M.KUMARASAMY
COLLEGE OF ENGINEERING
NAAC Accredited Autonomous Institution
Approved by AICTE & Affiliated to Anna University
ISO 9001:2015 & ISO 14001:2015 Certified Institution
Thalavapalayam, Karur – 639 113.

A Project Report

on

DETECTION OF PARKINSON'S DISEASE USING MACHINE LEARNING

Submitted in partialfulfillment of requirements for the award of the degree of

BACHELOR OF ENGINEERING

in

COMPUTER SCIENCE AND ENGINEERING

Under the Guidance of

Dr. S. SUJANTHI M.E., PH.D.,
Assistant Professor/CSE

Submitted by

TEAM ID: PNT2022TMID15475

927619BCS4066 – LALITHAA SHREE R

927619BCS4072 – MEGHA K

927619BCS4059 – KESAVAN S

927619BCS4038 – HARIHARAN S

NALAIYA THIRAN – EXPERIENTIAL PROJECT BASED LEARNING INITIATIVE

18CSE040L - PROFESSIONAL READINESS FOR INNOVATION, EMPLOYABILITY AND ENTERPRENURSHIP

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

M.KUMARASAMY COLLEGE OF ENGINEERING, KARUR
(Autonomous)
Karur - 639 113

November, 2022

CHAPTER 1

INTRODUCTION

1.1 PROJECT OVERVIEW

In the present decade of accelerated advances in Medical Sciences, most studies fail to lay focus on ageing diseases. These are diseases that display their symptoms at a much advanced stage and makes a complete recovery almost improbable. Parkinson’s disease (PD) is the second most commonly diagnosed neurodegenerative disorder of the brain. One could argue, that it is almost incurable and inflicts a lot of pain on the patients. All these make it quite clear that there is an oncoming need for efficient, dependable and expandable diagnosis of Parkinson’s disease. The aim of this work is to compare various machine learning models in the successful prediction of the severity of Parkinson’s disease and develop an effective and accurate model in order to help diagnose the disease accurately at an earlier stage which could in turn help the doctors to assist in the cure and recovery of PD Patients. For the aforementioned purpose we plan on using the Parkinson’s Tele monitoring dataset which was acquired from the UCIML repository.

1.2 PURPOSE

The aim of this work is to compare various machine learning models in the successful prediction of the severity of Parkinson's disease and develop an effective and accurate model in order to help diagnose the disease accurately at an earlier stage which could in turn help the doctors to assist in the cure and recovery. This project showed 90% efficiency. In our model, a huge amount of data is collected from the normal person and also previously affected person by Parkinson's disease.

CHAPTER - 2

LITERATURE SURVEY

2.1 EXISTING PROBLEM

S.No.	Author Name	Title	Methods	Description
1	Timothy J, Wroge	Parkinson’s Disease Diagnosis Using Machine Learning and Voice.	Decision support algorithm	It is difficult to detect early due to the subtle initial symptoms.
2	Johannes Frasnelli	Machine Learning for the Diagnosis of Parkinson's Disease.	Machine learning algorithm	Difficulties and to refine the diagnosis and assessment procedures of machine learning methods have been implemented for the classification.
3	Chirag Mittal	Parkinson’s Disease Detection Using Different Machine Learning Algorithms.	K Nearest Neighbors algorithm	Late detection leads to no treatment and loss of life by using this algorithm.

Table 2.1 - Literature Survey

2.2 REFERENCES

1. Chirag Mittal, Parkinson's Disease Detection Using Different Machine Learning Algorithms, International Journal of Scientific and Research Publications, Volume 12, Issue 2, February 2022 23 ISSN 2250-3153.

2. Johannes Frasnelli, Machine Learning for the Diagnosis of Parkinson's Disease, Front. Aging Neurosci, 06 May 2021Sec. Parkinson's Disease and Aging related Movement Disorders.

3. Timothy J, Wroge, Parkinson's Disease Diagnosis Using Machine Learning and Voice. 2018 IEEE Signal Processing in medicine and Biology symposium (SPMB).

2.3 PROBLEM STATEMENT DEFINITION

Problem Statement (PS)	I am (Customer)	I'm trying to	But	Because	Which makesme feel
PS-1	Patient	consult a doctor	I can't consult a doctor	There is more crowd.	Restless
PS-2	Person	Check whether I am a PD patient or not.	I don't know how to recognize	I don't know the method of recognition	I am not withenough knowledge.

