PSG COLLEGE OF TECHNOLOGY
(An Autonomous Institution)
COIMBATORE - 641004

PROJECT

Detecting Parkinson's Disease using Machine Learning

DONE BY

TEAM ID: PNT2022TMID13049

ASWATH S	**(19I205)**
GOWTHAM S	**(19I220)**
NISARANI	**(19I235)**
PRAGATHEESWARAN T	**(19I241)**

1. **INTRODUCTION**
    1. Project Overview
    2. Purpose
2. **LITERATURE SURVEY**
    1. Existing problem
    2. References
    3. Problem Statement Definition
3. **IDEATION & PROPOSED SOLUTION**
    1. Empathy Map Canvas
    2. Ideation & Brainstorming
    3. Proposed Solution
    4. Problem Solution fit
4. **REQUIREMENT ANALYSIS**
    1. Functional requirement
    2. Non-Functional requirements
5. **PROJECT DESIGN**
    1. Data Flow Diagrams
    2. Solution & Technical Architecture
    3. User Stories
6. **PROJECT PLANNING & SCHEDULING**
    1. Sprint Planning & Estimation
    2. Sprint Delivery Schedule
    3. Reports from JIRA
7. **CODING & SOLUTIONING (Explain the features added in the project along with code)**
    1. Feature 1
    2. Feature 2
    3. Database Schema (if Applicable)
8. **TESTING**
    1. Test Cases
    2. User Acceptance Testing
9. **RESULTS**
    1. Performance Metrics
10. **ADVANTAGES & DISADVANTAGES**
11. **CONCLUSION**
12. **FUTURE SCOPE**
13. **APPENDIX**

Source Code
GitHub & Project Demo Link

1. INTRODUCTION

1.1 Project overview

The Parkinson's disease estimates that over 10 million people worldwide suffer from Parkinson's disease. Even though Parkinson's cannot be cured, early diagnosis and the right medicine can greatly reduce symptoms and improve quality of life.The researchers discovered that among Parkinson's patients, the sketching pace was slower and the pen pressure was lower. Parkinson's disease symptoms include tremors and muscle rigidity, which make it challenging to draw smooth spirals and waves. Instead than monitoring the speed and pressure with which the pen strikes the paper, it is possible to diagnose Parkinson's disease solely by looking at the drawings. Our objective is to use the HOG method to assess these drawings' visual appearance before training a machine learning model to categorise them.In this research, utilising a Random Forest classifier and the Histogram of Oriented Gradients (HOG) image descriptor to automatically identify Parkinson's disease in hand-drawn spirals and waves.

1.2 Purpose

The Purpose of the project is to predicting the parkinson's disease.Before training a machine learning model to classify these pictures, evaluate their visual appeal using the HOG approach. In this project, hand-drawn spirals and waves with Parkinson's disease were automatically identified using a Random Forest classifier and the HOG image descriptor.In the application user interacts with the UI (User Interface) to upload the image as input.The uploaded image is analyzed by the model which is integrated. Once the model analyzes the uploaded image, the prediction is showcased on the UI .

2. LITERATURE SURVEY

Existing Problem

An early notable attempt in the area of image recognition research is by Huanliang in 1959. The origin of a great deal of research work in the early sixties was based on an approach known as analysis by-synthesis method suggested by Eden in 1968. The great importance of Eden's work was that he formally proved that all parkinson disease people can't able to draw a sprial or wave form, a point that was implicitly included in previous works.

2.2 References

1. Dr. Anupam Bhatia and Raunak Sulekh (2019), “Predictive Model for Parkinson’s Disease through Naive Bayes Classification” In this study, Naive Bayes was applied to predict the performance of the dataset. Rapid miner 7.6.001 is a tool, which was used to explore, statistically analyze, and mine the data. The Naive Bayes model performs with 98.5 % accuracy, and 99.75% of precision.

2. Carlo Ricciardi, et al (2019), “the use of gait analysis’ parameters to categorise Parkinsonism: A information mining method” in this gadget, Random woodland is

used for class in conjunction with comparing it with Gradient Boosted trees. these consequences are being categorized into three special categories specifically PSP, De Novo Parkinson's disorder and stable Parkinson's disease with their accuracy being as high as 86.4% in comparison to Gradient Boosted trees which have been correct to a meagre 70%. additionally the precision price of Random forest changed into most of ninety % against Gradient Boosted trees which were around most of 85%.

3.Gao et al. (2019) proposed a specific prediction and classification technique for Parkinson data analysis. This work was implemented with the help of data preprocessing techniques, cross fold validations, and ML approaches. The neuro data features and the tremor data features were analyzed to predict the symptoms. The methods used in this work provided valuable results but lacked in PD detection and sensitivity rate. Many research works were inherited with different perspectives for predicting PD.

