19L039 – PROFESSIONAL READINESS FOR INNOVATION EMPLOYABILITY AND ENTREPRENEURSHIP

TOPIC: DETECTING PARKINSONS DISEASE USING MACHINE LEARNING

TEAM ID: PNT2022TMID12773

Lolla Akshatha Devi (Roll No.:19L123)

Samritha D (Roll No.:19L129)

Saranya T (Roll No.:19L131)

Sundereswaran R (Roll No.:19L140)

Dissertation submitted in partial fulfillment of the requirements for the degree of

BACHELOR OF ENGINEERING

Branch: ELECTRONICS AND COMMUNICATION ENGINEERING

of Anna University

November 2022

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING

PSG COLLEGE OF TECHNOLOGY

(Autonomous Institution)

COIMBATORE – 641 004

CONTENTS

CHAPTER

1. INTRODUCTION

1.1. Need for the project

1.2. Purpose

2. LITERATURE REVIEW

2.1 Existing Problem

2.2 References

2.3 Problem Statement Definition

3. IDEATION AND PROPOSED SOLUTION

3.1. Empathy Map Canvas

3.2. Ideation and Brainstorming

3.3. Proposed Solution

3.4. Proposed Solution Fit

4. FUNCTIONAL ANALYSES

4.1. Functional Requirements

4.2. Non-Functional Requirements

5. PROJECT DESIGN

5.1. Dataflow Diagram

5.2. Solution Architecture

5.3. Technical Architecture

5.4. User Stories

6. PROJECT PLANNING AND ESTIMATION

6.1 Sprint Planning and Estimation

6.2 Sprint Delivery Schedule

7. CODING AND SOLUTIONING

8. TESTING

8.1 Test Cases

8.2 User Acceptance Testing

9. RESULTS

10. ADVANTAGES AND DISADVANTAGES

11. CONCLUSION

12. FUTURE SCOPE

APPENDIX

CHAPTER 1
INTRODUCTION

Parkinson's disease is a neurodegenerative disorder that affects predominately the dopamine-producing ("dopaminergic") neurons in a specific area of the brain called *substantia nigra*. It mostly occurs in older people with generalized slowing of movements and at least one other symptom of resting tremor or rigidity. This disease may be genetic, or caused by loss of dopaminergic neurons in the *substantia nigra* and the presence of Lewy bodies. For early detection of the Parkinson's disease, Machine Learning techniques can be applied, where the disease detection is modelled as a Classification problem.

1.1 Project overview

An early stage of Parkinson's disease causes tremors and muscle rigidity, making it difficult to draw smooth spirals and waves. The drawing speed and pen pressure are usually slower in Parkinson's patients. Rather than measuring speed and pressure of a pen on paper drawings can be used to detect the Parkinson's disease. The project focuses on quantifying the visual appearance of these drawings and training a machine learning model to classify them. A Histogram of Oriented Gradients (HOG) image descriptor and a Random Forest Classifier are used.

1.2 Purpose

While Parkinson's disease affects around 10 million people worldwide, there is no permanent cure for the disease. However, early detection and diagnosis can make pharmacological treatments effective and slower the disease process. Scientists are exploring ways to identify biomarkers for PD that can lead to earlier diagnosis and more tailored treatments. These prove the necessity of Machine Learning techniques that enable quicker and more accurate diagnosis.

CHAPTER 2
LITERATURE SURVEY

This chapter discusses the existing problems in the Parkinson's disease (PD) detection and the use of Machine learning techniques for its detection. It presents an overview of other existing works that aim at the early detection of the disease.

2.1 Existing problem

Tremor is the first symptom of Parkinson's disease in a person. However, tremor can be the first symptom of a variety of other neurological disorders, including multiple sclerosis and stroke. Bradykinesia, rigidity, and postural instability are progressive symptoms that appear later in the Parkinson's disease's progression, and at this stage, approximately 50-80% of the dopaminergic neurons are degenerated [1]. Initial diagnosis of this disease is complex due to the fact that tremor is the only symptom. Usually, history of family is analysed and clinical examination is performed to diagnose the PD. DAT-SPECT and MRI scans (Imaging based scans) are performed to rule out other neurological disorders, if clinical examinations are not sufficient to confirm that a person is suffered from Parkinson's disease.

Paper [3] presents a technique for Parkinson disease using deflection in the voice. The data collected from person with Parkinson's disease and without Parkinson's disease are used to train the machine learning algorithm. An XG Boost algorithm is used as the machine learning algorithm. Navie Bayes is used for predicting the dataset and accuracy of the proposed model is done through a decision tree. It is to be noted that speech abnormality occurs in the second stage of the Parkinson's disease.

Paper [4] discusses the Static Spiral Test (SST), Dynamic Spiral Test (DST), Stability around Centre Point (STCP) tasks performed on a digital tablet for the diagnosis. Similar tests studied in paper [5] are said to be scientific tests for motor sensitivity.

2.2 References

[1] George DeMaagd and Ashok Philip: Parkinson's Disease and Its Management: 2015 Aug: 40(8): 504-510, 532. [PMC]

[2] Saman Zafar, Sridhara S.Yaddanapudi: Parkinson's Disease, 2021 Aug; Stat Pearls. [NIB]

[3] Gomathy, C K. (2021). THE PARKINSON'S DISEASE DETECTION USING MACHINE LEARNING TECHNIQUES.

[4] Impedovo D, Pirlo G and Vessio G, “Dynamic Handwriting Analysis for Supporting Earlier Parkinson’s Disease Diagnosis” in Information, 2018, 9(10):247, doi: 10.3390/INFO9100247.

[5] Megha Kamble, Prashant Shrivastava and Megha Jain, “Digitized spiral drawing classification for Parkinson's disease diagnosis” in Measurement: Sensors, vol. 16, 100047, 2021, doi: 10.1016/J.MEASEN.2021.100047.