Table 2.2 - Problem Statement Definition

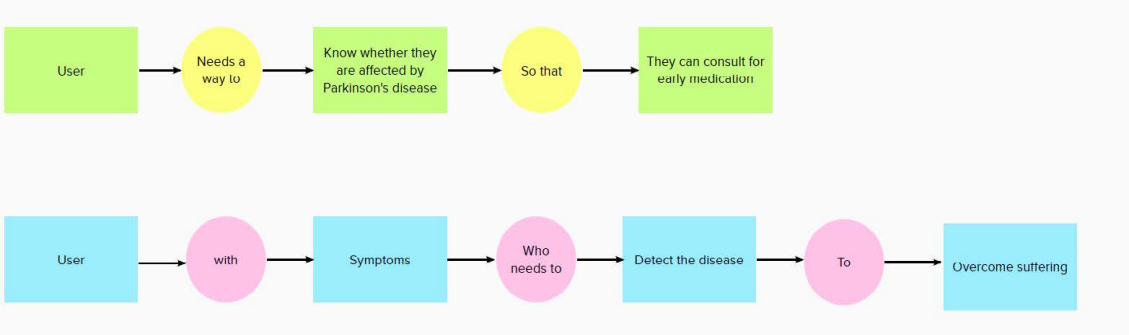

Figure 2.1- Problem Statement

Figure 2.2 - Problem Statement

CHAPTER 3

IDEATION AND PROPOSED SOLUTION

3.1 EMPATHY MAP CANVAS

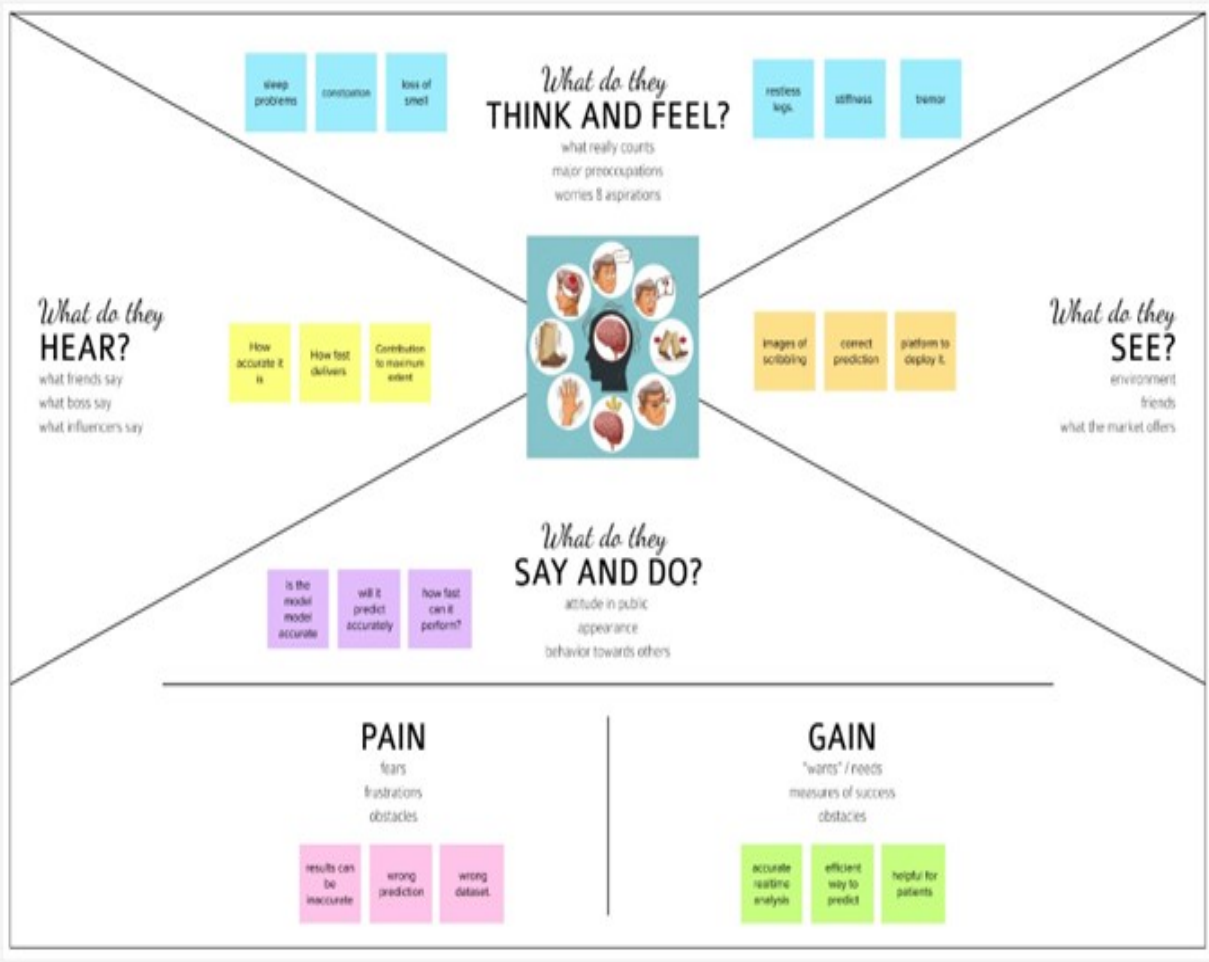

Figure 3.1 - Empathy Map Canvas

3.2 IDEATION AND BRAINSTORMING

Step-1: Team Gathering, Collaboration and Select the Problem Statement

Figure 3.2 - Ideation and Brainstorming

Step-2: Brainstorm, Idea Listing and Grouping

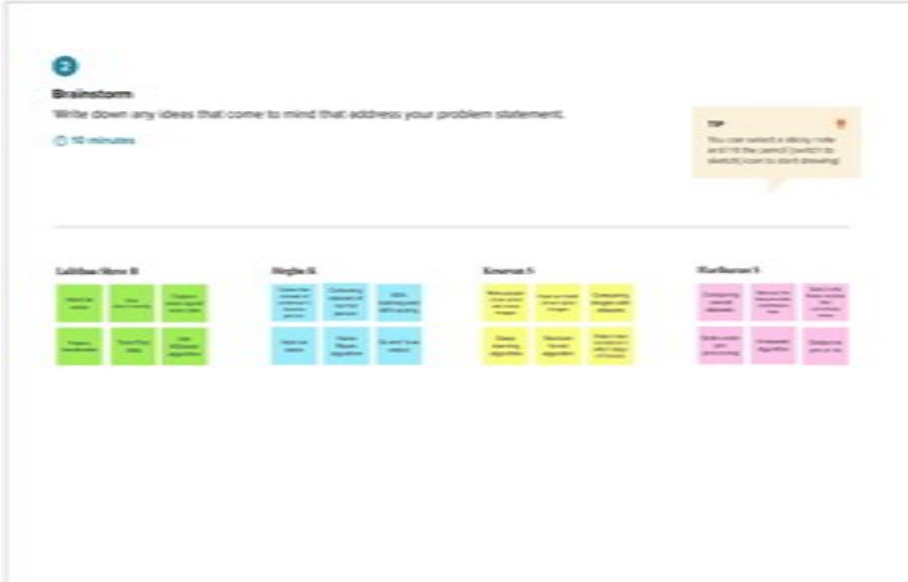

Figure 3.3 - Brainstorm, Idea Listing and Grouping

Step-3 : Idea Prioritization

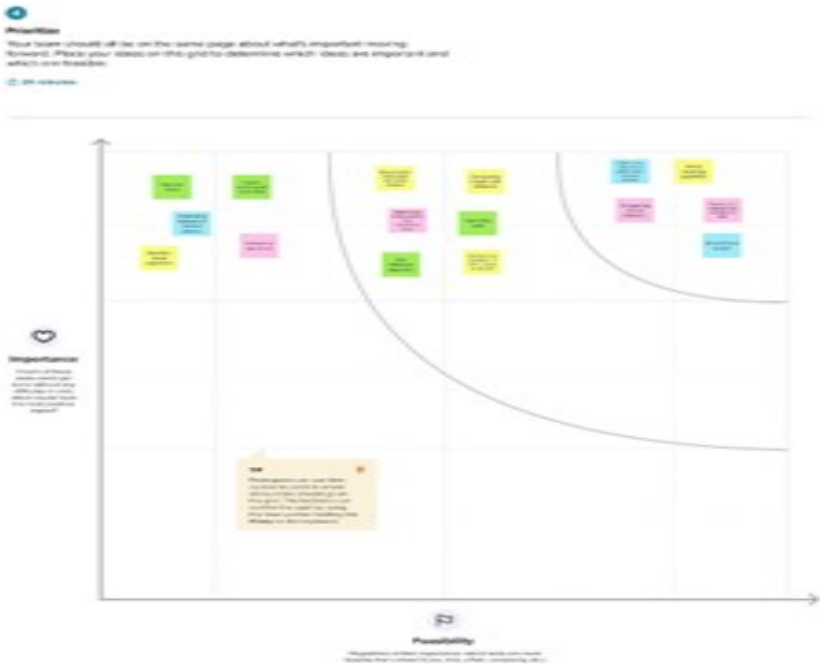

Figure 3.4 - Idea Prioritization

3.3 PROPOSED SOLUTION

S.No.	PARAMETER	DESCRIPTION
1	Problem Statement (Problem to besolved)	Parkinson's disease disorder is a brain disorderthat causesunintended or uncontrollablemovements, such as shaking, stiffness, and difficulty with balance.
2	Idea / Solution description	Studies investigates signals from sustained phonation and text dependent speech modalities for Parkinson's disease screening.Phonation corresponds to the vowel voicing taskand speech.
3	Novelty / Uniqueness	Testing 25 non impulsive patients with Parkinson's disease(PD) and27 PD Patients.
4	Social Impact / Customer Satisfaction	Since it is based on the voice based detection itis very convenient to use. As it helps the people to detect the Parkinson's disease in early stage.
5	Business Model (Revenue Model)	A free platform on the voice based detection itis very convenient to use. As it helps the people to detect the Parkinson's disease in early stage the loss of lifeis prevented.
6	Scalability of the Solution	Additional features can be added anytimeanywhere. Any number of users can access itall at once.