4.Rastegari et al (2021) evolved data gain evaluation model for detecting PD capabilities from the given dataset. This technique used diverse ML and data gain approached in mixed manner. This method labored well in PD findings. however they produced insignificant effects as compared to DL primarily based PD evaluation models. Seppi et al. (2021) analyzed and carried out Parkinson treatment evaluation for various non-motor signs. This work cautioned that the remedy analysis allows to improve and update the future subsequent level treatments. on this regard, the paintings gathered the evidences of numerous treatments and produced valuable suggestions. Many systems created clinical review reviews on PD and remedies however those structures were no longer equipped with own specific techniques.

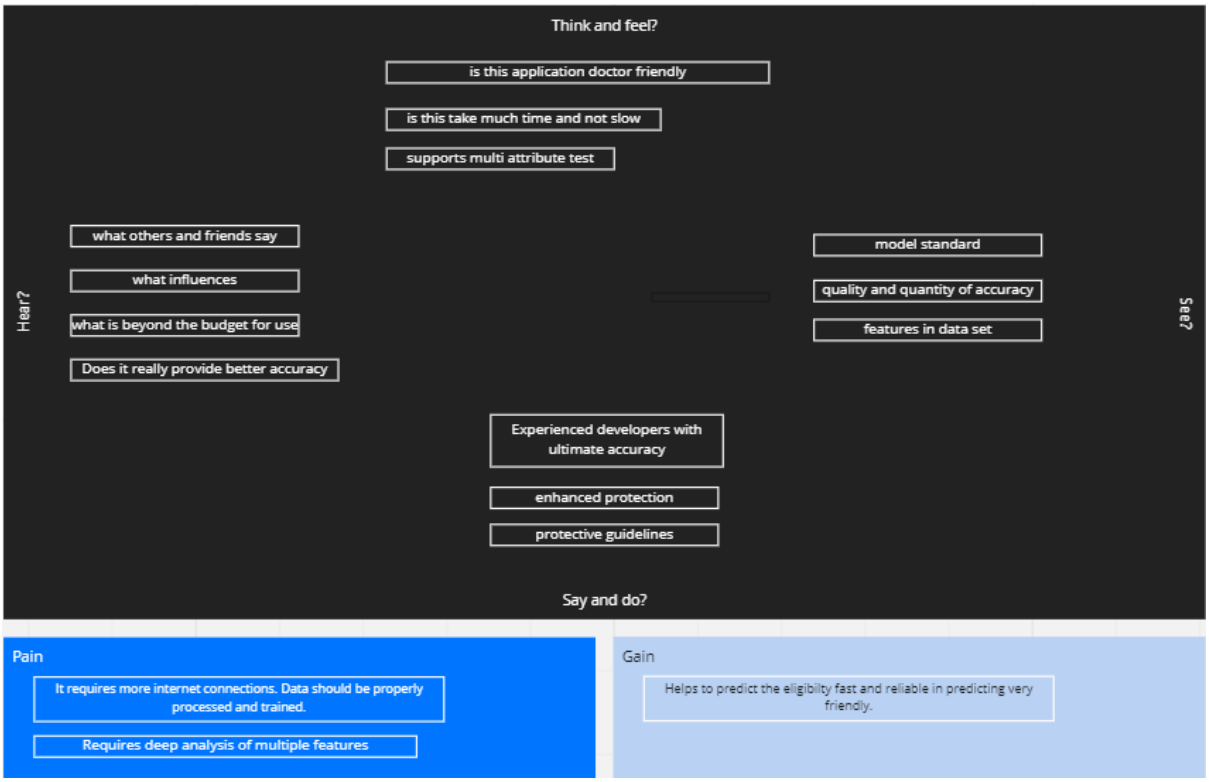
2.3 Problem Statement Definition

According to the Parkinson's Foundation, there are more than 10 million people with Parkinson's disease worldwide.Parkinson's cannot be cured, but with early detection and the right medication, symptoms and quality of life can be greatly enhanced. The researchers discovered that patients with Parkinson's had slower drawing speeds and lower pen pressure. The inability to draw smooth spirals and waves is one of the symptoms of Parkinson's disease, which is characterised by tremors and rigidity in the muscles.

3. IDEATION & PROPOSED SOLUTION

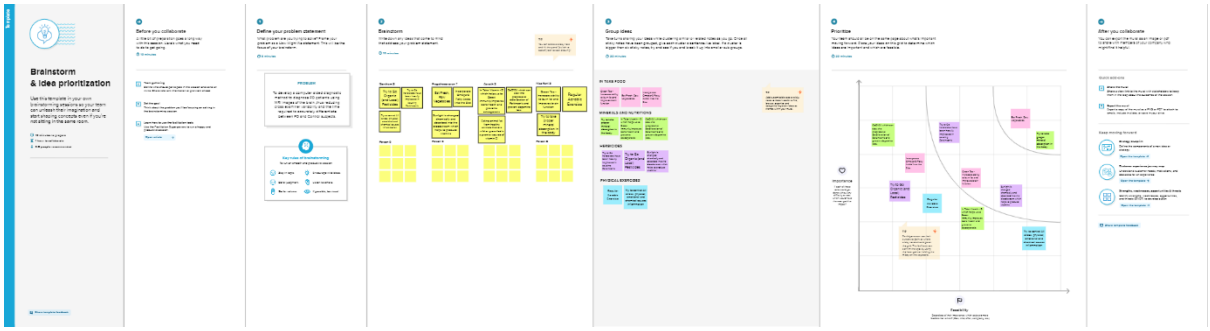
3.1 Empathy Map Canvas

An empathy map is a collaborative tool teams can use to gain a deeper insight into their customers. Much like a user persona, an empathy map can represent a group of users, such as a customer segment. The empathy map was originally created by Dave Gray and has gained much popularity within the agile community.



