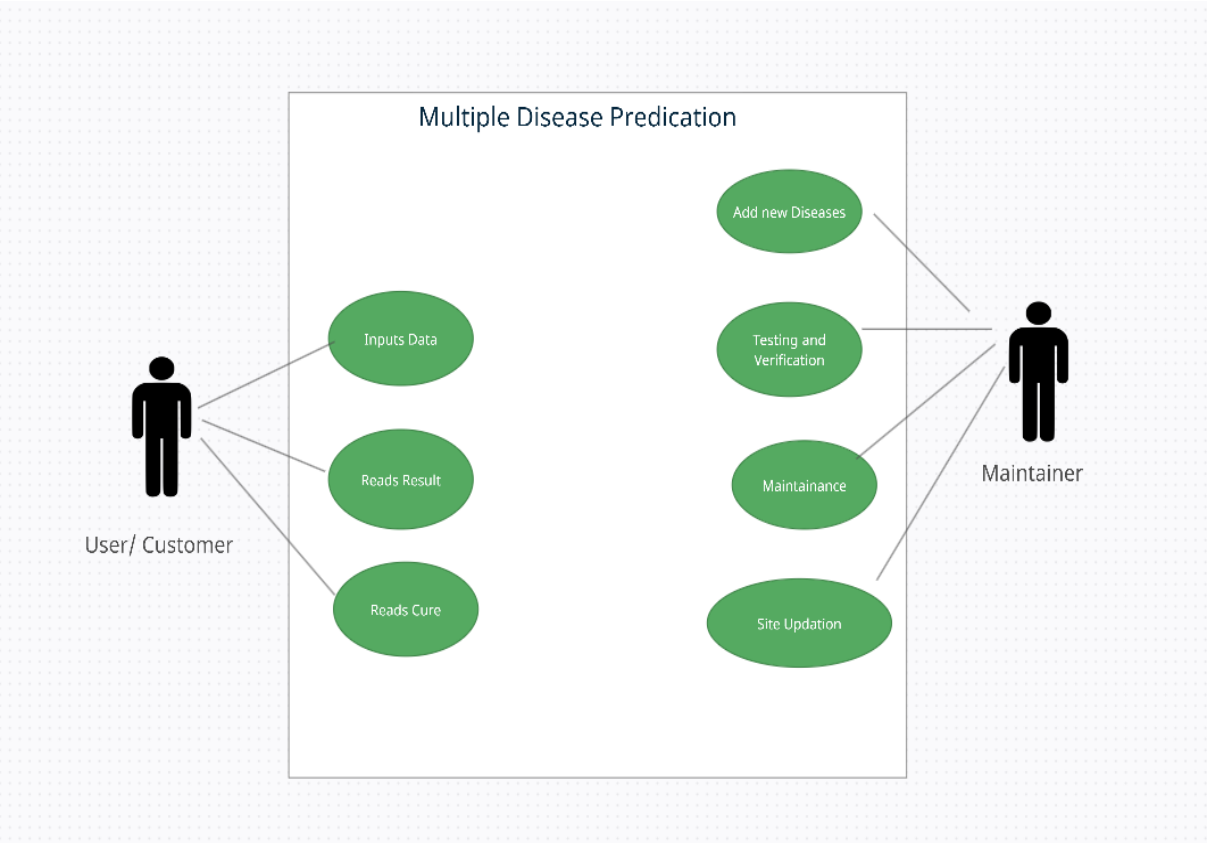

Figure 2-5 Use Case Diagram

2.4.2 Class Diagram

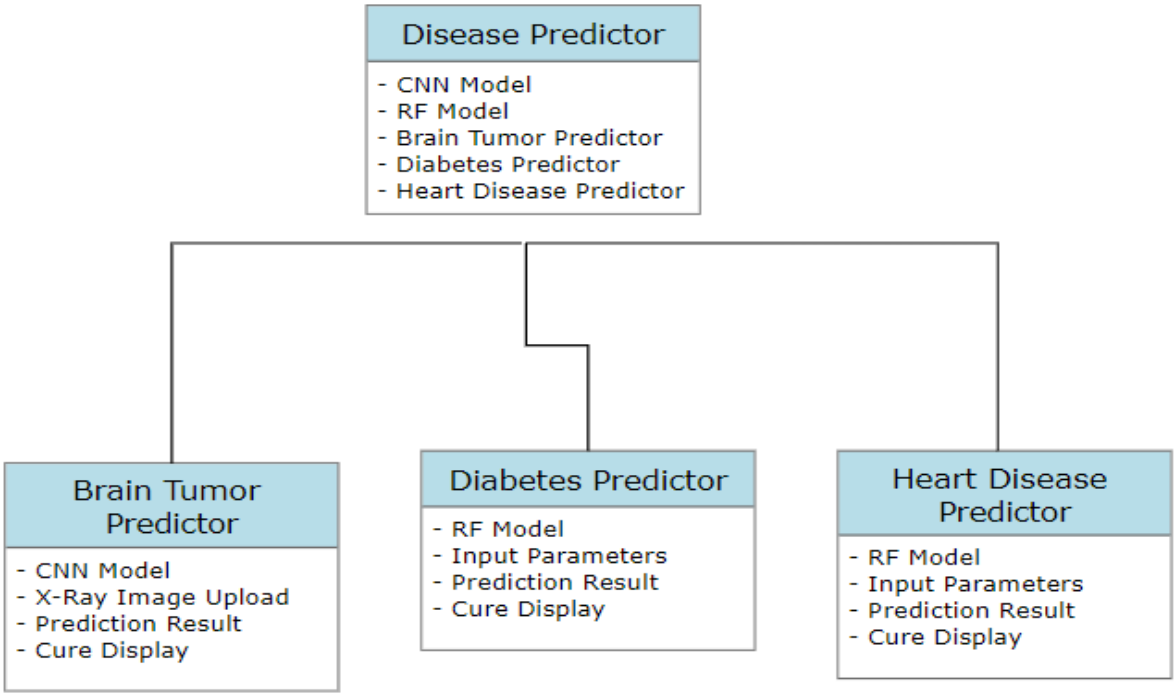

Figure 2-6 Class Diagram

2.4.3 State Chart Diagram

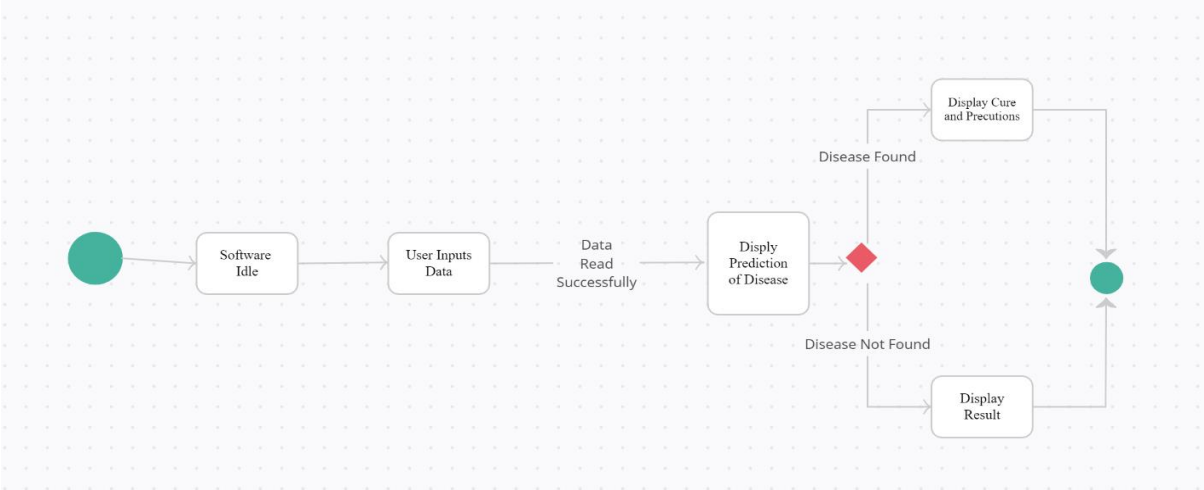

Figure 2-7 State Chart Diagram

2.4.4 Data Flow Diagram

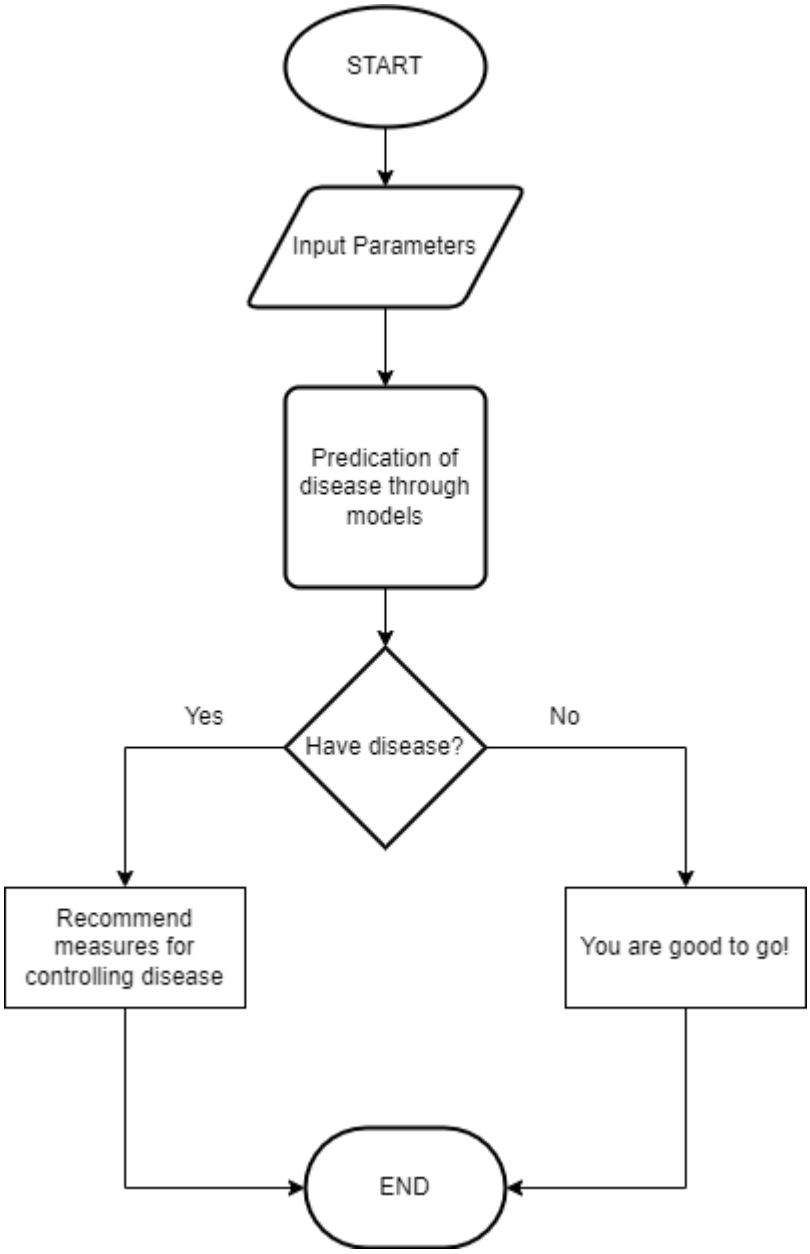

Figure 2-8 Data Flow Diagram

3 Implementation

3.1 Implemented Functionality

3.1.1 Working Methodology

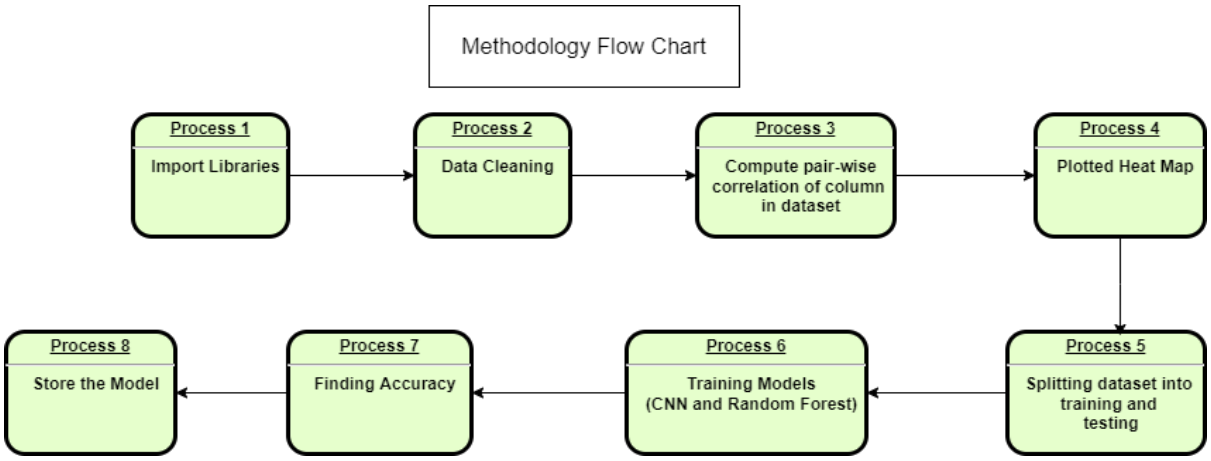

Figure 3-1 Methodology Flow Chart

- **Heart and diabetes disease working:** First, we import libraries which is necessary for prediction of diabetes and heart diseases. Then we add the csv file to Google Collab. Then we check for null values in rows and columns. If there is any null value, then it will be replaced by zero. Then we plotted the Heat-Map to find correlation between various parameter. Then we have divided the dataset into training and testing part. Test=0.2 and training=0.8. Then we apply Random Forest Algorithm to predict the disease. Later, we plotted confusion matrix and calculated the accuracy of the prediction. After that we save the model.