Table 3.5 - Proposed Solution

3.4 PROBLEM SOLUTION FIT

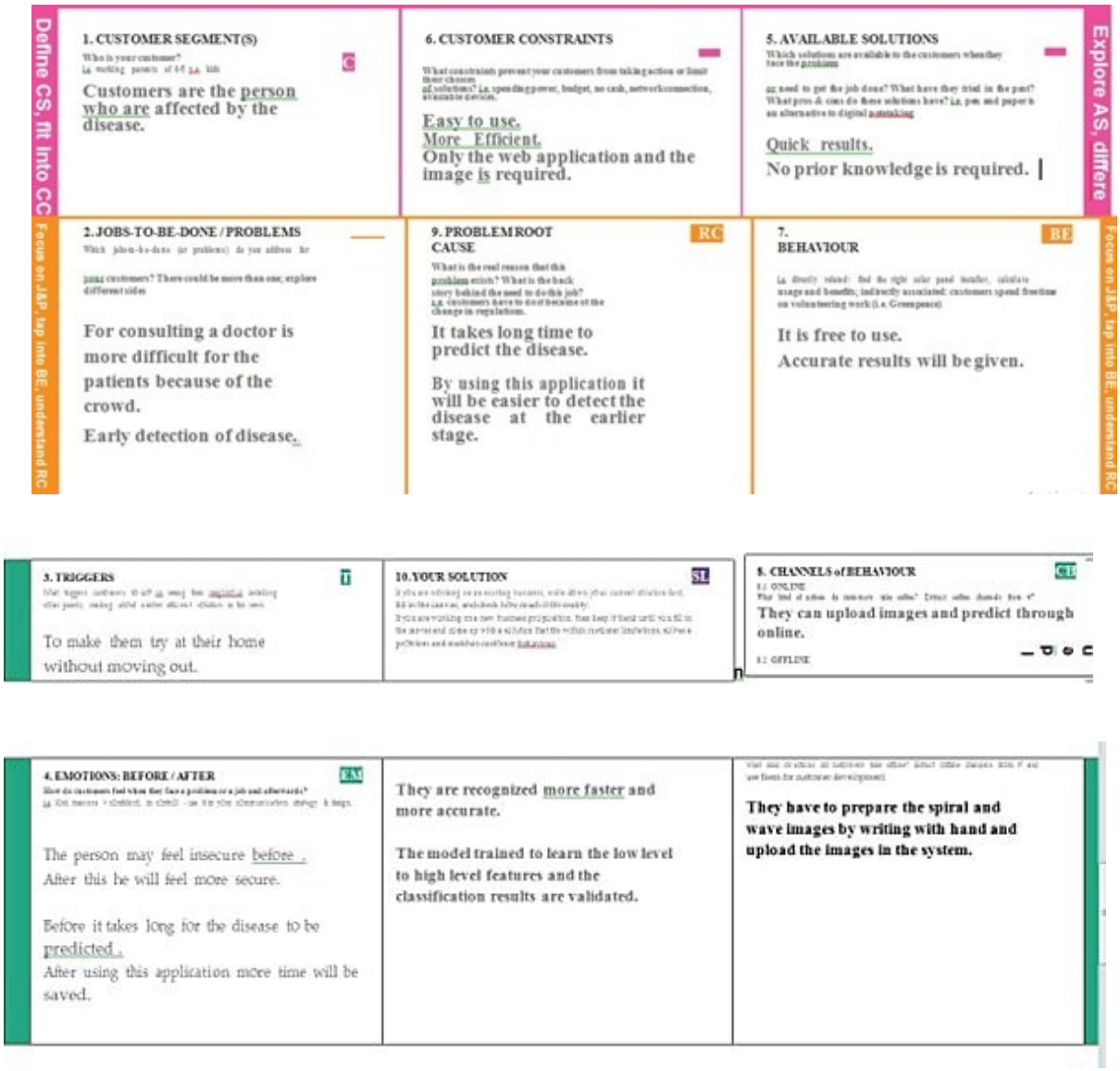

Figure 3.6 - Problem Solution Fit

CHAPTER 4

REQUIREMENT ANALYSIS

4.1 FUNCTIONAL REQUIREMENTS

FR No.	FUNCTIONAL REQUIREMENT	SUB REQUIREMENT
FR-1	User Registration	Registration through Gmail
FR-2	User Confirmation	Confirmation via Email
FR-3	Uploading Dataset	Spiral and wave images are to be uploaded.
FR-4	Requesting Solution	Uploaded images are compared with the pre-defined Model and solution is generated.
FR-5	Downloading Solution	The Output can be downloaded in the pdf format.

Table 4.1 - Functional Requirements

4.2 NON-FUNCTIONAL REQUIREMENTS

NFR.No.	NON-FUNCTIONAL REQUIREMENT	DESCRIPTION
NFR-1	Usability	The user interface screen will be very much user friendly.
NFR-2	Security	The data given by the user will be very secure.
NFR-3	Reliability	Users can access the website all time without any failure
NFR-4	Performance	Load time for the user interface screen
NFR-5	Availability	Maximum downtime will be about 4 hours
NFR-6	Scalability	System can handle

Table 4.2 - Non Functional Requirements

CHAPTER 5

PROJECT DESIGN

5.1 DATAFLOW DIAGRAM

A Data Flow Diagram (DFD) is a traditional visual representation of the information flows within a system.A neat and clear DFD can depictthe right amountof the system requirement graphically. It shows how data enters and leaves the system, what changes the information, and where data is stored.

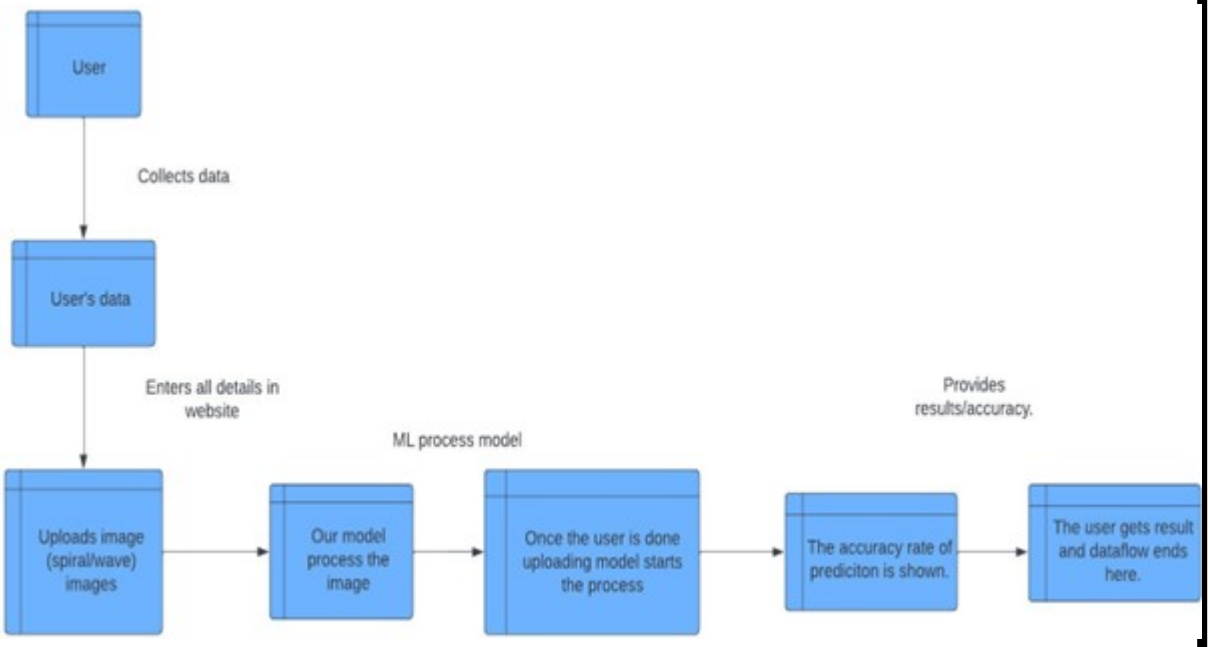

Figure 5.1 - Dataflow Diagram

5.2 SOLUTION AND TECHNICAL ARCHITECTURE

Solution architecture is a complex process – with many sub-processes – that bridges the gap betweenbusiness problems and technology solutions.

Its goals are to:

- Create and login to the IBM Credentials.
- Link the GitHub account with the IBM.
- Notebook downloads from the dataset and imports data to analyses the patients.
- After analyzing the affectedpatients we have to capture the images of them.
- By using Machine Learning Algorithm, we have train and test the data for the further evaluation process.
- After getting out the evaluation processwe have to predict the given model by using Machine Learning.