3.2 Ideation & Brainstorming

Ideation refers to the whole creative process of coming up with and communicating new ideas.Brainstorming is a group problem-solving method that involves the spontaneous contribution of creative ideas and solutions. This technique requires intensive, freewheeling discussion in which every member of the group is encouraged to think aloud and suggest as many ideas as possible based on their diverse knowledge.



3.3 Proposed Solution

Using the machine learning model, developing the model that automatically predicts the parkinson disease. Our goal is to quantify the visual appearance(using HOG method) of these drawings and then train a machine learning model to classify them. In this project using, Histogram of Oriented Gradients (HOG) image descriptor along with a Random Forest classifier to automatically detect Parkinson's disease in hand-drawn images of spirals and waves.

IDEA/SOLUTION DESCRIPTION:

The image database contains spiral and wave images for both the healthy and parkinson peoples. 80 percent of the images for training and 10 percent images for testing. We will create our model using HOG method. It works better for data that are represented as images; this is the reason why HOG works well for image identification problems. we will load our model. Then we will use ImagedataGenerator to get to the test folder and call the image files collectively. And will call the predict function and set it with the image generator function. Now, we will plot our results along with the images.

NOVELTY/UNIQUENESS:

Unlike other models here our model performs better with random forest classifier and histogram of oriented gradients (HOG) . These drawings' visual appearance is evaluated (using the HOG approach) before a machine learning model is trained to categorise them. Hand-drawn spiral and wave images are used to automatically detect Parkinson's illness using the Histogram of Oriented Gradients (HOG) image descriptor and a Random Forest classifier.The combination of HOG method and Random forest classifier improves the performance of the model and acuuracy.

SOCIAL IMPACT/CUSTOMER SATISFACTION:

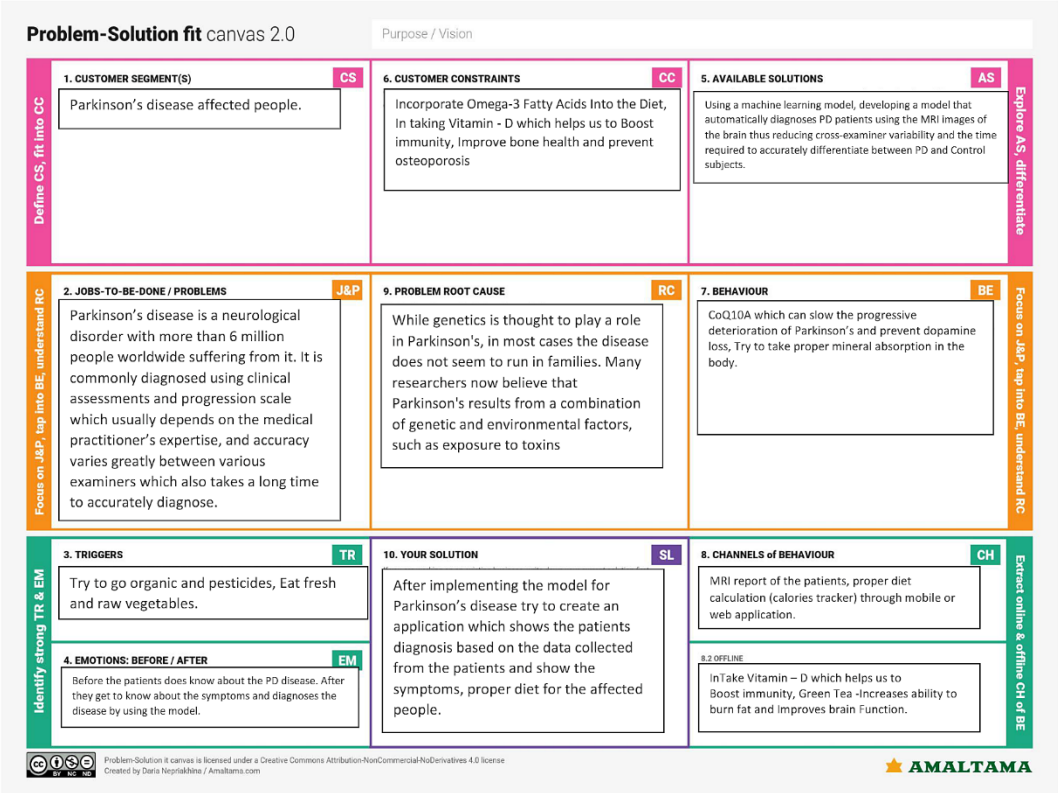
1. Easily find out whether the patient has PD disease or not.
2. Diagnosis of the disease.
3. Reduce the death rate.
4. Improve the success rate.