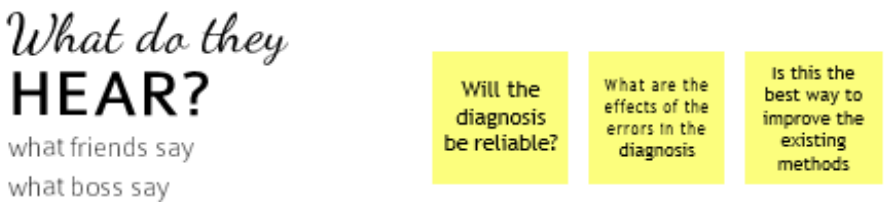
2.3 Problem Statement Definition

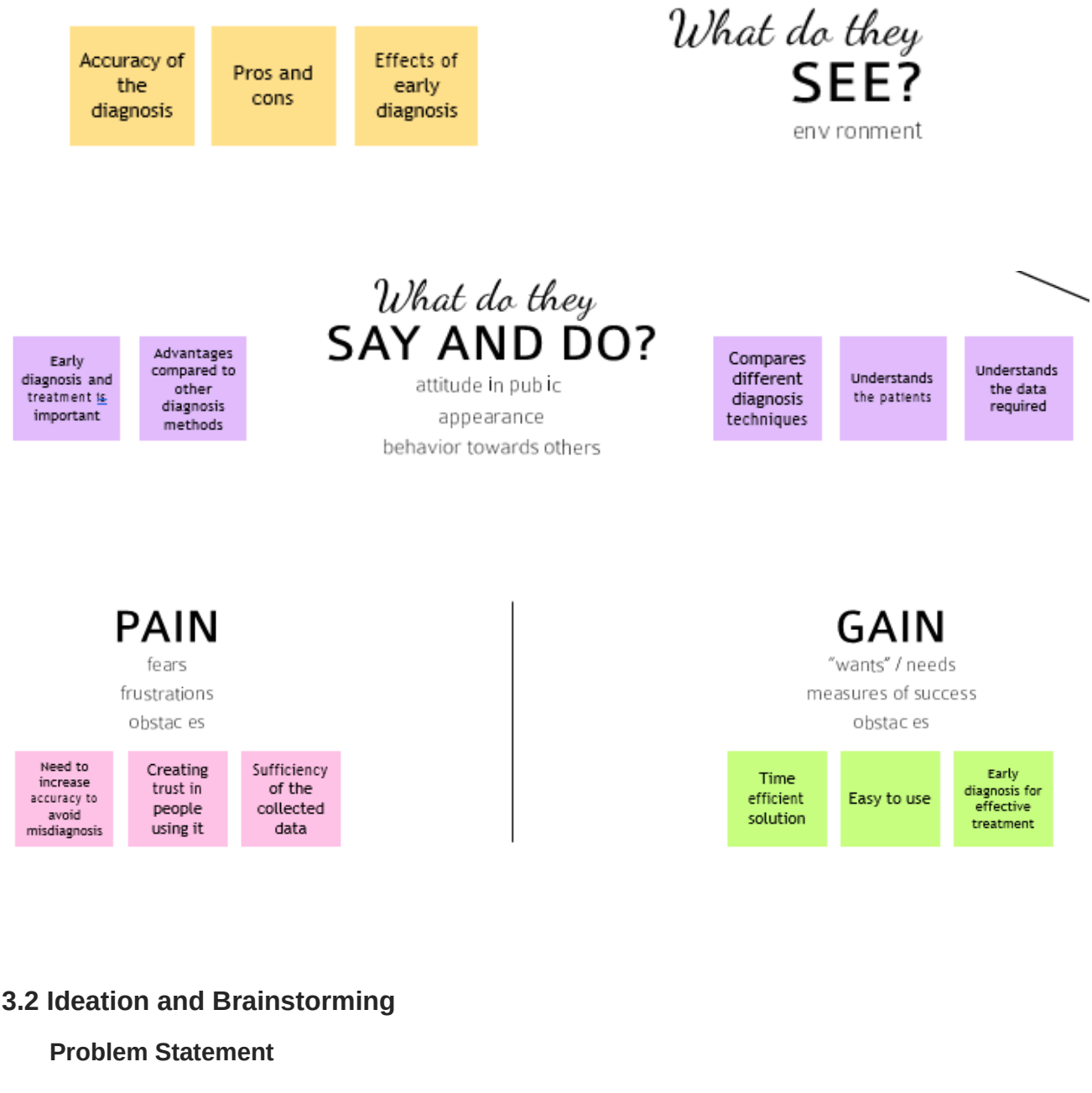
The project's goal is to present a solution for detecting Parkinson's disease using hand drawn images. Spiral and wave drawings by the person are used for this purpose. The Machine Learning model classifies whether a person is healthy or has Parkinson's disease based on the spiral or wave drawing done by the person. The visual appearance of the spiral or wave drawn by an affected person appears distorted and significantly differs from a perfect spiral drawing because of slow motor movements and decreased coordination between hand and brain. This is exploited to train the Machine Learning Classifier.

CHAPTER 3

IDEATION AND PROPOSED SOLUTION

3.1 Empathy Map Canvas





3.2 Ideation and Brainstorming

Problem Statement

PROBLEM

How might we detect Parkinson's disease in the elderly during the early stage?

Brainstorm

Akshatha

- Recording speed of drawing using digital tablet
- MRI scans to check for irregularities in the brain
- Classifying images of handwriting tasksusing machine learning
- Continuous monitoring for tremors in specific areas of the body, when person is at rest

Samritha

- Manual inspection of handwriting tasks
- Continuous monitoring of gait in elderly using accelerometer sensors
- Specially designed smelltests for detecting a dulled sense ofsmell
- Family medical history-based prediction of likelihood of getting Parkinson's

Saranya

- Questionnaire at a clinic 
- Continuous monitoring of gait in elderly via caretakers
- Government organization to conduct regular health check-ups on elderly citizens
- Make the virtual assistant (Alexa, Siri etc.) detect abnormalities in speech

Sundereswaran

- Monitoring sleep cycles to detect abnormalities
- Go for regular medical check-up for gait, speech
- Exercise equipment that can capture gait and stance of the elderly
- Online questionnaire based self-diagnosis that can be taken athome

Grouping Ideas:

Clinical one-time tests - effective only when done regularly

- Questionnaire at a clinic
- MRI scans to check for irregularities in the brain
- Medical check-up for gait, speech
- Manual inspection of handwriting tasks
- Specially designed smelltests for detecting a dulled sense ofsmell

Monitoring in home with specific tests

- Government organization to conduct regular health check-ups on elderly citizens
- Online questionnaire based self-diagnosis regularly taken up
- Continuous monitoring of gait in elderly via caretakers

Non-clinical one-time test - effective only when combined with other tests

Family medical history-based prediction of likelihood of getting Parkinson's

Monitoring integrated into day-to-day activities

Exercise equipment that can capture gait and stance of the elderly

Using handwriting tasks

Recording speed ofdrawing using digital tablet and comparing to healthy samples using machine learning