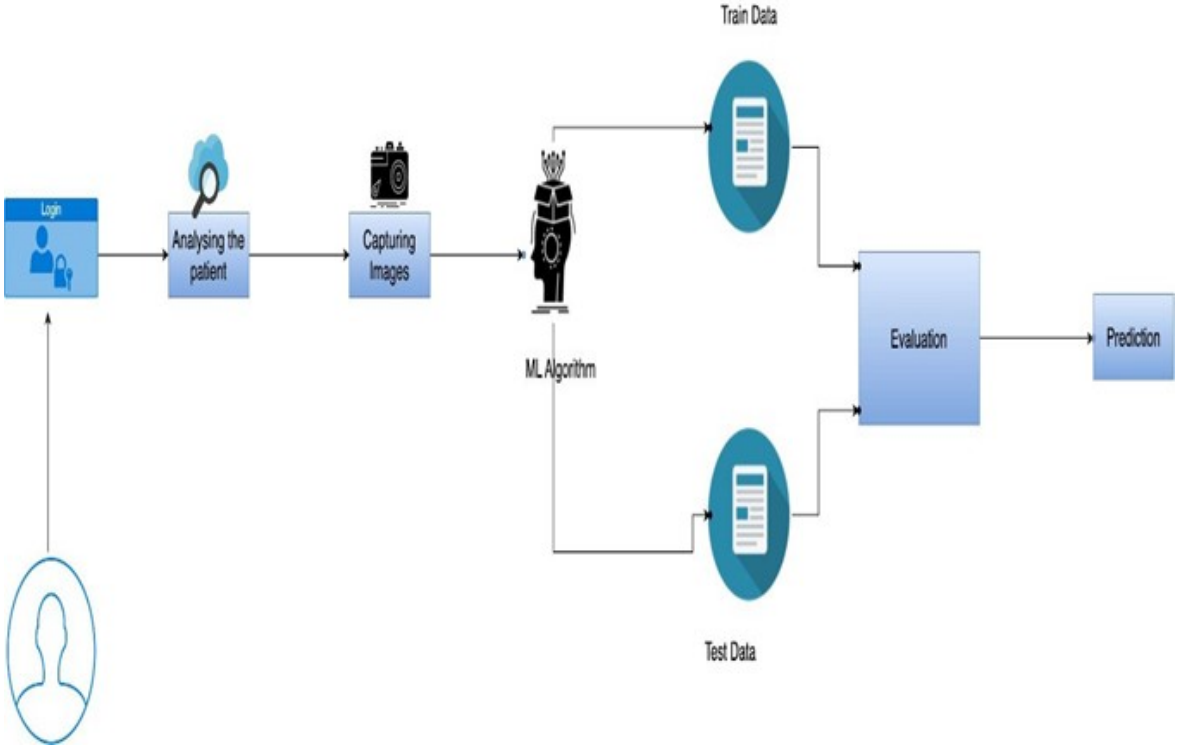

Figure 5.2 - Solution and Technical Architecture

5.3 USER STORY

User Type	Functional Requirement	User Story Number	User Story/Task	Acceptance Criteria	Priority	Release
Customer (Mobile user)	Registration	USN-1	As a user, I can register for the application before entering my email, password, and cofirming my password	I can access my account/ dashboard	High	Sprint-1
	Login	USN-2	As a user, I can log into the application by entering email & password	I can login using my E-mail ID accounts or user credentials	High	Sprint-1
	Dashboard	USN-3	As a user I can view the page of the application where I can upload my images of spiral and wave	I can access my account / dashboard	High	Sprint-2
	Login	USN-5	As a user, I can login to my website dashboard with the login credentials	I can login using my user credentials	High	Sprint-3
Administrator	Login	USN-7	As a admin,I can login to the website using my login credentials	I can login to the website using my login credentials	High	Sprint-1
	Dashboard	USN-8	As a admin, I can view the dashboard if the application	I can access my Dashboard	High	Sprint-2

Figure 5.3 - User Story

CHAPTER 6

PROJECT PLANNING AND SCHEDULING

6.1 SPRINT PLANNING AND ESTIMATION

Sprint	Fynctional Requirement	User Story NUmber	User Story/Task	Story Points	Priority	Team Members
Sprint-1	Data Pre-Processing	USN-1	Collect Dataset	5	High	LalithaaShree
Sprint-1		USN-2	Import the required Libraries,Read & Clean the datasets	5	High	LalithaaShree
Sprint-2	Building the model	USN-1	Split the data into dependent and independent variables	4	High	Megha
Sprint-2		USN-2	Apply the regression Model	2	Medium	Megha
Sprint-3	Application Building	USN-1	Build python flask application and HTML page	2	Medium	Kesavan
Sprint-3		USN-2	Execute and test the application	2	Medium	Kesavan
Sprint-4	Training the model	USN-1	Train machine learning model	5	High	HariHaran
		USN-2	Integrate flask	5	High	HariHaran

Table 6.1 - Sprint Planning and Estimation

6.2 PROJECT DELIVERY SCHEDULE

TITLE	DESCRIPTION	DATE
Ideation Phase	Literature Survey Empathy Map Brainstorming Problem Statement	29 August 2022 30 August 2022 14 September 2022 6 September 2022
Project Design Phase 1	Problem Solution Fit Proposed Solution Solution Architecture	19 September 2022 23 September 2022 28 September 2022
Project Design Phase 2	Requirement Analysis Customer Journey Data Flow Diagrams Technical Architecture	10 October 2022 15 October 2022 3 October 2022 12 October 2022
Project Planning Phase	Sprint Delivery Plan JIRA files	18 October 2022 22 October 2022
Project Development Phase	Sprint 1 Sprint 2 Sprint 3 Sprint 4	29 October 2022 2 November 2022 9 November 2022 16 November 2022

Table 6.2 - Project Delivery Schedule

CHAPTER 7

CODING AND SOLUTION

7.1 FEATURE 1 (Decision Tree Classifier)

Decision Tree Classifier is used to train and test the model for detecting the phishing website with the help of collected and preprocessed dataset collections. NumPy is a library for the Python programming language, adding support for large, multi-dimensional arrays and matrices, along with a large collection of high-level mathematical functions to operate on these arrays. Moreover, NumPy forms the foundation of the Machine Learning stack. Pandas is an open-source Python package that is most widely used for data science/data analysis and machine learning tasks. Sea born is a Python data visualization library based on matplotlib. It provides a high-level interface for drawing attractive and informative statistical graphics. For a brief introduction to the ideas behind the library, you can read the introductory notes or the paper.

Matplotlib is a comprehensive library for creating static, animated, and interactive visualizations in Python. Matplotlib makes easy things easy and hard things possible. Create publication quality plots. Make interactive figures that can zoom, pan, update.EDA is applied to investigate the data and summarize the key insights. It will give you the basic understanding of your data, it is distribution, null values and much more. You can either explore data using graphs or through some python functions. There will be two types of analysis. Descriptive statistics are brief informational coefficients that summarize a given data set, which can be either a representation of the entire population or a sample of a population. Descriptive statistics are broken down into measures of central tendency and measures of variability.

Label Encoding refers to converting the labels into a numeric form to convert them into the machine-readable form. Machine learning algorithms can then decide in a better way how those labels must be operated. It is an important pre-processing step for the structured dataset in supervised learning. “Pickling” is the process whereby a Python object hierarchy is converted into a byte stream, and “unpickling” is the inverse

operation, whereby a byte stream is converted back into an object hierarchy. 19 XGBoost is an optimized distributed gradient boosting library designed to be highly efficient, flexible, and portable. It implements machine learning algorithms under the Gradient Boosting framework.

7.2 FEATURE 2 (Flask Connection)

The framework is the basis upon which software programs are built. It serves as a foundation for software developers, allowing them to create a variety of applications for certain platforms. It is a set of functions and predefined classes used to connect with the system software and handle inputs and outputs. It simplifies the life of a developer while giving them the ability to use certain extensions and makes the online applications scalable and maintainable. Flask is a web application framework written in Python. A Web Application Framework or a simply a Web Framework represents a collection of libraries and modules that enable web application developers to write applications without worrying about lowlevel details such as protocol, thread management, among other examples.