Business model :

Detection of parkinson disease refers to a model’s (machine’s) capacity to detect any hand drawn images of spiral and wave classify them whether the patient has affected by parkinson or not.

We used the HOG(histogram of oriented gradients)algorithm and Random forest classifier to predict the parkinson disease in this project.

3.4 Problem Solution Fit



4. Requirements Analysis

4.1 Functional Requirements:

1. Image Data:

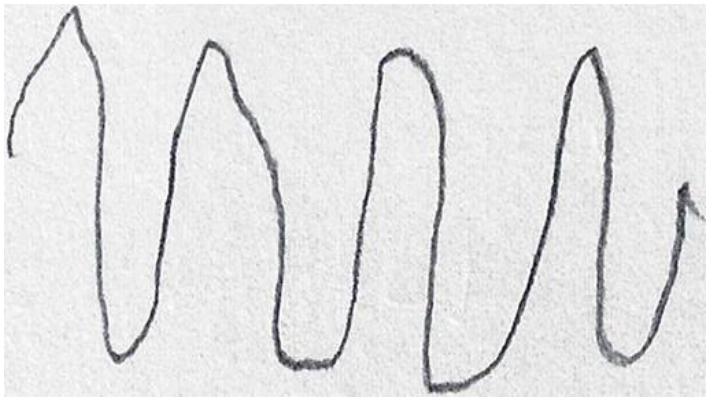
The handdrawn image is the ability of model to predict the parkinson disease. It is a hard task for the people to know about whether the person has affected by parkinson disease.

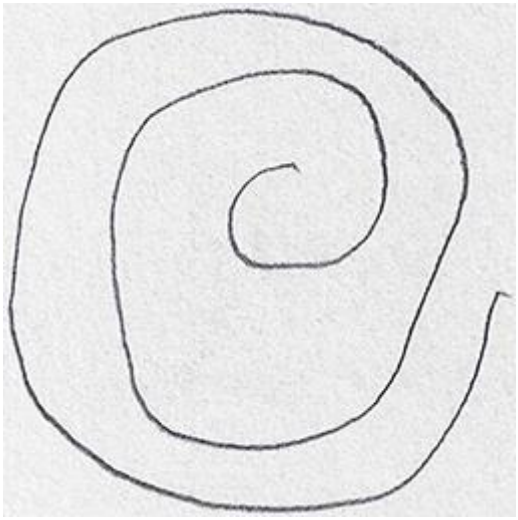
2. Hosting (also known as Web site hosting, Web hosting, and Webhosting) is the business of housing, serving, and maintaining files for one or more Web sites. More important than the computer space that is provided for Web site files is the fast connection to the Internet.

3. Use the spiral and wave dataset of handdrawn images to train a HOG method and Random forest classifier to predict the spiral and wave given an image. First obtain the training and validation data.

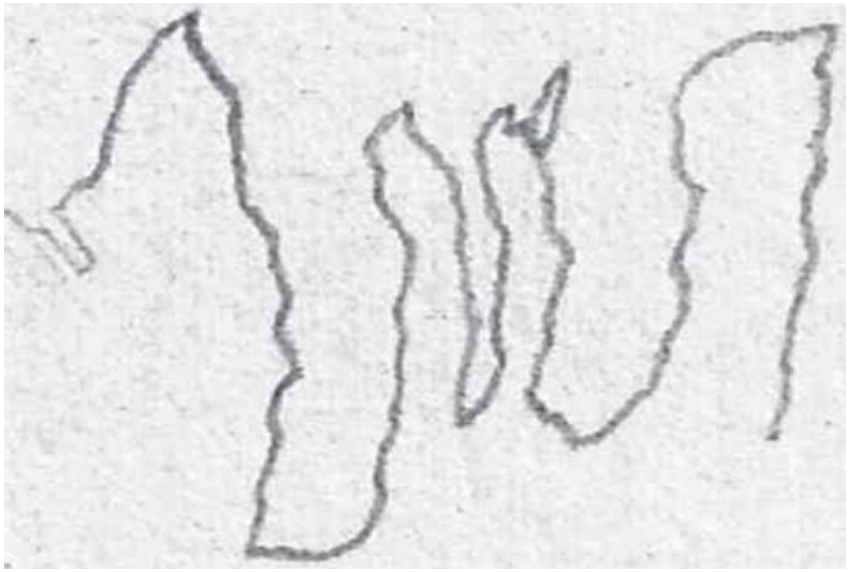
4.Dataset The dataset consists of two forms of data which is hand drawn images of spiral and wave.Each spiral and wave images has parkinson affected person's images and healthy person image.

Healthy :





Parkinson :



5. Cloud: The cloud provides a number of IT services such as servers,databases, software, virtual storage, and networking, among others. In layman's terms, Cloud Computing is defined as a virtual platform that allows you to store and access your data over the internet without any limitations.