- **Brain tumor disease working**: First, we import libraries which is necessary for prediction of Brain Tumor. Then we mount from google drive like images of training and testing. There are 4 types of tumors like glioma tumor, no tumor, meningioma tumor, pituitary tumor. Then we have done Image Data Augmentation. Image data augmentation is a technique that can be used to artificially expand the size of a training dataset by creating modified versions of images in the dataset. It uses techniques such as flipping, zooming, padding, cropping, etc. Then we Performed One Hot Encoding on the labels after converting it into numerical values: A one hot encoding is appropriate for

categorical data where no relationship exists between categories. After that we have performed Transfer Learning. Deep convolutional neural network models may take days or even weeks to train on very large datasets. A way to short-cut this process is to re-use the model weights from pre-trained models that were developed for standard computer vision benchmark datasets, such as the ImageNet image recognition tasks. After that we have done GlobalAveragePooling2D. This really helps in decreasing the computational load on the machine while training. Then we have divided the dataset into training and testing part. Then we apply CNN Algorithm to predict the disease. Later, calculated the accuracy of the prediction. After that we save the model. We have also created a widget or a button to upload the image and get the result

3.1.2 Brain Tumor Predication (Image Augmentation Implementation)

```
import matplotlib.pyplot as plt
import numpy as np
import pandas as pd
import seaborn as sns
import cv2
import tensorflow as tf
from tensorflow.keras.preprocessing.image import ImageDataGenerator
from tqdm import tqdm
import os
from sklearn.utils import shuffle
from sklearn.model_selection import train_test_split
from tensorflow.keras.applications import EfficientNetB0
from tensorflow.keras.callbacks import EarlyStopping, ReduceLROnPlateau,
TensorBoard, ModelCheckpoint
from sklearn.metrics import classification_report,confusion_matrix
import ipywidgets as widgets
import io
from PIL import Image
from IPython.display import display,clear_output
from warnings import filterwarnings

labels = ['glioma_tumor','no_tumor','meningioma_tumor','pituitary_tumor']

X_train = []
y_train = []
image_size = 150
for i in labels:
```

```
    folderPath =
os.path.join('/content/drive/MyDrive/archive','Training',i)
    for j in tqdm(os.listdir(folderPath)):
        img = cv2.imread(os.path.join(folderPath,j))
        img = cv2.resize(img,(image_size, image_size))
        X_train.append(img)
        y_train.append(i)

for i in labels:
    folderPath = os.path.join('/content/drive/MyDrive/archive','Testing',i)
    for j in tqdm(os.listdir(folderPath)):
        img = cv2.imread(os.path.join(folderPath,j))
        img = cv2.resize(img,(image_size,image_size))
        X_train.append(img)
        y_train.append(i)

X_train = np.array(X_train)
y_train = np.array(y_train)

X_train, y_train = shuffle(X_train,y_train, random_state=101)

datagen = ImageDataGenerator(
    rotation_range=30,
    width_shift_range=0.1,
    height_shift_range=0.1,
    zoom_range=0.2,
    horizontal_flip=True)

datagen.fit(X_train)
X_train.shape

X_train,X_test,y_train,y_test = train_test_split(X_train,y_train,
test_size=0.1,random_state=101)

y_train_new = []
for i in y_train:
    y_train_new.append(labels.index(i))
y_train = y_train_new
y_train = tf.keras.utils.to_categorical(y_train)


y_test_new = []
for i in y_test:
    y_test_new.append(labels.index(i))
y_test = y_test_new
y_test = tf.keras.utils.to_categorical(y_test)
```

```
effnet =
EfficientNetB0(weights='imagenet',include_top=False,input_shape=(image_size
,image_size,3))

model = effnet.output
model = tf.keras.layers.GlobalAveragePooling2D()(model)
model = tf.keras.layers.Dropout(rate=0.5)(model)
model = tf.keras.layers.Dense(4,activation='softmax')(model)
model = tf.keras.models.Model(inputs=effnet.input, outputs = model)

model.summary()

model.compile(loss='categorical_crossentropy',optimizer = 'Adam', metrics=
['accuracy'])

tensorboard = TensorBoard(log_dir = 'logs')
checkpoint =
ModelCheckpoint("effnet.h5",monitor="val_accuracy",save_best_only=True,mode
="auto",verbose=1)
reduce_lr = ReduceLROnPlateau(monitor = 'val_accuracy', factor = 0.3,
patience = 2, min_delta = 0.001,mode='auto',verbose=1)


history = model.fit(X_train,y_train,validation_split=0.1, epochs =12,
verbose=1, batch_size=32,callbacks=[tensorboard,checkpoint,reduce_lr])


pred = model.predict(X_test)
pred = np.argmax(pred,axis=1)

print(classification_report(y_test_new,pred))

#save model
from tensorflow.python.keras.models import load_model
import keras
import keras.utils
from keras import utils as np_utils

keras.models.save_model(model,'tumor_prediction.h5',
overwrite=True,include_optimizer=True)

model.save('tumor_prediction.h5')

def img_pred(upload):
    for name, file_info in uploader.value.items():
        img = Image.open(io.BytesIO(file_info['content']))
    opencvImage = cv2.cvtColor(np.array(img), cv2.COLOR_RGB2BGR)
    img = cv2.resize(opencvImage,(150,150))
```

```
    img = img.reshape(1,150,150,3)
    p = model.predict(img)
    p = np.argmax(p,axis=1)[0]

    if p==0:
        p='Glioma Tumor'
    elif p==1:
        print('The model predicts that there is no tumor')
    elif p==2:
        p='Meningioma Tumor'
    else:
        p='Pituitary Tumor'

    if p!=1:
        print(f'The Model predicts that it is a {p}')

uploader = widgets.FileUpload()
display(uploader)

button = widgets.Button(description='Predict')
out = widgets.Output()
def on_button_clicked(_):
    with out:
        clear_output()
        try:
            img_pred(uploader)

        except:
            print('No Image Uploaded/Invalid Image File')
button.on_click(on_button_clicked)
widgets.VBox([button,out])
```