Flask is a web application framework written in Python. Flask is based on the Werkzeg WSGI toolkit and the Jinja2 template engine. Both are Pocco projects. The Web Server Gateway Interface (Web Server Gateway Interface, WSGI) has been used as a standard for Python web application development. WSGI is the specification of a common interface between web servers and web applications. Flask is often referred to as a micro-framework. It is designed to keep the core of the application simple and scalable. Instead of an abstraction layer for database support, Flask supports extensions to add such capabilities to the application. Unlike the Django framework, Flask is very Pythonic. It’s easy to get started with Flask, because it doesn’t have a huge learning curve.HTML stands for Hyper Text Markup Language. HTML is the standard markup language for creating Web pages. HTML describes the structure of a Web page. HTML consists of a series of elements. HTML elements tell the browser how to display the content. Flask is used for developing web applications using python, implemented on Werkzeug and Jinja2. Advantages of using Flask framework are: There is a built-in development server and a fast debugger provided. The model deployed using Flask is used to predict the Chronic Kidney Disease. Hypertext markup language (HTML) is the

basic language used to create documents for the Web and, along 20 with HTTP (hypertext transfer protocol) and URLs (universal resource locators), is one of the three main protocols of the Web. Hypertext is text that contains hyperlinks. A hyperlink is an automated cross-reference to another location on the same document or to another document which, when selected by a user, causes the computer to display the linked location or document within a concise period.

A markup language is a set of tags that can be embedded in digital text to provide additional information about it, including its content, structure and appearance. This information facilitates automated operations on the text, including formatting it for display, searching it and even modifying it. Some type of markup language is employed by every word processing program and by nearly every other program that displays text, although such languages and their tags are typically hidden from the user.HTML consists of a set of predefined tags that can be embedded in text by web site designers in order to indicate the details of how web pages are rendered (i.e., converted into a final, easily usable, form) by web browsers. These details include paragraphing, margins, fonts (including style and size), columns, colors (background and text), links, the location of images, text flow around images, tables, and user input form elements (such as spaces for adding text and submit buttons).

CHAPTER 8

TESTING

8.1 TEST CASES

Test case ID	Feature Type	Component	Test Scenario	Expected Result	Actual Result	Status
HP_TC_001	UI	Home Page	Verify UI elements in the Home Page	The Home page must be displayed properly	Working as expected	Pass
HP_TC_002	UI	Home Page	Check if the UI elements are displayed properly in different screen sizes	The Home page must be displayed properly in all sizes	Working as expected	Pass
HP_TC_003	Functional	Home page	Check if user can upload their file	The input image should be uploaded to the application successfully	Working as expected	Pass
HP_TC_004	Functional	Home page	Check if user cannot upload unsupported files	The application should not allow user to select a non image file	upload any file	Pass

Figure 8.1 - Test Case 1

Test case ID	Feature Type	Component	Test Scenario	Expected Result	Actual Result	Status
HP_TC_005	Functional	Home page	Check if the page redirects to the result page once the input is given	The page should redirect to the results page	Working as expected	Pass
BE_TC_001	Functional	Backend	Check if all the routes are working properly	All the routes should properly work	Working as expected	Pass
M_TC_001	Functional	Model	Check if the model can handle various image sizes	The model should rescale the image and predict the results	Working as expected	Pass
M_TC_002	Functional	Model	Check if the model predicts the image	The model should predict the image	Working as expected	Pass

Figure 8.2 - Test Case 2

Test case ID	Feature Type	Component	Test Scenario	Expected Result	Actual Result	Status
M_TC_003	Functional	Model	Check if the model can handle complex input image	The model should predict the number in the complex image	Working as expected	Pass
RP_TC_001	UI	Result Page	Verify UI elements in the Result Page	The Result page must be displayed properly	Working as expected	Pass
RP_TC_002	UI	Result Page	Check if the input image is displayed properly	The input image should be displayed properly	The size of the input image exceeds the display container	Fail
RP_TC_003	UI	Result Page	Check if the result is displayed properly	The result should be displayed properly	Working as expected	Pass

Figure 8.3 - Test Case 3

8.2 USER ACCEPTANCE TESTING

8.2.1 Defect Analysis

Resolution	Severity 1	Severity 2	Severity 3	Severity 4	Severity 5
By Design	1	0	1	0	2
Duplicate	0	0	0	0	0
External	0	0	2	0	2
Fixed	4	1	0	1	6
Not Reproduced	0	0	0	1	1
Skipped	0	0	0	1	1
Won't Fix	1	0	1	0	2
Total	6	1	4	3	14

Table 8.1 - Defect Analysis

8.2.2 Test Case Analysis

SECTION	TOTAL CASES	NOT TESTED	FAIL	PASS
Client Application	10	0	3	7
Security	2	0	1	1
Performance	3	0	1	2

Table 8.2 - Test Case Analysis

CHAPTER 9

RESULTS

9.1 PERFORMANCE METRICS

Confusion Matrix: The confusion matrix is used to measure the introduction of two class issue for the given instructive record. The right corner to corner parts TP(True positive) and TN (True Negative) adequately describe instances similarly as FP (false positive) and FN (false negative) wrongly request instances. Confusion Matrix correctly classify instance TP+TN incorrectly classify instances.

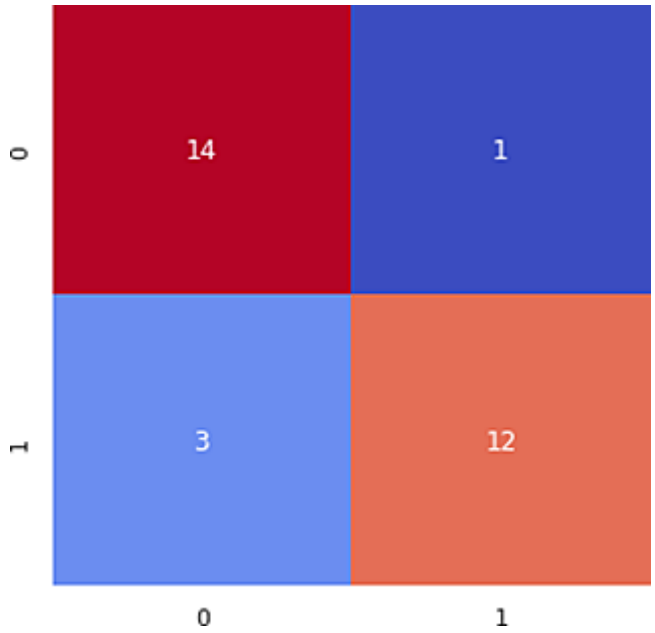

Figure 9.1 Confusion Matrix

Classification Report: is used to measure the quality of predictions from a classification algorithm. How many predictions are True and how many are False. More specifically, True Positives, False Positives, True negatives and False Negatives are used to predict the metrics of a classification report .

- Precision is the ability of a classifier not to label an instance positive that is actually negative. For each class it is defined as the ratio of true positives to the sum of true and false positives.

Precision = TP/(TP + FP)

➤ Recall is the ability of a classifier to find all positive instances. For each class it is defined as the ratio of true positives to the sum of true positives and false negatives.

Recall = TP/(TP+FN)

➤ The F1 score is a weighted harmonic mean of precision and recall such that the best score is 1.0 and the worst is 0.0. Generally speaking, F1 scores are lower than accuracy measures as they embed precision and recall into their computation. As a rule of thumb, the weighted average of F1 should be used to compare classifier models, not global accuracy.