Manual inspection of handwriting tasks

Classifying images of handwriting tasksusing machine learning

Monitoring in home using sensor technology

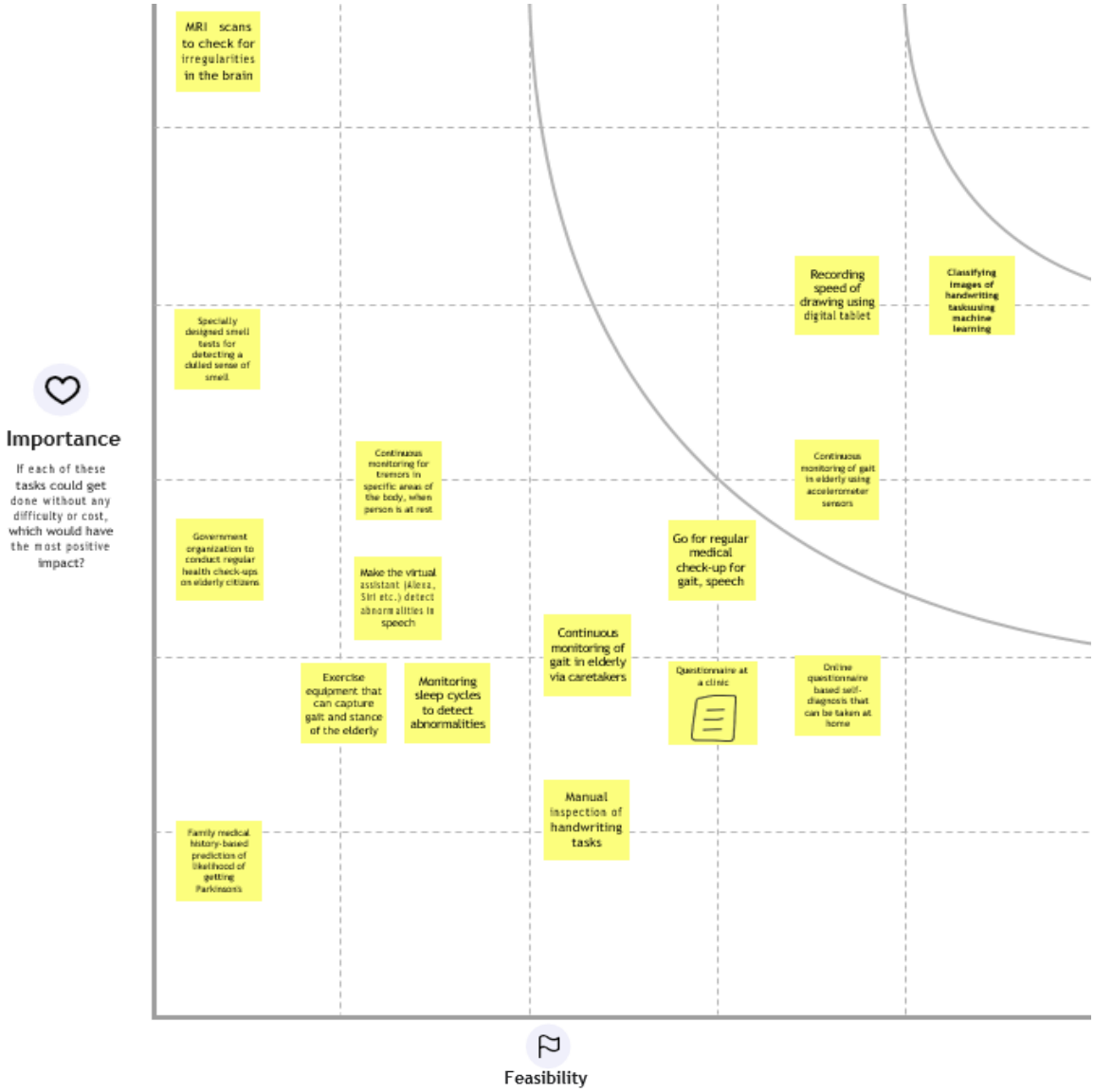
Continuous monitoring for tremors in specific areas of the body, when person is at rest

Make the virtual assistant (Alexa, Siri etch) detect abnormalities in speech

Continuous monitoring of gait in elderly using accelerometer sensors

Monitoring sleep cycles to detect abnormalities

Prioritizing



3.3 Proposed Solution

S.No.	Parameter	Description
1	Problem Statement (Problem to besolved)	Detect the probability of occurrence of Parkinson's disease in patients when no observable symptoms are present.

2	Idea / Solution description	Make the patient use pencil and paper to draw spirals and waves. Take a photo of each spiral and wave, and upload it to the server via an app, by connecting to the internet. The app will display the probability of Parkinson's disease by processing the images using Machine Learning.
3	Novelty / Uniqueness	Compared to traditional methods of diagnosis of Parkinson's, this method can be applied easily at home as a pre-diagnosis; since ML model is tuned to minimize false positives, unnecessary visits to the hospital can be avoided. By performing this simple test regularly (every three months or so), onset of Parkinson's can be detected in an early stage, so that physiotherapy can be organised to help patients cope with the disease.
4	Social Impact / Customer Satisfaction	Will be particularly useful in old age homes and medical centres for the elderly, wherein a community of elderly people can be monitored without regular hospital visits. This app can be integrated with other health monitoring systems to provide a holistic regular check-up without frequent visits to the hospital.
5	Business Model (Revenue Model)	Since the product consists mainly of an app making use of IBM cloud services, a one-time fee can be charged for lifetime access to the app. Advertisement-based and subscription-based revenue models were also considered, but were deemed unsuitable due to the fact that the app will not be used often, on a monthly basis at most. Integration with a more holistic health-

		monitoring system can prove to be more profiting in terms of revenue generated.
6	Scalability of the Solution	The solution is scalable since the cloud resource requirement will not increase drastically with increasing users. Previously uploaded images by users can be deleted, since the ML model is not dynamically trained and works on a single image at a time. Computational resources required will also not increase too much, since app will be used very infrequently, and computation can be scheduled easily.

3.4 Proposed Solution Fit

CHAPTER 4
FUNCTIONAL ANALYSES

4.1 Functional Requirements

Following are the functional requirements of the proposed solution.

FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	User/ Patient Registration	Registration through Form Registration through Gmail Registration through Hospital membership Registration through Mobile number
FR-2	User Confirmation	Confirmation via EmailConfirmation via OTP
FR-3	Authentication	Authentication through Password and Patient ID (ifregistered with any Hospital)
FR-4	External Interfaces	Web application/ Android mobile application for auser-friendly GUI.
FR-5	Medical requirements	The model only provides a prediction of the diagnosis based on the uploaded images; however, a further medical examination would be necessary and consulting a physician is advised in case ofpositive diagnosis.
FR-6	User preferences	The user can prefer to use the prediction model multiple times with different input data. They can locate relevant websites and articles to verify the prediction's accuracy. They can choose not to rely on the prediction and further consult a physician to confirm the diagnosis provided by the model and opt for further treatment, if any.

4.2 Non-Functional requirements

Following are the non-functional requirements of the proposed solution.