3.1.3 Diabetes Prediction

```
import pandas as pd
import matplotlib.pyplot as plt
import numpy as np

data = pd.read_csv('/content/diabetes.csv')
data.head()

data.shape

data.isnull().sum()

data.corr()

import seaborn as sns
```

```
plt.figure(figsize=(10,10))
sns.heatmap(data.corr(), annot = True)

X = data.iloc[:,:-1]
y = data['Outcome']

X.shape

y.shape

from sklearn.model_selection import train_test_split

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size = 0.2,
random_state = 42)

print("Train Set: ", X_train.shape, y_train.shape)
print("Test Set: ", X_test.shape, y_test.shape)

from sklearn.ensemble import RandomForestClassifier
model = RandomForestClassifier(n_estimators=20)
model.fit(X_train, y_train)

from sklearn.metrics import accuracy_score

print(accuracy_score(y_test, model.predict(X_test))*100)

import pickle
pickle.dump(model, open("diabetes.pkl",'wb'))
```

3.1.4 Heart Disease Prediction

```
# Commented out IPython magic to ensure Python compatibility.
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
# %matplotlib inline

data = pd.read_csv('/content/heart.csv')

data.head()

data.info()

data.isnull().sum()

import seaborn as sns
```

```
corr = data.corr()

plt.figure(figsize = (15,15))
sns.heatmap(corr, annot = True)

corr

sns.set_style('whitegrid')
sns.countplot(x = 'target', data = data)

dataset = data.copy()
dataset.head()

X = dataset.iloc[:,:-1]
y = dataset['target']

X.columns

from sklearn.model_selection import train_test_split

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size = 0.2,
random_state = 42)

from sklearn.ensemble import RandomForestClassifier
model = RandomForestClassifier(n_estimators=20)
model.fit(X_train, y_train)

pred = model.predict(X_test)
pred[:10]

from sklearn.metrics import confusion_matrix
confusion_matrix(y_test, pred)

from sklearn.metrics import accuracy_score

print(f"Accuracy of model is {round(accuracy_score(y_test, pred)*100,
2)}%")

import pickle
pickle.dump(model, open('heart.pkl', 'wb'))
```

3.2 Results and Reports

The actions performed in this work are done by the laptop with an i5 processor and developed the code using python.

The proposed model uses 3 different models and along with that, the data set we have used are huge with more than 700 values to train the model. On considering the brain tumour predication model we will use a large data set with more than 950 different images for each 3 different type of brain tumours. Not to mention, we achieve good accuracy to make the software reliable for practical use:

- Diabetes Model: 77.27%
- Heart Disease Model: 81.97%
- Brain Tumour Model: 98%

3.3 Snapshots

3.3.1 Main Page Diabetes Predictor

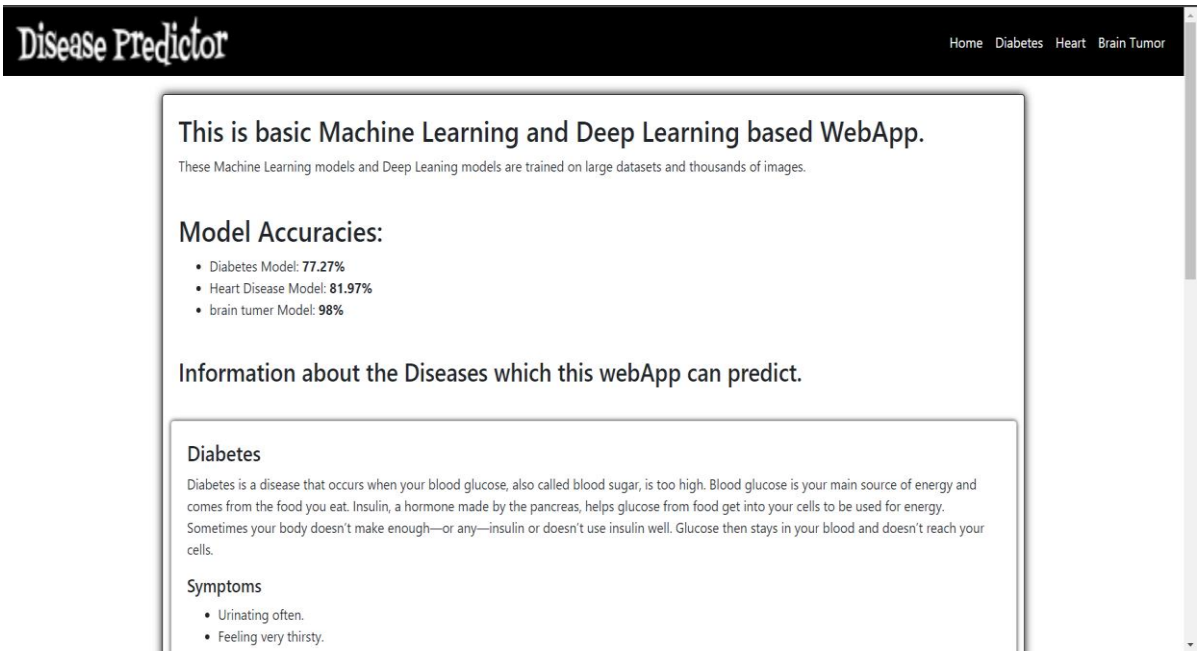

Figure 3-2 Main Page

3.3.2 Diabetes Predictor

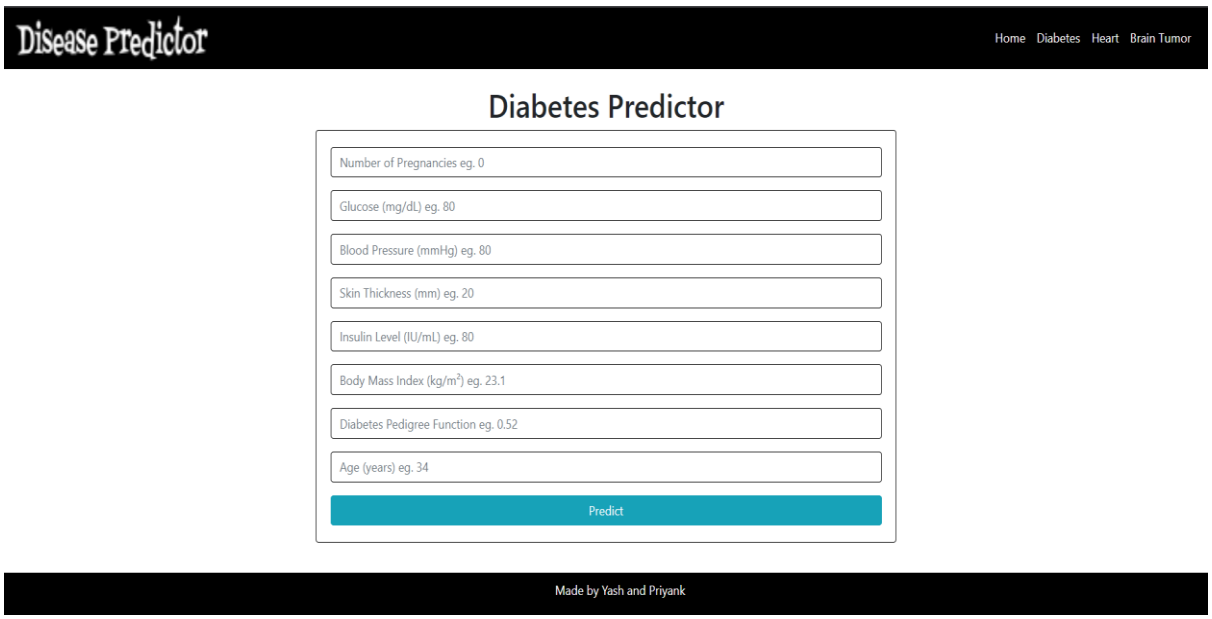

Figure 3-3 Diabetes Predictor Value Input

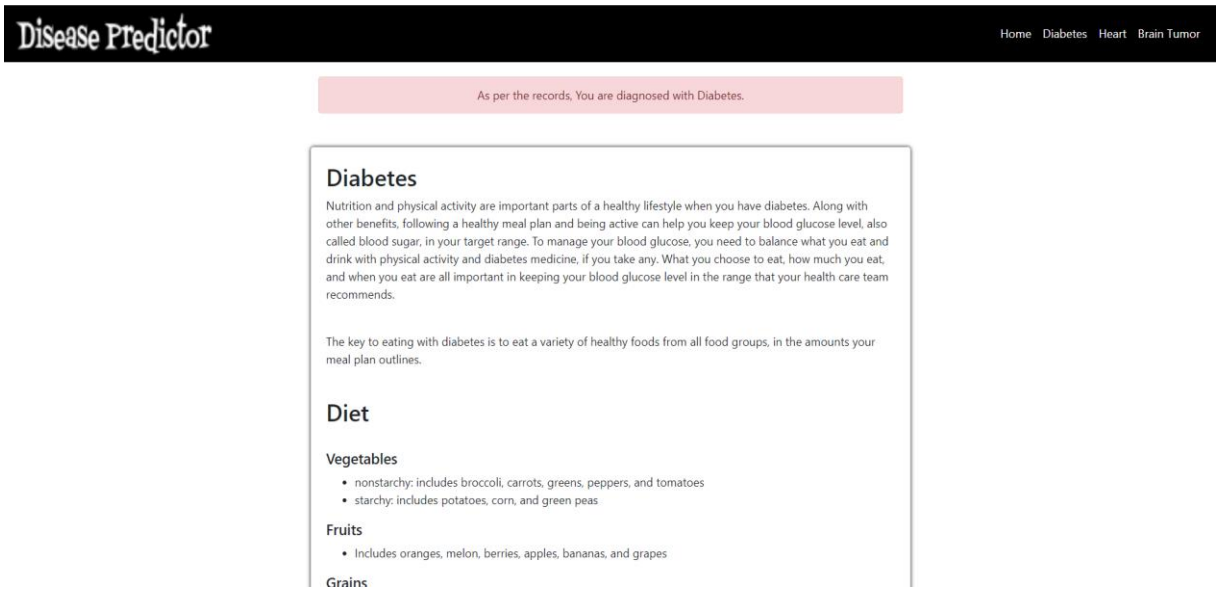

Figure 3-4 Diabetes Predictor Result

3.3.3 Heart Disease Predictor

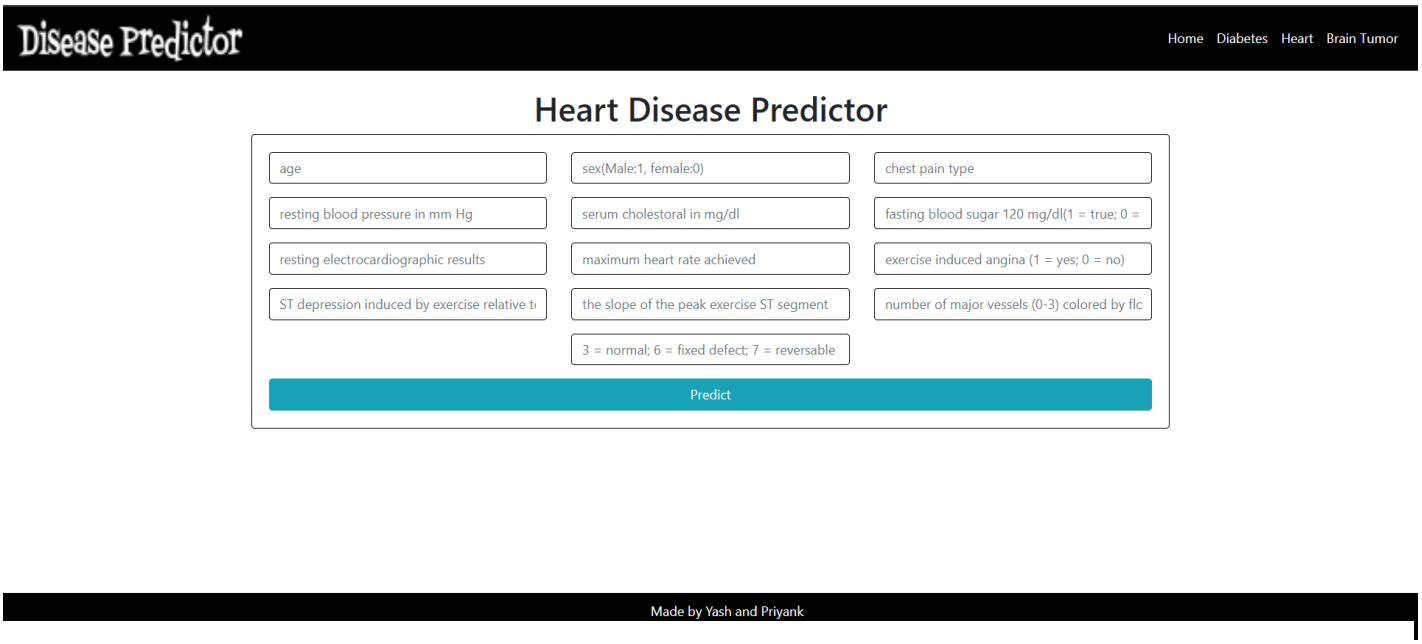

Figure 3-5 Heart Disease Predictor

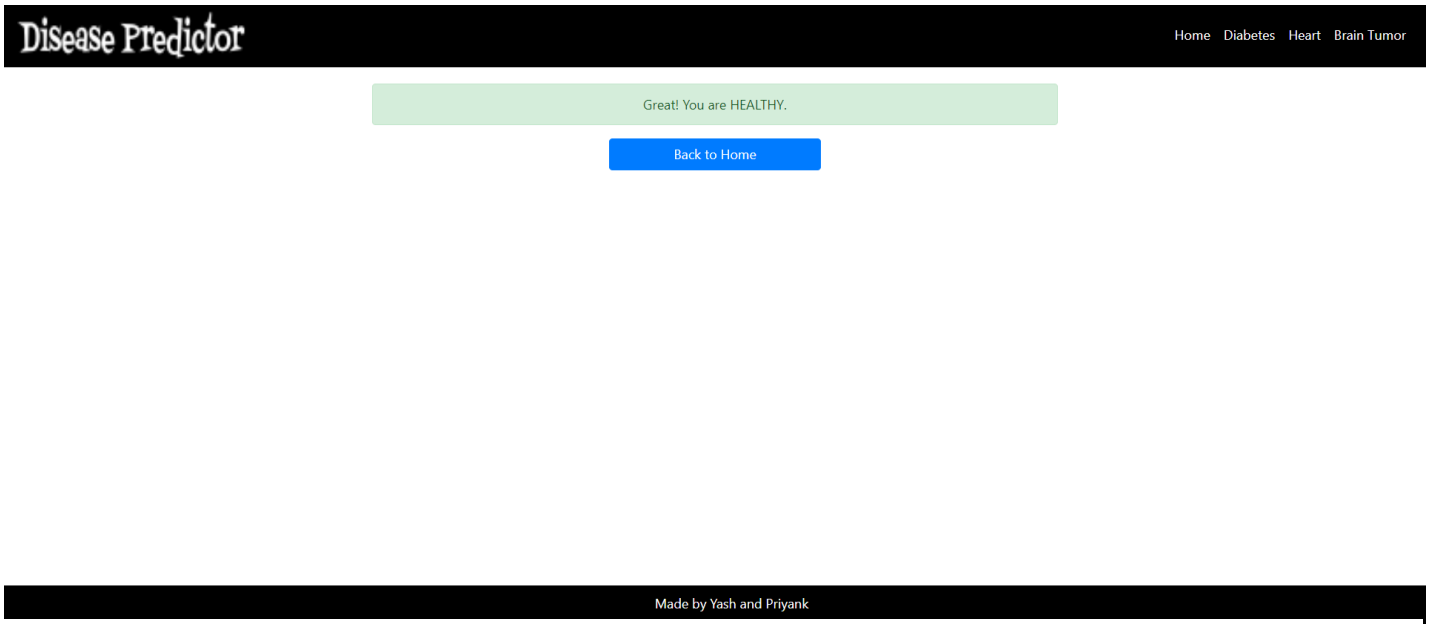

Figure 3-6 Heart Disease Result

3.3.4 Brain Tumor Prediction

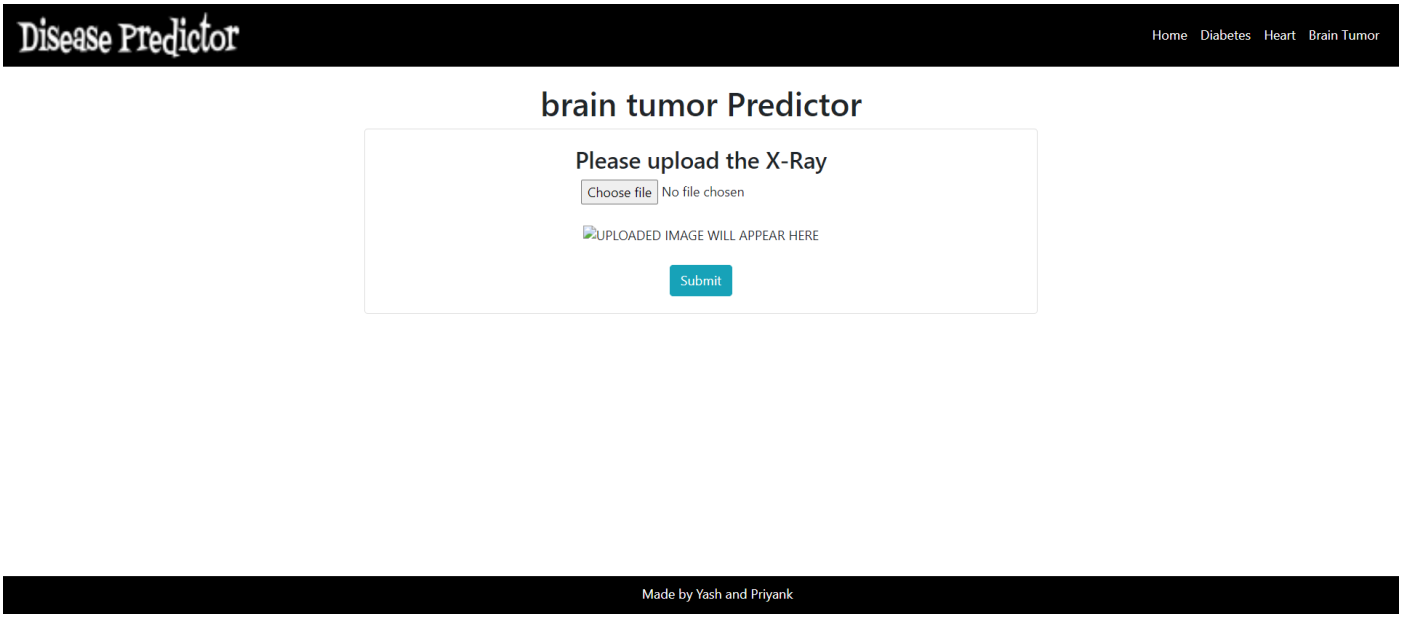

Figure 3-7 Brain Tumor

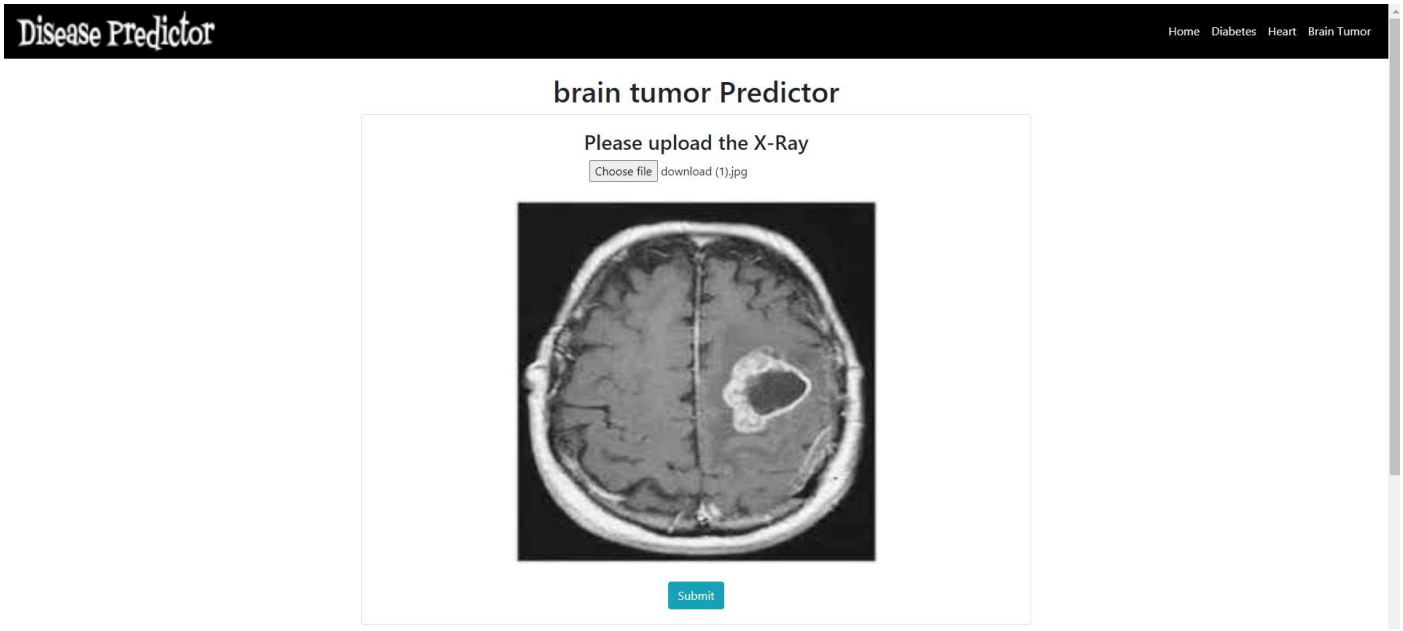

Figure 3-8 Brain Tumor X-Ray Upload

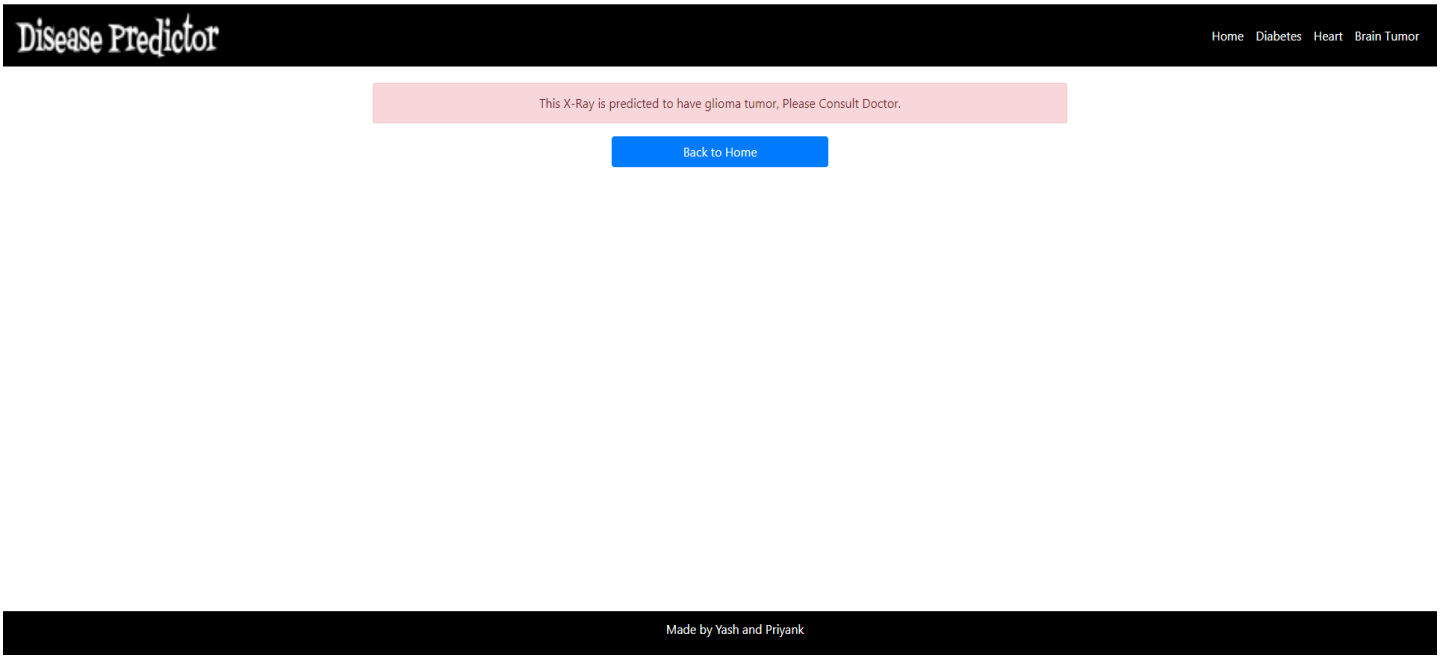

Figure 3-9 Brain Tumor Result

3.4 Testing and Verification

Software testing is a very important step that analyzes the software that is created based on factors such as accuracy, loss, failure or bleu score.

Target of testing are:

- Accuracy – The amount of accuracy the software has based on the results it provides through provided input data