F1 Score = 2*(Recall * Precision) / (Recall + Precision)

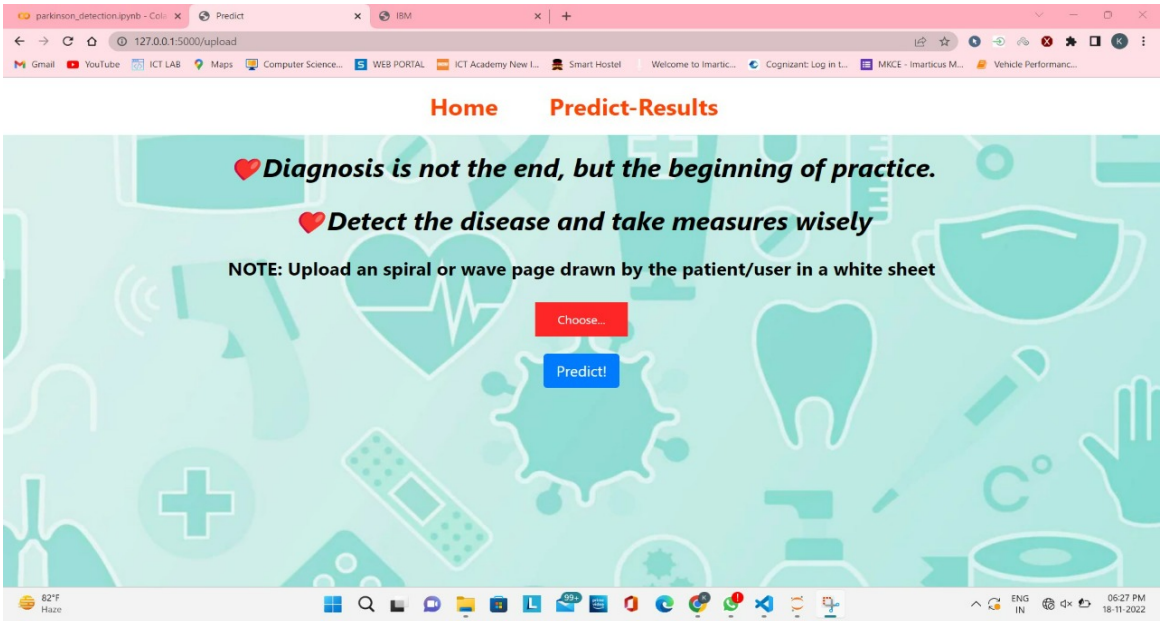

Figure 9.2

CHAPTER 10

ADVANTAGES AND DISADVANTAGES

ADVANTAGES

a. Reduces manual work

b. More accurate than average human

c. Capable of handling a lot of data

d. Can be used anywhere from any device.

DISADVANTAGES

a. Cannot handle complex data

b. All the data must be in image format

c. Requires a high performance server for faster predictions

d. Prone to occasional errors

CHAPTER 11

CONCLUSION

Parkinson’s Disease is a totally grave disease and has no cure till date. since it impacts the actions of the parts of the body, the speech additionally stands affected. here, the gadget tries to offer a way of detecting Parkinson’s ailment so one can bring about a quick action to reduce or even put off it from affecting the whole body. This gadget aims to make this method of expertise a case of Parkinson’s on the earliest via each, the affected person as well as scientific experts. hence, the goal is to apply numerous machine getting to know strategies like Random Forest Classifier , CNN, for buying the maximum accurate result. Here using Decision Tree and buildinga classifier results in an accuracy of 98%.

CHAPTER 12

FUTURE SCOPE

This project is far fromcomplete and there is a lot of room for improvement. Some of the improvements that can be made to this project are as follows:

i. Add support to detect from multiple imagesand save the results
ii. Add support to detect multipleimages
iii. Improve model to detect from complex images

This project has endless potentialand can always be enhancedto become better. Implementing this concept in the real world will benefit several industries and reduce the workload on many workers, enhancing overall work efficiency.

CHAPTER 13

APPENDIX

Importing the Necessary Libraries

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import zipfile as zf
import os
import random
import cv2
import pickle
from imutils import build_montages
from imutils import paths
from sklearn.metrics import classification_report,confusion_matrix
from sklearn import metrics
from sklearn.preprocessing import LabelEncoder,LabelBinarizer
from sklearn.model_selection import train_test_split
from sklearn.ensemble import
RandomForestClassifier,GradientBoostingClassifier,ExtraTreesClassifier
from skimage import feature
from google.colab.patches import cv2_imshow
```

Loading the training and testing dataset

```
handle_spiral = zf.ZipFile(r&#39;dataset1.zip&#39;)
handle_spiral.extractall(&#39;dataset1&#39;)
handle_spiral.close()
```

```
spiral_train_healthy = os.listdir(&#39;dataset1/dataset/spiral/training/hea
lthy/&#39;)
spiral_train_park = os.listdir(&#39;dataset1/dataset/spiral/training/parkin
son/&#39;)
fp_spiral_train_healthy = &#39;dataset1/dataset/spiral/training/healthy/&#39;
fp_spiral_train_park = &#39;dataset1/dataset/spiral/training/parkinson/&#39;
spiral_test_healthy = os.listdir(&#39;dataset1/dataset/spiral/testing/healt
hy/&#39;)
spiral_test_park = os.listdir(&#39;dataset1/dataset/spiral/testing/parkinso
n/&#39;)
fp_spiral_test_healthy = &#39;dataset1/dataset/spiral/testing/healthy/&#39;
fp_spiral_test_park = &#39;dataset1/dataset/spiral/testing/parkinson/&#39;
```

Quantifying Images

```
def quantify_image(image):
  features = feature.hog(image,orientations=9,
            pixels_per_cell=(10,10),cells_per_block=(2,2),transform
_sqrt=True,block_norm=&quot;L1&quot;)
  return features
```

Splitting up of training and testing data

```
trainX = []
testX = []
outputs = []
trainY = []
testY = []
for i in spiral_train_healthy:
  image = cv2.imread(fp_spiral_train_healthy+i)
  image = cv2.cvtColor(image , cv2.COLOR_BGR2GRAY)
```

```
  image = cv2.resize(image , (200,200))
  image =cv2.threshold(image, 0, 255,cv2.THRESH_BINARY_INV | cv2.THRESH
_OTSU)[1]
  features = quantify_image(image)
  trainX.append(features)
  trainY.append(&amp;#39;healthy&amp;#39;)
 for i in spiral_train_park:
  image = cv2.imread(fp_spiral_train_park+i)
  image = cv2.cvtColor(image , cv2.COLOR_BGR2GRAY)
  image = cv2.resize(image , (200,200))
  image = cv2.threshold(image ,0,255,cv2.THRESH_BINARY_INV | cv2.THRESH
_OTSU)[1]
  features = quantify_image(image)
  trainX.append(features)
  trainY.append(&amp;#39;parkinson&amp;#39;)
  for i in spiral_test_healthy:
  image = cv2.imread(fp_spiral_test_healthy+i)
  outputs.append(image)
  image = cv2.cvtColor(image , cv2.COLOR_BGR2GRAY)
  image = cv2.resize(image , (200,200))
  image = cv2.threshold(image ,0,255,cv2.THRESH_BINARY_INV | cv2.THRESH
_OTSU)[1]
  features = quantify_image(image)
  testX.append(features)
  testY.append(&amp;#39;healthy&amp;#39;)
 for i in spiral_test_park:
  image = cv2.imread(fp_spiral_test_park+i)
  outputs.append(image)
  image = cv2.cvtColor(image , cv2.COLOR_BGR2GRAY)
  image = cv2.resize(image , (200,200))
  image = cv2.threshold(image ,0,255,cv2.THRESH_BINARY_INV | cv2.THRESH
```

```
_OTSU)[1]
  features = quantify_image(image)
  testX.append(features)
  testY.append(&#39;parkinson&#39;)
trainX = np.array(trainX)
testX = np.array(testX)
trainY = np.array(trainY)
testY = np.array(testY)
trainX
trainY
```

Label Encoding

```
le = LabelEncoder()
trainY = le.fit_transform(trainY)
testY = le.transform(testY)
print(trainX.shape,trainY.shape)
trainY
testY
```

Model Building

Training the model

```
print(&quot;Training model....&quot;)
model = RandomForestClassifier(n_estimators=100)
model.fit(trainX,trainY)
preds = model.predict(testX)
preds
```

Model Evaluation

```
cnf = confusion_matrix(testY,preds)
cnf
```

```
array([[14, 1], [ 3, 12]]) plt.figure(figsize=(5,5))
sns.heatmap(cnf , annot=True , cmap=&quot;coolwarm&quot; , cbar=False)
plt.show()

acc = metrics.accuracy_score(testY,preds)
acc
indexes = np.random.randint(0,30,25)
indexes
```

Testing Model