FR No.	Non-Functional Requirement	Description
NFR-1	Usability	The user can easily interact with the model using the Simple User Interface to upload their drawing image and get the diagnosis prediction.
NFR-2	Reliability	The Machine Learning model would have a good accuracy to increase the reliability of the solution. Underfitting and overfitting of the model would be prevented.
NFR-3	Performance	The application developed would require minimum processing time and faster response, thus providing a satisfactory user experience.
NFR-4	Availability	The application would be easily available to all sectors of the population and can be accessed from anywhere.
NFR-5	Scalability	As many users use the application, the collected data can be used to further train the model. The model would be scalable to train with a larger dataset and provide more accurate prediction.
NFR-6	Security	The encrypted user details and data collected would be stored in a secure database.

CHAPTER 5
PROJECT DESIGN

5.1 Dataflow Diagram

The Data Flow Diagram (DFD) gives a representation of the information flows within the proposed system. It depicts the amount of the system requirement graphically. It shows how data enters and leaves the Parkinson's Disease detection system, what changes the information, and where the data are stored.

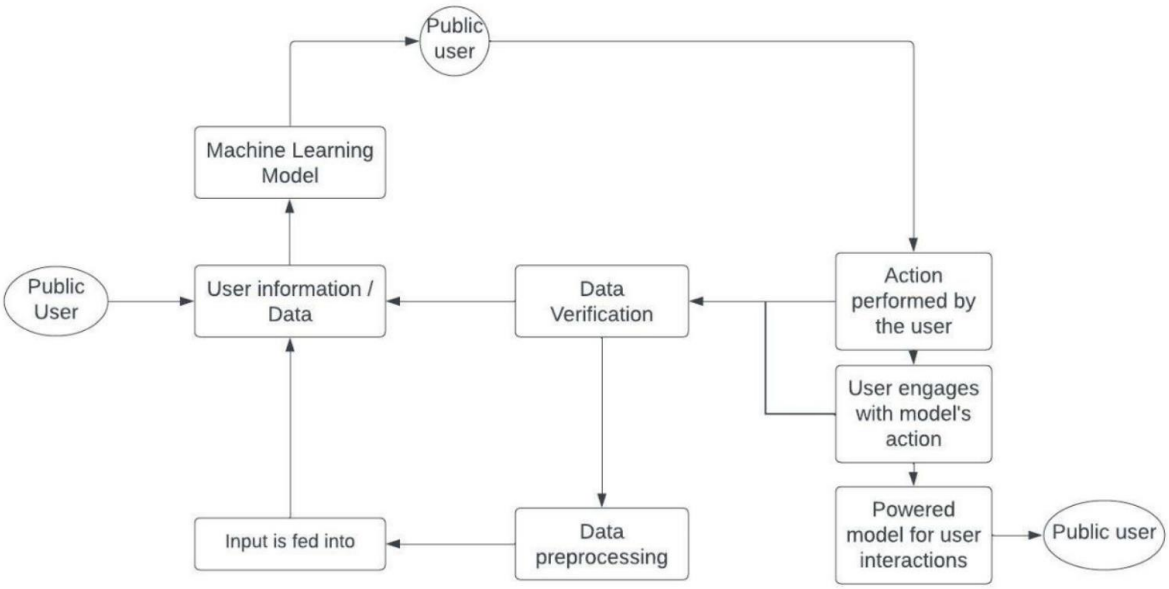
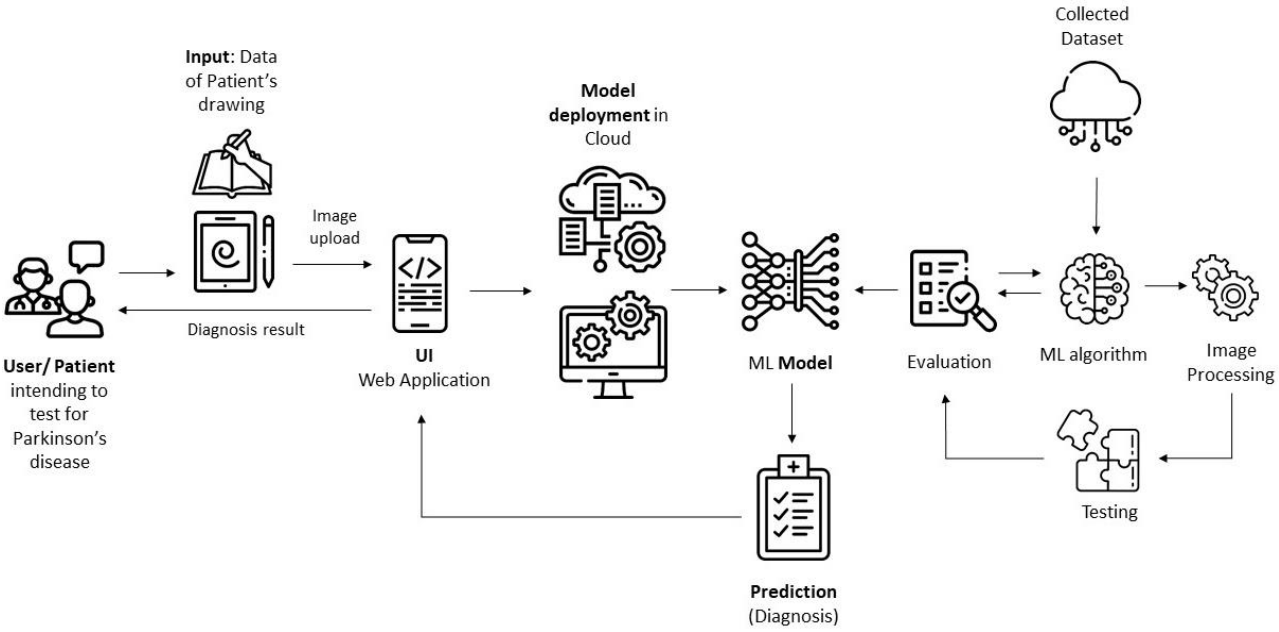