4.2 Non-Functional Requirements

Usability:

Models for image recognition can help with many illnesses' diagnosis. The models can be trained to scan MRI or X-ray pictures as well as other visual outputs to find and flag up medical anomalies that they have been trained to recognise.

Reliability:

1) the system not only produces a classification of the image but also it detects various disease associated with this kind of dataset.
2) the generative models can perform recognition driven segmentation.

Performance:

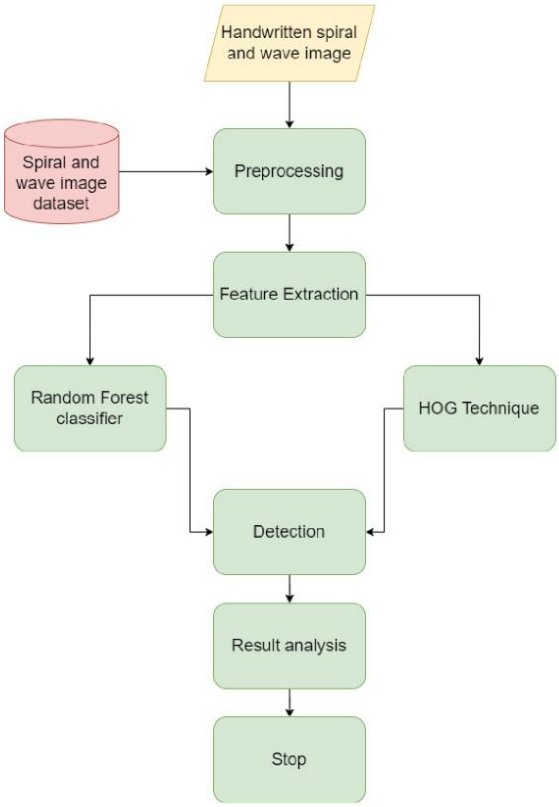
The neural network uses the examples to automatically infer rules for recognizing handdrawn images. Furthermore, by increasing the number of training examples, the network can learn more about handwriting, and so improve its accuracy. There are a number of ways and algorithms to recognize handdrawn images, including Deep Learning/CNN, SVM, Gaussian Naive Bayes, KNN, Decision Trees, Random Forests, etc.

Accuracy:

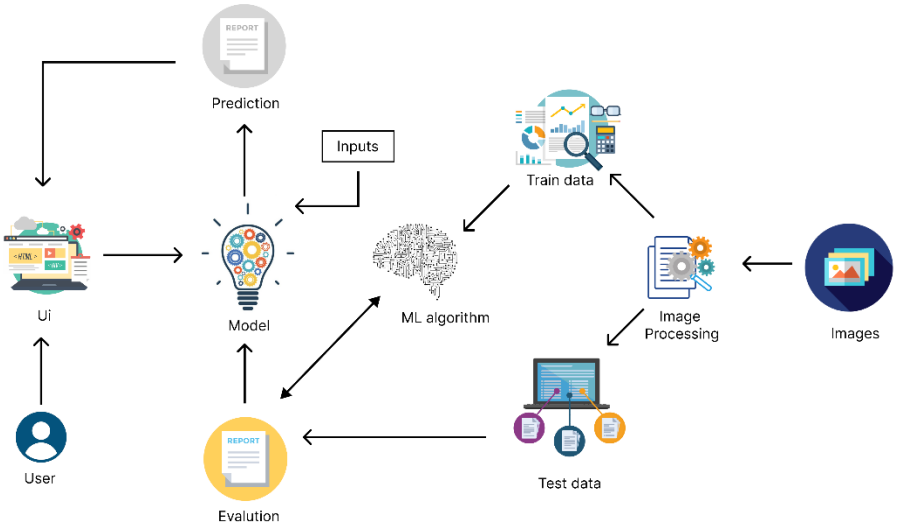
Spiral and wave dataset - Classification of image (accuracy=76%).
The dataset is an image dataset of handdrawn image which consists of sprial and wave form. It has has 80 percent training images and 20 percent test images.