```
testpath=list(paths.list_images(fp_spiral_train_healthy))
idxs=np.arange(0,len(testpath))
idxs=np.random.choice(idxs,size=(25,),replace=False)
images=[]
for i in idxs:
   image=cv2.imread(testpath[i])
   output=image.copy()
   output=cv2.resize(output,(128,128))
   image=cv2.cvtColor(image,cv2.COLOR_BGR2GRAY)
   image=cv2.resize(image,(200,200))
                image=cv2.threshold(image,0,255,cv2.THRESH_BINARY_INV     |
cv2.THRESH_
OTSU)[1]
   features= quantify_image(image)
   preds=model.predict([features])
   label=le.inverse_transform(preds)[0]
   if label==&quot;healthy&quot;:
color=(0,255,0)
   else:
     (0,0,255)
```

```
    cv2.putText(output,label, (3,20),cv2.FONT_HERSHEY_SIMPLEX,0.5,color
,2)
    images.append(output)
&#39;&#39;&#39;montage = build_montages(images,(128,128),(5,5))[0]
cv2.imshow(montage)
cv2.waitKey(0)&#39;&#39;&#39;
montage=build_montages(images,(128,128),(5,5))[0]
cv2_imshow(montage)
cv2.waitKey(0)
predictions = model.predict(testX)
cm = confusion_matrix(testY, predictions).flatten()
print(cm)
(tn, fp, fn, tp) = cm
accuracy = (tp + tn) / float(cm.sum())
print(accuracy)
```

Flask App

```
from flask import Flask, request, render_template
import pickle
import cv2
from skimage import feature
import os.path
#from werkzeug.utils import secure_filename

#from model import model


app = Flask(__name__)


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

```
    return render_template("home.html")


@app.route("/home")
def home():
    return render_template("home.html")


@app.route("/upload")
def test():
    return render_template("pred.html")


@app.route("/logout")
def log():
    return render_template("home.html")


@app.route('/predict', methods=['GET', 'POST'])
def upload():
    if request.method == 'POST':
        f = request.files['file']  # requesting the file
        #filename_secure = secure_filename(f.filename)
        basepath = os.path.dirname(
            '__file__')  # storing the file directory
        # storing the file in uploads folder
        filepath = os.path.join(basepath, "uploads", f.filename)
        f.save(filepath)  # saving the file

        # Loading the saved model
        print("[INFO] loading model...")
        model = pickle.loads(open('parkinson.pkl', "rb").read())
        '''local_filename = "./uploads/"
        local_filename += filename_secure
        print(local_filename)'''

        # Pre-process the image in the same manner we did earlier
        image = cv2.imread(filepath)
        output = image.copy()

        # Load the input image, convert it to grayscale, and resize
        output = cv2.resize(output, (128, 128))
        image = cv2.cvtColor(image, cv2.COLOR_BGR2GRAY)
        image = cv2.resize(image, (200, 200))
```

```
        image = cv2.threshold(image, 0, 255,
                        cv2.THRESH_BINARY_INV | cv2.THRESH_OTSU)[1]

        # Quantify the image and make predictions based on the extracted features using the last trained
Random Forest
        features = feature.hog(image, orientations=9,
                           pixels_per_cell=(10, 10), cells_per_block=(2, 2),
                           transform_sqrt=True, block_norm="L1")
        preds = model.predict([features])
        print(preds)
        ls = ["healthy", "parkinson"]
        result = ls[preds[0]]
        '''color = (0, 255, 0) if result == "healthy" else (0, 0, 255)
        cv2.putText(output, result, (3, 20),
                    cv2.FONT_HERSHEY_SIMPLEX, 0.5, color, 2)
        cv2.imshow("Output", output)
        cv2.waitKey(0)'''
        return result
    return None


if __name__ == '__main__':
    app.run()
```

Home Page Html

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
    <meta http-equiv="X-UA-Compatible" content="ie=edge" />
    <title>HomePage</title>
    <style>
      body {
        background: linear-gradient(to right, #33ccff 0%, #99ffcc 100%);
        background-size: cover;
        background-position: relative;
        background-repeat: no-repeat;
        height: 100%;
        width: 100%;
      }
      h3 {
        text-align: center;
```

```
  color: white;
}
.main {
  margin-top: 100px;
}
p {
  color: black;
  text-indent: 10px;
  margin: 10px;
  font-size: 20px;
}

a {
  color: grey;
  float: right;
  text-decoration: none;
  font-style: normal;
  padding-right: 20px;
}

a:hover {
  background-color: black;
  color: white;
  font-size: 30px;
  padding-left: 10px;
  border-radius: 5px;
}

ul {
  align-items: center;
  display: flex;
  list-style-type: none;
  width: 100%;
  gap: 3rem;
  justify-content: center;
  font-size: 2rem;
  position: fixed;
  top: 0;
  margin: 0;
  padding: 1rem;
  background-color: white;
}

li {
```

```
    cursor: pointer;
   }
   li a {
    text-decoration: none;
    color: inherit;
   }
   li.active {
    font-weight: bold;
    color: orangered;
   }

   img {
    width: 450px;
    height: 400px;
    padding: 25px;
   }
   img:hover {
    border-color: grey;
   }
   #im {
    width: 1450px;
    height: 700px;
    padding: 25px;
   }
  </style>
</head>
<body>
  <nav>
   <ul>
    <li class="active"><a href="/home">Home</a></li>
    <li class="active"><a href="/upload">Predict-Results</a></li>
   </ul>
  </nav>
  <br /><br /><br />
  <h1>
   <center>
    <b class="pd"
     ><font color="black" size="15" font-family="Comic Sans MS"
      >Detection of Parkinson's Disease using ML</font
     ></b
    >
   </center>
  </h1>
  <div>
```

```
<center>
  <p style="text-align: left">
    Parkinson disease (PD) is a progressive neuro degenerative disorder
    that impacts more than 6 million people around the world. Parkinson's
    disease is non-communicable, early-stage detection of Parkinson's can
    prevent further damages in humans suffering from it.
    However,Nonetheless, non-specialist physicians still do not have a
    definitive test for PD, similarly in the early stage of the diseased
    person where the signs may be intermittent and badly characterized. It
    resulted in a high rate of misdiagnosis (up to 25% among
    non-specialists) and many years before treatment, patients can have
    the disorder. A more accurate, unbiased means of early detection is
    required, preferably one that individuals can use in their home
    setting.However, it has been observed that PD's presence in a human is
    related to its hand-writing as well as hand-drawn subjects. From that
    perspective, several techniques have been proposed by researchers to
    detect Parkinson's disease from hand-drawn images of suspected people.
    But the previous methods have their constraints.
  </p>
</center>
<h4>
  <center>
    <b class="pd"
      ><font color="black" size="12" font-family="Comic Sans MS"
        >Causes and Symptoms of Parkinson's Disease</font
      >
    </b>
  </center>
</h4>
<span>
  <img
    src="https://www.narayanahealth.org/blog/wp-content/uploads/2015/04/parkinson.png"
    title="Disease"
  />
</span>
<span>
  <img

src="https://stanfordmedicine25.stanford.edu/the25/parkinsondisease/_jcr_content/main/panel_builder_0/panel_0/panel_builder_0/panel_0/panel_builder/panel_0/image.img.476.high.png/1.png"
    title="Symptoms"
/></span>
<span
  ><img
```

src="https://www.verywellhealth.com/thmb/Aaqo8oM3QDHSNHCt_DlKCNeWoUk=/1500x0/filters:no_upscale():max_bytes(150000):strip_icc()/zhansen-5200700_Finaledit2-3e7eb00f1bdb4806adb3f67ca4404894.jpg"
title="Stages"
/>
<span
><img
src="https://www.gutmicrobiotaforhealth.com/wp-content/uploads/2016/12/parkinson.jpg"
title="Effect"
/>
<span
><img
src="https://i.pinimg.com/originals/02/16/e4/0216e4b8a5db4d6e2a3f7043eaf7dc32.jpg"
title="Cause"
/>
<span
><img
src="https://jnnp.bmj.com/content/jnnp/91/8/795/F4.large.jpg"
title="diagnosis"
/>
<h3>

<font color="black" size="12" font-family="Comic Sans MS"
>Treatment for parkinson disease

</h3>
<span
><img
src="https://www.mdpi.com/biomolecules/biomolecules-11-00612/article_deploy/html/images/biomolecules-11-00612-g001.png"
title="diagnosis"
/>
<span
><img
src="https://media.springernature.com/m685/springer-static/image/art%3A10.1038%2Fs41401-020-0365-y/MediaObjects/41401_2020_365_Fig1_HTML.png"
title="diagnosis"
/>
<span
><img
src="https://www.verywellhealth.com/thmb/BgjmOKb2W-7z0gqLZryKBd4FFHs=/1500x0/filters:no_upscale():max_bytes(150000):strip_icc()/advanced-