Fig. 5.1 Dataflow diagram of detecting Parkinson's disease

5.2 Solution Architecture



5.3 Technical Architecture

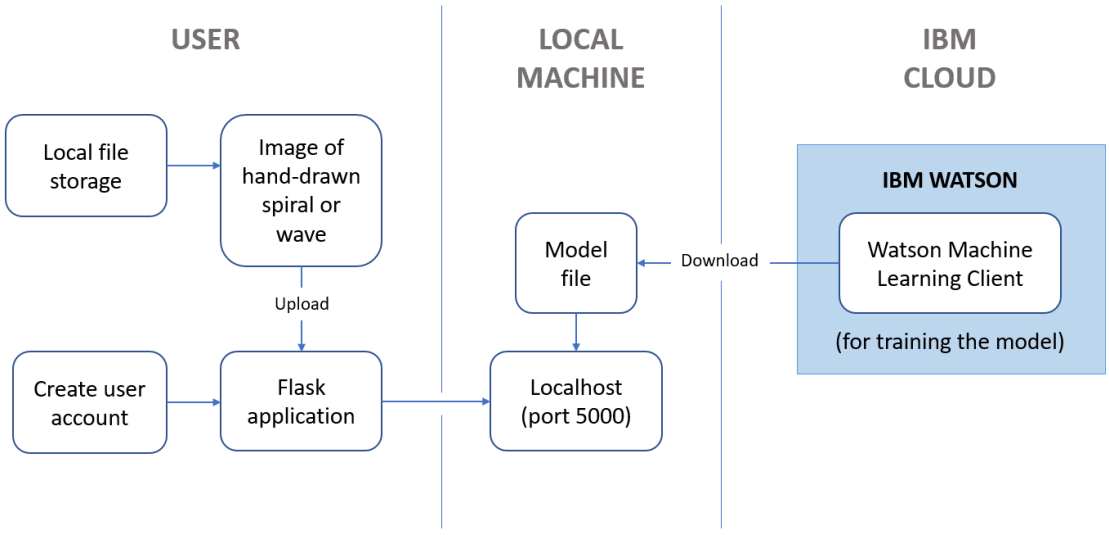


Table 1 Components & Technologies

S. No.	Component	Description	Technology
1	User Interface	User interacts with a Web UI	HTML, Flask framework
2	Application Logic - 1	Logic for a process in the application	Python (Flask)
3	Application Logic - 2	To execute the machine learning model on cloud	IBM Watson Machine Learning Client
4	Cloud Database	Database Service used for storing user data,images uploaded	MySQL
5	File Storage	To store the image file of the spiral / wave drawing	Local File system
6	Machine Learning Model	To detect whether the image is drawn by a person with Parkinson’s or not	Random Forest Classifier
7	Infrastructure	Application Deployment on locally	Flask frame work

Table 2 Application Characteristics

S. No.	Characteristics	Description	Technology
1	Open-Source Frameworks	List the open-source frameworks used	Flask, Pickle, Sklearn, OpenCV
2	Scalable Architecture	Cloud native applications are hyper-scalable	IBM Watson
3	Availability	Using containerized workloads, scales clusters	Running on local machine
4	Performance	Serve up to 100 requests per sec	Flask frame work

5.3 User Stories

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
Customer (Mobile user)	Account creation	USN-1	As a user, I can connect my google into the application.	I can access my account / dashboard	High	Sprint-1
Input data	Addition of data	USN-2	As a user, I can make the data to be fed as an input into the application to classify the true and false samples of data.	I can cross verify the data or information entered inthe initial step.	High	Sprint-1
Data validation	Accuracy checking	USN-3	As a user, I can check the ability and precision of the machine learning model for obtaining thenecessary or required information or data.	I can login to my account and check the model's capability.	Medium	Sprint-2
Classification	Data classification	USN-4	As a user, I can see the actual data or information.	I can verify my data with the actual data.	Medium	Sprint-2
App work	Work flow	USN-5	As a user, I can examine the working functionof the application model.	I can view how the application works and responds to the actions being imposed.	High	Sprint-2

Image classification	Checking the disease	USN-6	As a user, I can be able to verify with the application that the image is identified with theoriginal disease with the help of trained and tested data samples.	I can confirm that the data gives the accurate or precise result.	Low	Spirit-3
User interaction	Powered machine learning model	USN-7	As a user, I can interact with machine learningmodel till the application processes the result in the correct time.	I can view the results from the interaction with powered model.	Low	Spirit-3
Medical assistance	Medical suggestions	USN-8	As a user, I can get the medical advises and suggestions to boost the action of curing theParkinson's disease.	I can get enough assistance by getting suggestions for curing the disease.	High	Spirit-3
Data extraction	Obtaining the data	USN-9	As a user, I can retrieve the result from the application for storage of data for future usageby medical research users.	I can download the result in the form of data to showit as a proof to healthcare team.	Medium	Spirit-4

CHAPTER 6

PROJECT PLANNING AND ESTIMATION

6.1 Sprint Planning and Estimation

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-1	Image pre-processing	USN-6	Pre-processing the spiral and wave images and extracting the features	3	High	Lolla Akshatha Devi Samritha D Saranya T Sundereswaran R
Sprint-1	Model Building	USN-6	Using the dataset building a model to classify the image	3	High	Lolla Akshatha Devi Samritha D Saranya T Sundereswaran R
Sprint-1	Image processing localization	USN-9	The uploaded image is pre-processed and fed into trained model.	3	High	Lolla Akshatha Devi Samritha D Saranya T Sundereswaran R
Sprint-2	Web application	USN-2	Creating the web application using Flask web frame work	2	Medium	Lolla Akshatha Devi Samritha D Saranya T Sundereswaran R
Sprint-2	Dashboard	USN-6	As a user, I can upload my images and get my details.	3	High	Lolla Akshatha Devi Samritha D Saranya T Sundereswaran R
Sprint-3	Training the model on IBM Watson	USN-6	Using the dataset building a model to classify the image in IBM watson	2	Medium	Lolla Akshatha Devi Samritha D Saranya T Sundereswaran R
Sprint-3	Classification and prediction	USN-9	The model classifies and predicts the type of disease.	3	High	Lolla Akshatha Devi Samritha D Saranya T Sundereswaran R

Sprint-4	Login	USN-5	As a user, I can log into the application by enteringemail & password	3	High	Lolla Akshatha Devi Samritha D Saranya T Sundereswaran R
Sprint-4	Report generation	USN-10	Based on the prediction of Parkinson's disease, the health care is generated to provide the feedback.	2	Medium	Lolla Akshatha Devi Samritha D Saranya T Sundereswaran R

6.2 Sprint Delivery Schedule

Sprint	Total Story Point	Duration	SprintStart Date	Sprint End Date (Planned)	Story Points Completed (as on Planned End Date)	Sprint Release Date (Actual)
Sprint-1	20	6 Days	24 Oct 2022	29 Oct 2022	20	29 Oct 2022
Sprint-2	20	6 Days	31 Oct 2022	05 Nov 2022	20	03 Nov 2022
Sprint-3	20	6 Days	07 Nov 2022	12 Nov 2022	20	08 Nov 2022
Sprint-4	20	6 Days	14 Nov 2022	19 Nov 2022	20	13 Nov 2022

Sprint	Total Story Point	Duration	SprintStart Date	Story Points Completed (as on Planned End Date)	Sprint Delivered Date
Sprint-1	20	14 Days	24 Oct 2022	20	06 Nov 2022
Sprint-2	20	7 Days	31 Oct 2022	20	06 Nov 2022
Sprint-3	20	1 Days	07 Nov 2022	20	07 Nov 2022
Sprint-4	20	4 Days	14 Nov 2022	20	17 Nov 2022