5. PROJECT DESIGN

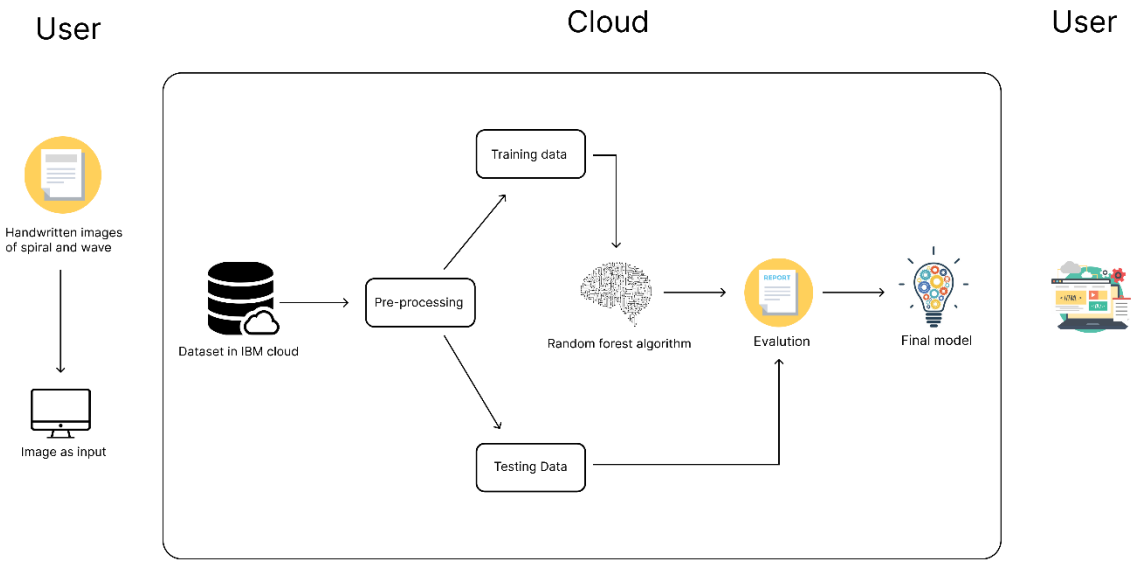
5.1 DataFlow Diagram



5.2 Solution & Technical Architecture



Interactive Technology Architecture



5.3 User Stories

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
Common User	Dashboard	USN-1	As a user, I can I must be able to upload image of spiral and wave handwritten.	I can upload or take image	High	Sprint-1
		USN-2	As a user, I will receive the diagnosis as to whether I have Parkinson's disease or not	I can receive the diagnosis	High	Sprint-1
		USN-3	As a user, I receive the severity of the Parkinson's	I can receive the severity of the retinopathy	Medium	Sprint-2
		USN-4	As a user, I can receive the suggested remedy	I can receive the suggested remedy	Medium	Sprint-2

6. Project Planning & Scheduling

6.1 Sprint Planning

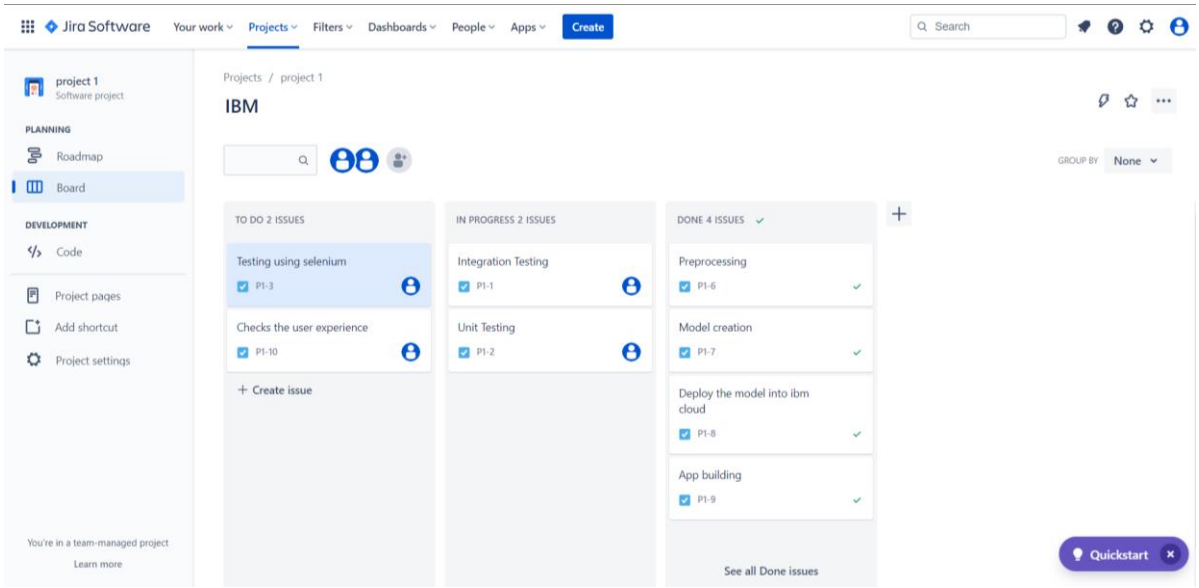
Sprint	Functional Requirement	Task
Sprint-1	Image Data	As a User need to collect the Image Data of Handly Written Images to train the model.
Sprint-2	Dash Board or Website	We using Python Flask Framework to create a dynamic Webpage to host our model (UI).
Sprint-3	Classifier Model	Using random forest and HOG Method for classification
Sprint-4	Cloud	Hosting the Organized appication in Cloud platform.