```
parkinsons-disease-5200544_color_text_v1-3bc74418259340ceaf5f6d407daeff73.jpg"
        title="diagnosis"
      /></span>

      <h3>
        <center>
          <font color="black" size="12" font-family="Comic Sans MS"
            >How brains looks during PD?</font
          >
        </center>
      </h3>

      <span
        ><img
          id="im"

src="https://ichef.bbci.co.uk/news/976/cpsprodpb/16161/production/_107456409_parkinsons.jpg"
          title="Stage"
      /></span>
      <span
        ><img
          id="im"
          src="https://img.parkinsonsinfoclub.com/wp-content/uploads/back-conditions-neck-conditions-london-back-pain-clinic-scaled.jpeg"
          title="Stage"
      /></span>
      <br /><br />
    </div>
  </body>
</html>\
```

Base Page Html

```
<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
    <meta http-equiv="X-UA-Compatible" content="ie=edge" />
    <title>Predict</title>
    <link
      href="https://cdn.bootcss.com/bootstrap/4.0.0/css/bootstrap.min.css"
      rel="stylesheet"
    />
    <script src="https://cdn.bootcss.com/popper.js/1.12.9/umd/popper.min.js"></script>
```

```
<script src="https://cdn.bootcss.com/jquery/3.3.1/jquery.min.js"></script>
<script src="https://cdn.bootcss.com/bootstrap/4.0.0/js/bootstrap.min.js"></script>
<link
  href="{{ url_for('static', filename='css/main.css') }}"
  rel="stylesheet"
/>
<style>
  body {
    background-image: url("https://img.freepik.com/free-vector/clean-medical-patterned-background-vector_53876-140867.jpg?w=1060&t=st=1667911964~exp=1667912564~hmac=4298568f384f42cfc60423d63ac6a8c806e4fe025c1bed2f32ae68b3f15b2139");
    background-position: center;
    background-repeat: no-repeat;
    background-size: cover;
    height: 100%;
    width: 100%;
  }
  h1 {
    font-size: 40px;
    text-align: center;
    color: black;
    font-style: italic;
    font-weight: bolder;
  }
  h2 {
    font-size: 35px;
    text-align: center;
    color: black;
    font-style: italic;
    font-weight: bolder;
  }
  h5 {
    font-size: 25px;
    text-align: center;
    color: black;
    font-weight: bolder;
  }

  a {
    color: grey;
    float: right;
    text-decoration: none;
    font-style: normal;
```

```
    padding-right: 20px;
   }

   a:hover {
    background-color: black;
    color: white;
    font-size: 30px;
    padding-left: 10px;
    border-radius: 5px;
   }

   ul {
    align-items: center;
    display: flex;
    list-style-type: none;
    width: 100%;
    gap: 3rem;
    justify-content: center;
    font-size: 2rem;
    position: fixed;
    top: 0;
    margin: 0;
    padding: 1rem;
    background-color: white;
   }

   li {
    cursor: pointer;
   }
   li a {
    text-decoration: none;
    color: inherit;
   }
   li.active {
    font-weight: bold;
    color: orangered;
   }
  </style>
</head>
<body>
  <nav>
   <ul>
    <li class="active"><a href="/home">Home</a></li>
    <li class="active"><a href="/upload">Predict-Results</a></li>
```

```
    </ul>
   </nav>
   <br />
   <h1><b>Prevention is better than cure!</b></h1>
   <br />
   <h2>
    <center>
     Diagnosis is not the end, but the beginning of practice.
    </center>
   </h2>
   <br />
   <h2><center>Detect the disease and take measures wisely</center></h2>
   <br />
   <h5>
    NOTE: Upload an spiral or wave page drawn by the patient/user in a white
    sheet
   </h5>
   <div class="container">
    <center>
     <div id="content" style="margin-top: 2em">
      {% block content %}{% endblock %}
     </div>
    </center>
   </div>
  </body>

  <footer>
   <script
    src="{{ url_for('static', filename='js/main.js') }}"
    type="text/javascript"
   ></script>
  </footer>
</html>
```

Prediction Page Html

```
{% extends "base.html" %} {% block content %}

<div>
   <form id="upload-file" method="post" enctype="multipart/form-data">
      <center>
         <label for="imageUpload" class="upload-label">
            Choose...
         </label>
```

```
        <input type="file" name="file" id="imageUpload" accept=".png, .jpg, .jpeg">
      </center>
    </form>

   <center> <div class="image-section" style="display:none;">
      <div class="img-preview">
        <div id="imagePreview">
        </div></center>
      </div>
      <center>
        <div>
          <button type="button" class="btn btn-primary btn-lg " id="btn-predict">Predict!</button>
        </div>
      </center>
    </div>

    <div class="loader" style="display:none;"></div>

    <h3 id="result">
      <span> </span>
    </h3>

</div>

{% endblock %}
```

Home Page Css

```
.img-preview {
    width: 256px;
    height: 256px;
    position: relative;
    border: 5px solid #F8F8F8;
    box-shadow: 0px 2px 4px 0px rgba(0, 0, 0, 0.1);
    margin-top: 1em;
    margin-bottom: 1em;
}

.img-preview>div {
    width: 100%;
    height: 100%;
    background-size: 256px 256px;
    background-repeat: no-repeat;
    background-position: center;
```

```
}

input[type="file"] {
    display: none;
}

.upload-label {
    display: inline-block;
    padding: 12px 30px;
    background: #fe2727;
    color: #fff;
    font-size: 1em;
    transition: all .4s;
    cursor: pointer;
}

.upload-label:hover {
    background: #34495E;
    color: #39D2B4;
}

.loader {
    border: 8px solid #f3f3f3;
    /* Light grey */
    border-top: 8px solid #3498db;
    /* Blue */
    border-radius: 50%;
    width: 50px;
    height: 50px;
    animation: spin 1s linear infinite;
}

@keyframes spin {
    0% {
        transform: rotate(0deg);
    }
    100% {
        transform: rotate(360deg);
    }
}
```

Home Page JS

```
$(document).ready(function() {
```

```
// Init
$('.image-section').hide();
$('.loader').hide();
$('#result').hide();

// Upload Preview
function readURL(input) {
    if (input.files && input.files[0]) {
        var reader = new FileReader();
        reader.onload = function(e) {
            $('#imagePreview').css('background-image', 'url(' + e.target.result + ')');
            $('#imagePreview').hide();
            $('#imagePreview').fadeIn(650);
        };
        reader.readAsDataURL(input.files[0]);
    }
}
$("#imageUpload").change(function() {
    $('.image-section').show();
    $('#btn-predict').show();
    $('#result').text('');
    $('#result').hide();
    readURL(this);
});

// Predict
$('#btn-predict').click(function() {
    var form_data = new FormData($('#upload-file')[0]);

    // Show loading animation
    $(this).hide();
    $('.loader').show();

    // Make prediction by calling api /predict
    $.ajax({
        type: 'POST',
        url: '/predict',
        data: form_data,
        contentType: false,
        cache: false,
        processData: false,
        async: true,
        success: function(data) {
            // Get and display the result
```

```
            $('.loader').hide();
            $('#result').fadeIn(600);
            $('#result').text('Prediction : ' + data);
            console.log('Success!');
        },
    });
  });

});
```

GITHUB

https://github.com/IBM-EPBL/IBM-Project-21125-1659773682

PROJECT DEMO LINK

https://drive.google.com/file/d/1Itoc0fK8vPaYUEtDLPlK3kGhI3RvxRU3/view?usp=drivesdk