CHAPTER 7

CODING & SOLUTIONING

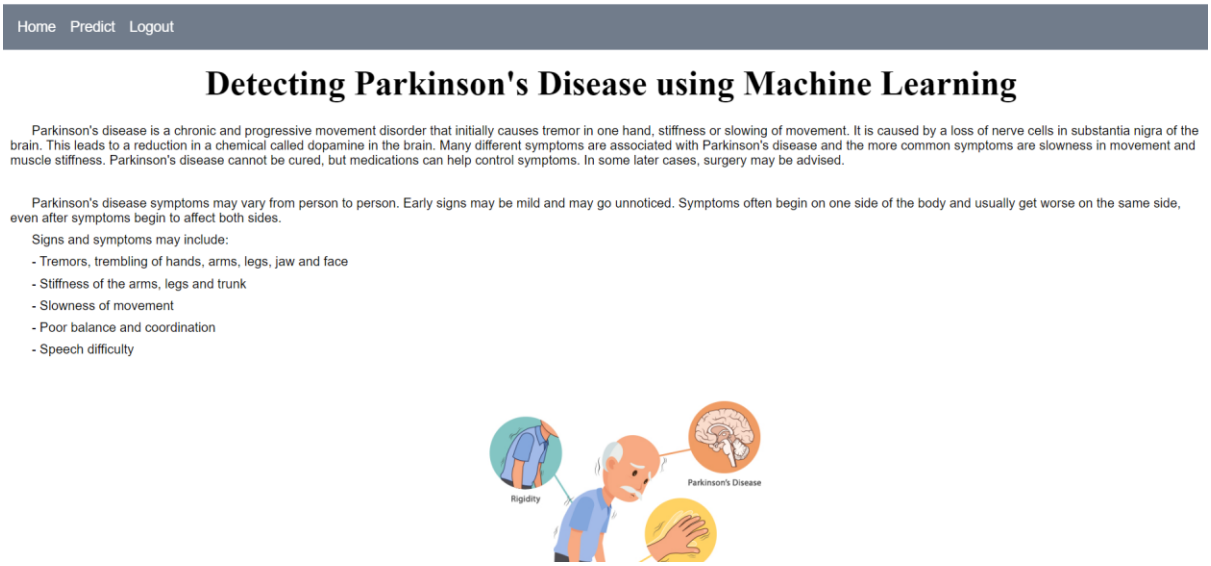
7.1 HOG feature descriptor

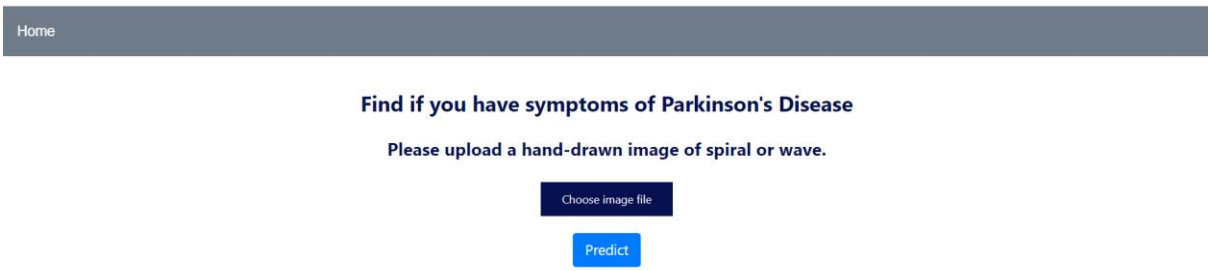
The Histogram of Oriented Gradients (HOG) feature descriptor is used here. The approach counts the number of times a gradient orientation appears in the localized portion of an image. HOG technique focuses on an object's structure or shape. In HOG, the local object appearance and shape within the image can be described by the distribution of intensity gradients or edge directions. Here the image is divided into small connected regions called as cells. For the pixel within each cell, a histogram of gradient directions is compiled. This makes

```
def quantify_image(image):
    features = feature.hog(image,
                           orientations=9,
                           pixels_per_cell=(5,5),
                           cells_per_block=(2,2),
                           transform_sqrt=True,
                           block_norm="L1")
    return features
```

7.2 Webpage

The webpage has a simple User Interface making it comfortable and easy-to-use. It is designed keeping in mind the various sections of people who might want to use the service.





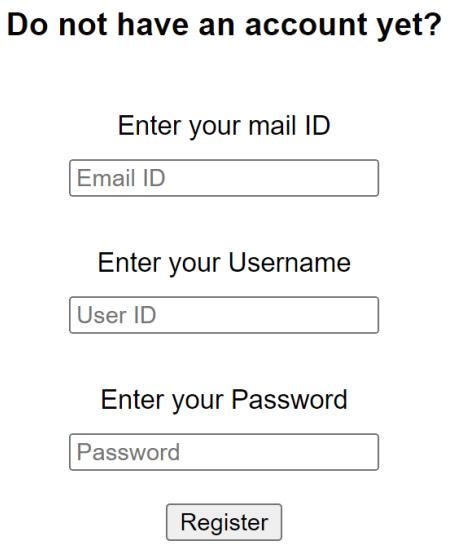
The image of the drawing can be easily uploaded in the **Predict** page. .png, .jpg, .jpeg image formats are supported.

7.3 User registration

Code snippet:

```
@app.route('/form_reg',methods=['POST','GET'])
def reg():
    name2=request.form['userid']
    pwd1=request.form['pwd']
    if name2 in database:
        return render_template('index.html',info='Username Already Taken!')
    else:
        database.add(name2,pwd1)
        return render_template("index.html")
```

Feature as displayed in the webpage:



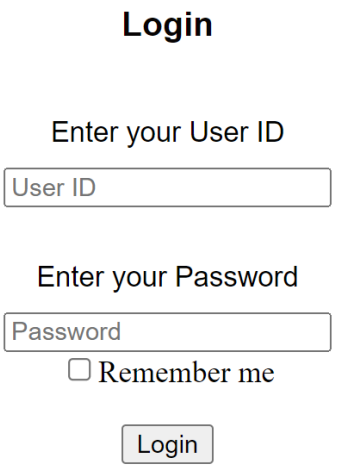
The user can register with their unique Email ID, username and a password of their choice.

7.5 User login

Code snippet:

```
@app.route('/form_login',methods=['POST','GET'])
def login():
    name1=request.form['userid']
    pwd=request.form['pwd']
    if name1 not in database:
        return render_template('index.html',info='Invalid User!')
    else:
        if database[name1]!=pwd:
            return render_template('index.html',info='Invalid Password!')
        else:
            return render_template('home.html',name=name1)
```

Feature as displayed in the webpage:



The user can login with registered Email ID and the corresponding password set at the time of registration. *Remember me* feature allows the browser to store the login information for easy future login.