- Loss – The higher the loss indicates that data is loss and not being processed properly, reducing quality of predictions
- Failure – It is said to be the inability of the system to perform the desired task. Failure occurs when faults exist in the system.

Software Testing has a dual function; it is used to identify the defects in program and it is used to help judge whether or not program is usable in practice. Thus, software testing is

used for validation and verification, which ensure that software conforms to its specification and meets need of the software customer.

Developer resorted Alpha testing, which usually comes in after the basic design of the program has been completed. The project scientist will look over the program and give suggestions and ideas to improve or correct the design. They also report and give ideas to get rid of around any major problems. There is bound to be a number of bugs after a program have.

3.4.1 Testing Strategy

There are types of testing that we implement. They are as follows:

- While deciding on the focus of testing activities, study project priorities. For example, for an on-line system, pay more attention to response time.
- Decide on the effort required for testing based on the usage of the system. If the system is to be used by a large number of users, evaluate the impact on users due to a system failure before deciding on the effort.
- A necessary part of the test case is a definition of the expected result.
- Write test cases for invalid and unexpected as well as valid and expected input conditions.
- Thoroughly inspect the results of each test.

We have performed both Block Box Testing and White Box Testing on software to detect and fix errors.

We have solved programming errors in the testing phase and then done verification of each functionality after solving errors

3.4.1.1 Black Box Testing

The technique of testing without having any knowledge of the interior workings of the application is called black-box testing. The tester is oblivious to the system architecture and does not have access to the source code. Typically, while performing a black-box

test, a tester will interact with the system's user interface by providing inputs and examining outputs without knowing how and where the inputs are worked upon.