6.2 Sprint Estimation and Delivery schedule

TITLE	DESCRIPTION	DATE
Prepare Empathy Map	Prepare Empathy Map Canvas to capture the user Pains & Gains, Prepare list of problem statements	24 SEPTEMBER 2022
Literature Survey & Information Gathering	Literature survey on the selected project & gathering information by referring the, technical papers,research publications etc.	28 SEPTEMBER 2022
Ideation	List the by organizing the brainstorming session and prioritize the top 3 ideas based on the feasibility & importance.	25 SEPTEMBER 2022
Proposed Solution	Creation of proposed solution document, which includes the novelty, feasibility of idea, business model, social impact, scalability of solution, etc.	23 SEPTEMBER 2022
Problem Solution Fit	Creation of problem solution fit document.	30 SEPTEMBER 2022
Solution Architecture	Creation of solution architecture document.	28 SEPTEMBER 2022
Customer Journey	Prepare the customer journey maps to understand the user interactions & experiences with the application.	20 OCTOBER 2022
Data Flow Diagrams	Draw the data flow diagrams and submit for review.	9 OCTOBER 2022
Technology Architecture	Prepare the technology architecture diagram.	10 OCTOBER 2022

6.3 Reports from jiira :

- Jira Software is an agile project management tool that supports any agile methodology, be it scrum, kanban, or your own unique flavor. From agile boards, backlogs, roadmaps, reports, to integrations and add-ons you can plan, track, and manage all your agile software development projects from a single tool.
- We can make mention our bugs or issues there and we can proceed accordingly
- We can make a group and share it to the group members and we can work it as a group



7.1 Feature [1]

i) Using Random forest Model in our Project : Random Forest grows multiple decision trees which are merged together for a more accurate prediction. The logic behind the Random Forest model is that multiple uncorrelated models (the individual decision trees) perform much better as a group than they do alone

ii) The Working process of random forest :

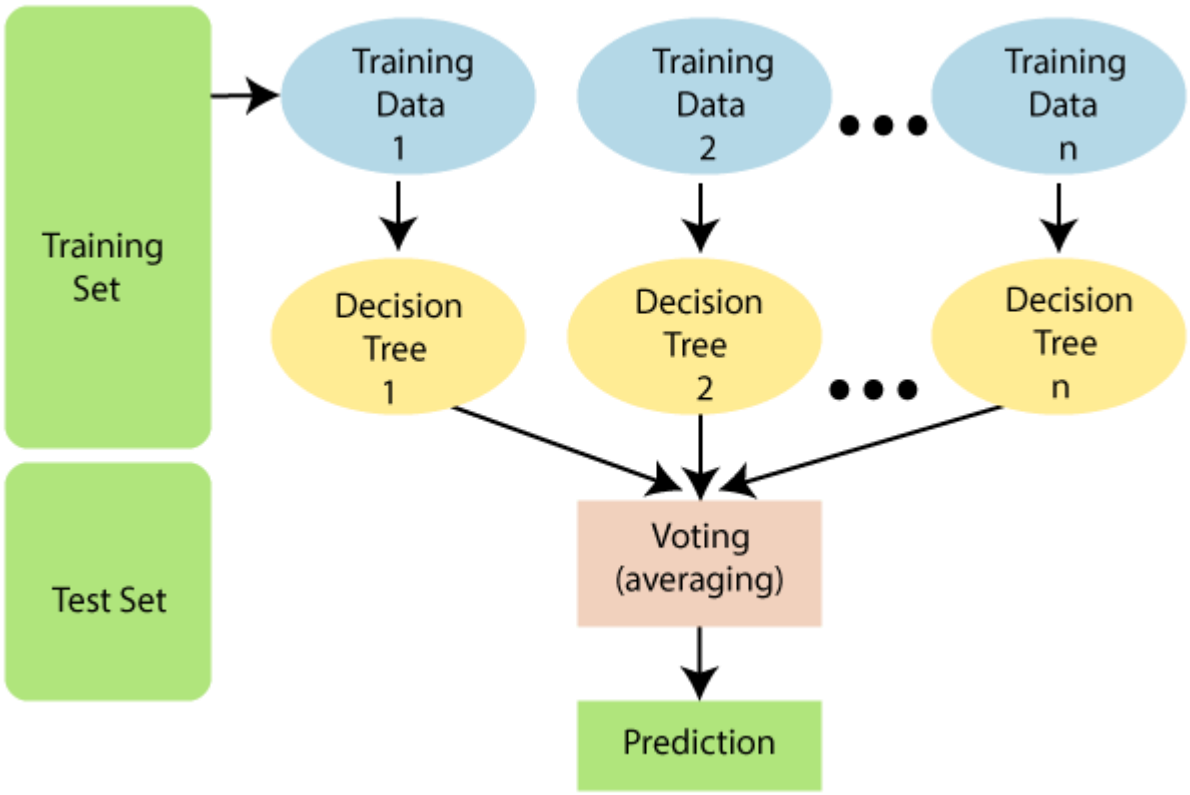
Step-1: Select random K data points from the training set.

Step-2: Build the decision trees associated with the selected data points (Subsets).

Step-3: Choose the number N for decision trees that you want to build.

Step-4: Repeat Step 1 & 2.

Step-5: For new data points, find the predictions of each decision tree, and assign the new data points to the category that wins the majority votes



7.2 Feature [2]

ii).Using Flask application in our Project : Flask is a micro web framework written in Python. It is classified as a microframework because it does not require particular tools or libraries. It has no database abstraction layer, form validation, or any other components where pre-existing third-party libraries provide common functions. However, Flask supports extensions that can add application features as if they were implemented in Flask itself. Extensions exist for object-relational mappers, form validation, upload handling, various open authentication technologies and several common framework related tools.