7.6 Deployment in IBM cloud

Getting the unique API key

```
from ibm_watson_machine_learning import APIClient
wml_credentials = {
                   "url" : "https://us-south.ml.cloud.ibm.com",
                   "apikey" : "9_NV6WjMkXK1YyDswn6CifrfHVpv6WzJzSDkedXQR9bH"
                   }
```

Creating a deployment space

```
space_id = guid_from_space_name(client,'ParkinsonsDetectionDep')
print('Space UID = ' + space_uid)
```

```
Space UID = 4572c734-84dd-4550-a50c-e2178b21aa59
```

Deployment

```
DEPLOYMENT_NAME="ParkinsonsDetectionDep"
DEMO_MODEL=model
```

```
model_details
model_id=client.repository.get_model_id(model_details)
dep_props={
 client.deployments.ConfigurationMetaNames.NAME:DEPLOYMENT_NAME,
 client.deployments.ConfigurationMetaNames.ONLINE:{}
}
```

```
deployment=client.deployments.create(artifact_uid=model_id,meta_props=dep_props)
```

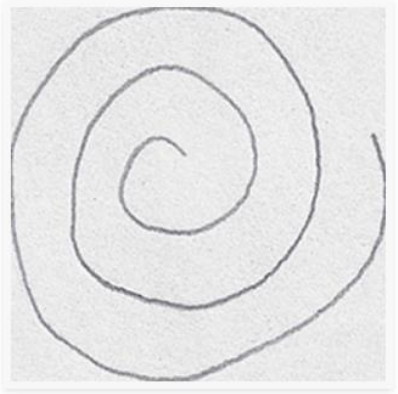
```
#######################################################################################

Synchronous deployment creation for uid: '8c4254c5-1a33-40b8-b432-5e75ea7497c2' started

#######################################################################################


initializing
Note: online_url is deprecated and will be removed in a future release. Use serving_urls instead.

ready


------------------------------------------------------------------------------------------------
Successfully finished deployment creation, deployment_uid='e223c7d8-0cd9-42f2-a9a2-351450574d40'
------------------------------------------------------------------------------------------------
```

CHAPTER 8
TESTING

8.1 Test case report and User Acceptance Testing

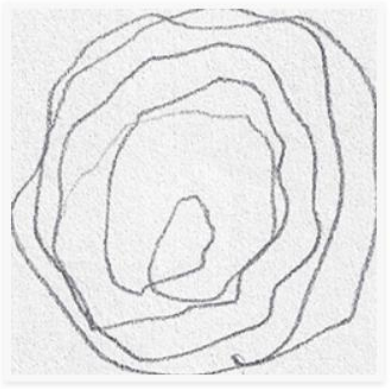
The designed model was tested several times with various input drawings. Few samples of them are given here.

Test case 1: Drawing of a healthy person - Predicted correctly



Prediction : You are healthy !

Test case 2: Drawing of a person with Parkinson's disease – Predicted correctly



Prediction : You may have Parkinson's disease.

Test case ID	Feature Type	Component	Test Scenario	Steps To Execute	Test Data	Expected Result	Actual Result	Status
Login Page_ TC_0 01	Functional	Login / Register Page	Verify user is able to see the Login/Register page on opening the website	1. Run the app using Command Prompt by navigating to the folder in which the app is stored, and executing the command: python app.py 2. Enter URL (localhost:5000) in a browser and go 3. Verify Login/Register page is displayed or not	https://localhost:5000	Login/Register page should display	Working as expected	Pass
Login Page_ TC_0 02	UI	Login / Register Page	Verify the UI elements in Login/Signup page	1. Follow step 1, 2 of first test case 2. Verify Login/Register page with below UI elements: a. email text box b. userID test box c. password text box d. Register button e. Login button	https://localhost:5000	Application should show below UI elements: a. email text box b. userID test box c. password text box d. Register button e. Login button	Working as expected	Pass
Login Page_ TC_0 03	Functional	Login / Register Page	Verify user is able to register with valid credentials	1. Follow step 1, 2 of first test case 2. Enter valid email in Email text box under Register section 3. Enter a username in the UserID textbox under Register section 4. Enter a password in password text box using Register section 5. Click on register button	Email: akd1lolla@gmail.com UserID : Akshatha Password: Testing123	User should be prompted to login	Working as expected	Pass

Login Page_ TC_0 04	Functional	Login / Register Page	Verify user is able to register with valid credentials	1. Follow steps 1-5 of the previous test case 2. Enter valid username in UserID text box under Login section 4.Enter valid password in password text box under Login section 5.Click on Login button	UserID : Akshatha Password: Testing123	User should be redirected to Home page of the website	Working as expected	Pass
Login Page_ TC_0 05	Functional	Login / Register Page	Verify user is unable to register with invalid credentials	1. Follow step 1, 2 of first test case 2. Enter invalid email in Email text box under Register section 3. Enter a username in the UserID textbox under Register section 4. Enter a password in password text box using Register section 5. Click on register button	Email: akd1lolla@gmail UserID : Akshatha Password: Testing123	Application should show 'Invalid email' validation message.	Working as expected	Pass
Login Page_ TC_0 06	Functional	Login / Register Page	Verify user is unable to login with invalid credentials	1. Follow steps 1-5 of the previous test case 2. Enter invalid username in UserID text box under Login section 4.Enter valid password in password text box under Login section 5.Click on Login button	UserID : Akshatha123 Password: Testing123	Application should show 'Incorrect User ID' validation message.	Working as expected	Pass

Login Page_ TC_0 06	Functional	Login / Register Page	Verify user is unable to login with invalid credentials	1. Follow steps 1-5 of the previous test case 2. Enter valid username in UserID text box under Login section 4.Enter invalid password in password text box under Login section 5.Click on Login button	UserID : Akshatha Password: Testing345	Application should show 'Incorrect password' validation message.	Working as expected	Pass

Home Page_ TC_0 07	UI	Home page	Verify the UI elements in Login/Signup page	Verify Home page with below UI elements: a. Informative text about Parkinson's disease b. Infographics and images about Parkinson's disease c. Navigation buttons: Home d. Navigation buttons: Predict e. Navigation buttons: Logout		Application should show below UI elements: a. Informative text about Parkinson's disease b. Infographics and images about Parkinson's disease c. Navigation buttons:	Working as expected	Pass
Home Page_ TC_0 08	Functional	Home page	Verify user is able to navigate to Predict page	1. Follow steps to login 2. Click on 'Predict' navigation button on the top of the page		User should be redirected to Predict page of the website	Working as expected	Pass
Home Page_ TC_0 09	Functional	Home page	Verify user is able to Logout	1. Follow steps to login 2. Click on 'Logout' button on the top of the page		User should be logged out and redirected to Login/Register page of the website	Working as expected	Pass