- **Test Procedures**

  Specific knowledge of the application's code/internal structure and programming knowledge in general is not required. The tester is aware of what the software is supposed to do but is not aware of how it does it. For instance, the tester is aware that a particular input returns a certain, invariable output but is not aware of how the software produces the output in the first place.

- **Test Cases**

  Test cases are built around specifications and requirements, i.e., what the application is supposed to do. Test cases are generally derived from external descriptions of the software, including specifications, requirements and design parameters. Although the tests used are primarily functional in nature, non-functional tests may also be used. The test designer selects both valid and invalid inputs and determines the correct output without any knowledge of the test object's internal structure.

- **Test Design Techniques**

  Typical black-box test design techniques include:

  - Decision table testing
  - All-pairs testing
  - State transition Analysis
  - Equivalence partitioning
  - Boundary value analysis
  - Cause–effect graph
  - Error guessing

3.4.1.2 White Box Testing

White-box testing is the detailed investigation of internal logic and structure of the code. White-box testing is also called glass testing or open-box testing. In order to

perform **white-box** testing on an application, a tester needs to know the internal workings of the code.

The tester needs to have a look inside the source code and find out which unit/chunk of the code is behaving inappropriately.

White-box test design techniques include:

- Control flow testing
- Data flow testing
- Branch testing
- Path testing
- Statement coverage
- Decision coverage

4 Conclusion

4.1 Summary of the results

This project aims to predict the disease on the basis of the symptoms or image (in case of brain tumor disease). The project is designed in such a way that the system takes symptoms or image (in case of brain tumor disease) from the user as input and produces output i.e., predict disease. We have created a website for prediction of diseases using HTML, CSS, python. We have used random forest and CNN algorithms for prediction of disease. We are getting 77.7% accuracy in diabetes prediction, 81.97% accuracy in heart disease prediction and 98.9% accuracy in brain tumor disease prediction.

4.2 Advantages of your Work/ Result/ Methodologies

The existing models are having a combined data of different models. Even though, this approach saves a lot of time of the users but it takes a huge toll on accuracies of predication. On top of that, existing brain tumour models use BraTS dataset which contain almost 290-300 images. In addition to this, they use DCNN model approach which accumulates an accuracy around 85-90%.

The proposed model uses 3 different models and along with that, the data set we have used are huge with more than 700 values to train the model. On considering the brain tumour predication model we will use a large data set with more than 950 different images for each 3 different type of brain tumours. Not to mention, we achieve good accuracy to make the software reliable for practical use:

- Diabetes Model: 77.27%
- Heart Disease Model: 81.97%
- Brain Tumour Model: 98%

On top of that our software is preloaded with many advantageous features such as:

- It is less complex, hence very easy to use.
- It is available 24/7, so that removes the stress of appointment timing with doctors and their availability even at late night.
- Cure of diseases is also displayed along with results so patient can follow the provided doctor prescribed guidelines until meeting with doctor.

4.3 Scope of Future Work

Machine Learning is one of the best career choices of the 21st century. It has plenty of job opportunities with a high-paying salary. Also, the future scope of Machine Learning is on its way to make a drastic change in the world of automation. Further, there is a wide scope of Machine Learning in India. Thus, you can make a lucrative career in the field of Machine Learning to contribute to this growing digital world. In this blog, we will discuss various trends and the future scope of Machine Learning to contribute to this growing digital world. In this blog, we will discuss various trends and the future scope of Machine Learning.

Machine learning allows building models to quickly analyze data and deliver results, leveraging historical and real-time data. With machine learning, healthcare service providers can make better decisions on patient's diagnoses and treatment options, which lead to an overall improvement of healthcare services.

Machine learning algorithms can also be helpful in providing vital statistics, real-time data and advanced analytics in terms of the patient's disease, lab test results, blood pressure, family history, clinical trial data, etc., to doctors. machine learning being processed on computing devices can consider a large number of variables, which results in a better accuracy of healthcare data.

In this project we can further increase the number of diseases which webapp can predict. And decrease the time taken to predict the output. We can also increase dataset by which accuracy will increase. We can also can the layout of the website to make it more attractive. Additionally, we can use more advanced models which might lead to decrease in prediction time taken.

4.4 Unique features of your Project

The existing models are having a combined data of different models. Even though, this approach saves a lot of time of the users but it takes a huge toll on accuracies of predication. On top of that, existing brain tumour models use BraTS dataset which contain almost 290-300 images. In addition to this, they use DCNN model approach which accumulates an accuracy around 85-90%.

The proposed model uses 3 different models and along with that, the data set we have used are huge with more than 700 values to train the model. On considering the brain tumour predication model we will use a large data set with more than 950 different images for each 3 different type of brain tumours. Not to mention, we achieve good accuracy to make the software reliable for practical use:

- Diabetes Model: 77.27%
- Heart Disease Model: 81.97%
- Brain Tumour Model: 98%

Other than that, the project has some additional features such as:

- It is less complex, hence very easy to use.
- It is available 24/7, so that removes the stress of appointment timing with doctors and their availability even at late night.
- Cure of diseases is also displayed along with results so patient can follow the provided doctor prescribed guidelines until meeting with doctor.

4.5 Attainment of POs and PSOs

PO / PSO	Attainment Level	Justification
PO1	3	Note: Write justification about how PO1 is mapped with your project or why it is not mapped. Justification should be based on Attainment Level (0/1/2/3 where 0 means “not mapped” and 3 means “Highly mapped”)
PO2	3	
PO3	3	
PO4	2	
PO5	3	
PO6	2	

PO7	1	
PO8	0	
PO9	1	
PO10	1	
PO11	1	
PO12	1	
PSO1	2	
PSO2	2	

References

1. Priyanka Sonar, Prof. K. Jaya Malini, “Diabetes Prediction Using Different Machine Learning Approaches,” presented at the Proceedings of the Third International Conference on Computing Methodologies and Communication, ICCMC, 2019.
2. Samrat Kumar Dey, Ashraf Hossain and Md. Mahbubur Rahman, “Implementation of a Web Application to Predict Diabetes Disease: An Approach Using Machine Learning Algorithm,” presented at the 21st International Conference of Computer and Information Technology (ICCIT), 2018.
3. Deeraj Shetty, Kishor Rit, Sohail Shaikh and Nikita Patil, “Diabetes Disease Prediction Using Data Mining,” presented at International Conference on Innovations in Information, Embedded and Communication Systems (ICIIECS), 2017.
4. Zhilbert Tafa, Nerxhivane Pervetica and Bertran Karahoda, “An Intelligent System for Diabetes Prediction,” presented at the 4thMediterranean Conference on Embedded Computing MECO, Budva, Montenegro, 2015.

Appendix A – Review Card I and II