PredictPage_TC_010	UI	Predict page	Verify UI elements in Predict page	Verify Predict page with below UI elements: a. Text instructions b. Upload button c. Predict button d. Navigation buttons: Home		Application should show below UI elements: a. Text instructions b. Upload button c. Navigation buttons: Home d. Navigation buttons: Predict e. Navigation buttons: Logout	Working as expected	Pass
PredictPage_TC_011	Functional	Predict page	Verify user is able to upload an image and preview the uploaded image	1. Follow steps to login 2. Navigate to Predict page 3. Click on 'Upload Image' button. File Explorer will open 4. Choose an image 5. Click on open 6. Wait for the preview of the uploaded image to show on the webpage. Verify if it is the same as the uploaded image	https://drive.google.com/drive/folders/1SMCUjXClLw2zp2wnEUgWbaLzQSj0PKSp?usp=sharing	On opening the File Explorer, only images of format .png, .jpg, and .jpeg show. After uploading, a preview of the image uploaded should load below the 'Upload Image' button	Working as expected	Pass
PredictPage_TC_013	Functional	Predict page	Verify user is able to reupload the image	1. Follow the steps 1-3 under test case 011 2. Click on 'Upload Image' button again	https://drive.google.com/drive/folders/1SMCUjXClLw2zp2wnEUgWbaLzQSj0PKSp?usp=sharing	The new image preview should show on the webpage	Working as expected	Pass
PredictPage_TC_014	Functional	Predict page	Verify user is able to view the predicted result	1. Follow the steps 1-3 under test case 011 2. Click on 'Predict' button	https://drive.google.com/drive/folders/1SMCUjXClLw2zp2wnEUgWbaLzQSj0PKSp?usp=sharing	Application should show "You are healthy !" (or) "You may have Parkinson's disease." depending on the predicted result	Working as expected	Pass

PredictPage_TC_015	Functional	Predict page	Verify user is able to navigate to Home page	1. Follow steps to login 2. Navigate to Predict page 3. Click on 'Home' navigation button		User should be redirected to Home page of the website	Working as expected	Pass

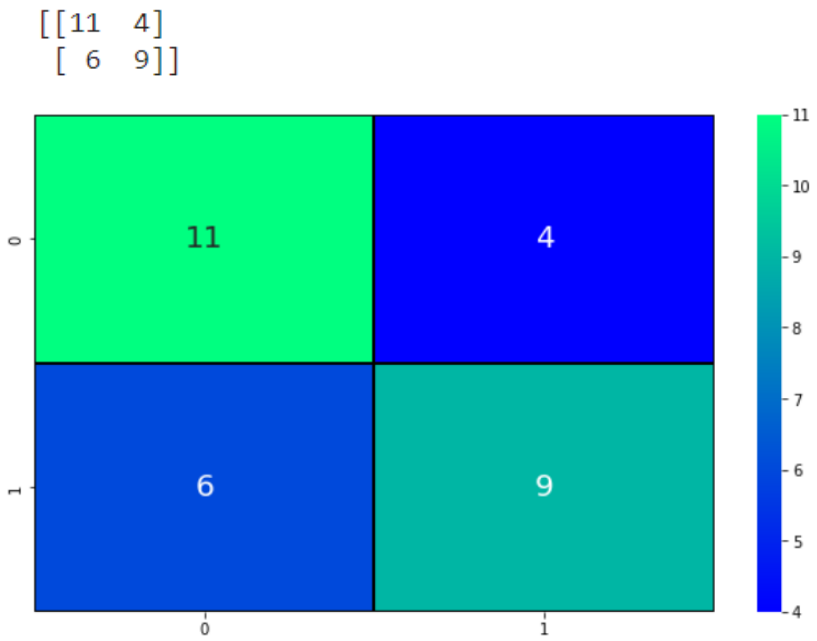
CHAPTER 9
RESULTS

9.1 Performance Metrics

The performance of the Classification model was evaluated using the following metrics. They provide a quantitative measure of the quality of the model.

Confusion matrix:

It is a table with combinations of predicted and actual values. It describes the performance of the classification model for a set of test data for which true values are known. It is the best evaluation metric. When a test sample of 30 images were given to the trained model, the confusion matrix was as follows,



Accuracy:

Accuracy of the Parkinson's disease classifier is found as the percentage of correct predictions for the test data. It is defined as the ratio between the number of correct predictions to the total number of predictions.

$$\text{Accuracy} = \frac{TP+TN}{TP+TN+FP+FN}$$

A higher accuracy of the model does not always indicate a good performance; it can be misleading at times.

Precision:

Precision explains the number of correctly predicted cases actually turned to be positive. It is defined as the ration of number of true positives to the number of predicted positives.

$$\text{Precision} = \frac{\text{TP}}{\text{TP+FP}}$$

Recall:

Recall explains the number of actual positive cases we are able to predict correctly with our model. It is defined as the ratio of number of true positives to the total number of actual positives. It is also called Sensitivity.

$$\text{Recall (Sensitivity)} = \frac{\text{TP}}{\text{TP+FP}}$$

Metrics as measured for the model:

```
Sensitivity for Random Forest 0.6
Specificity for Random Forest 0.7333333333333333
Precision for Random Forest 0.6923076923076923
accuracy for Random Forest 0.6666666666666666
```

Classification report

	precision	recall	f1-score	support
0	0.65	0.73	0.69	15
1	0.69	0.60	0.64	15
accuracy			0.67	30
macro avg	0.67	0.67	0.67	30
weighted avg	0.67	0.67	0.67	30

CHAPTER 10

ADVANTAGES, DISADVANTAGES AND CONCLUSION

The web application developed is easy to use and has a simple, user friendly Interface. This makes it accessible to all groups of people who would want to check the presence of Parkinson's disease. The use of Machine learning makes the first stage diagnosis easier and also reliable. However, further medical examinations are recommended for positive cases. Though the model has a good accuracy, extensive trial and error to tune the hyper-parameters might provide a yet better model. Except for object orientation, this method represents the occurrence of specific gradient orientation in local parts of images for the early detection of disease. It is invariant to geometric and photometric transformations.

The computational speed is delayed while detecting the large-scale images because this method uses the sliding window technique to extract features from every pixel of an image. The proposed machine learning algorithm's accuracy is lower when compared to other current convolutional neural networks and is more sensitive to image rotation.

CHAPTER 11

FUTURE SCOPE

The proposed method can be integrated with the existing voice-based Parkinson's disease detection method to guarantee a person is suffering or not from Parkinson's disease and if a person is suffering, then with this work, it is able to analyse whether if a person is in stage one or stage two. Instead of uploading the hand drawn images, the drawing can be digitalised.

APPENDIX

GITHUB Link: IBM-EPBL/IBM-Project-33629-1660224712: Detecting Parkinson's Disease using Machine Learning (github.com)

Demo Link: https://youtu.be/qd4Jbh-vLCU