8. TESTING

8.1 Test Cases

Testing is defined as an activity to check whether the actual results match the expected results and to ensure that the software system is defect free. It involves the execution of a software component or system component to evaluate one or more properties of interest. Software testing also helps to identify errors, gaps, or missing requirements in contrary to the actual requirements.

i)Unit Testing:

When the testing happens for some individual group or some related units then that type of testing is called as Unit Testing.It is often done by programmer to test the part of the program he or she has implemented.Unit Testing is successful means all the modules has been successfully tested and it can proceed further.

ii)Functional Testing:

This type of testing is tested because to check the functional components or the functionality required from the system is gained or not .It actually falls under the testing of the Black Box testing of Software Engineering.This part includes the feeding of the inputs in the system or the project and to check if that system or the project is getting the same value or not as expected if not then calculate the error as wanted and check for more.Functional Testing of this project mainly involves below things. All of these are tested successfully and errors are also calculated.

i)Verifyng the input image
ii)Verifying the work flow
iii)Correct recognition and calculate the error

iii)Integration Testing:

In a total project or the system, many groups of components are getting added or summed up in the purpose of the project query. Integration testing is about to check the interaction between various modules of the project or the system. This module also includes the hardware and the software requirements of the project.All the individual modules are integrated and tested together.All the best and extreme cases that the modules are interacting or not are successfully checked and passed,errors are calculated for the deep learning platforms.

iv)System Testing:

This type of testing is actually meant for the system or the project and also the platform and the integrated softwares and tools,technologies are also tested.The idea or purpose behind the system testing is to check all the requirements that will be provided by the system.This application of the project along with the tools and technologies has been tested in both windows and linux.It passed successfully.

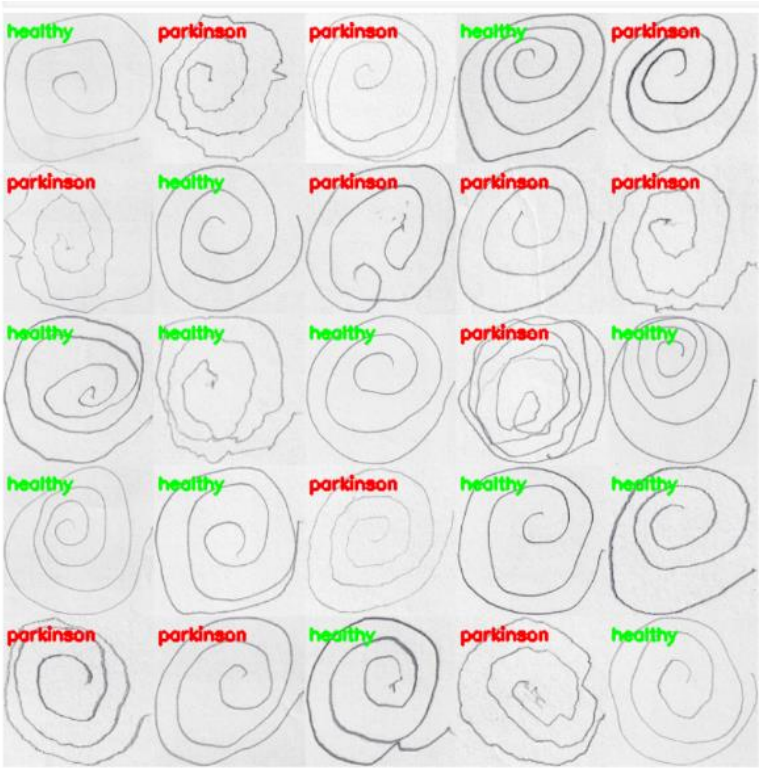
8.2 User Accceptance Testing

This is a type of system or software testing where a system has been tested for availability.The purpose of this test is to check the business requirements and assess whether it will be accepted for delivery.

9. RESULTS

9.1 Performance Metrics

i).Model Metrics : Our model perform 82% of accuracy when train and testing session.



Accracy and confusion matrix:

Model Evaluation

```
]:
predictions = model.predict(X_test)

cm = confusion_matrix(y_test, predictions).flatten()
print(cm)
(tn, fp, fn, tp) = cm
accuracy = (tp + tn) / float(cm.sum())
print(accuracy)
```

```
[13  2  5 10]
0.7666666666666667
```

Precision:

```
precision_positive = metrics.precision_score(y_test, predictions, pos_label=1)
precision_negative = metrics.precision_score(y_test, predictions, pos_label=0)
precision_positive,precision_negative
```

```
(0.9230769230769231, 0.8235294117647058)
```

Jaccard score:

```
jaccard = metrics.jaccard_score(y_test, predictions)
jaccard
```

```
0.75
```

Hamming loss:

```
hamming_loss = metrics.hamming_loss(y_test, predictions)
hamming_loss
```

```
0.13333333333333333
```

Cross_entrophy_loss:

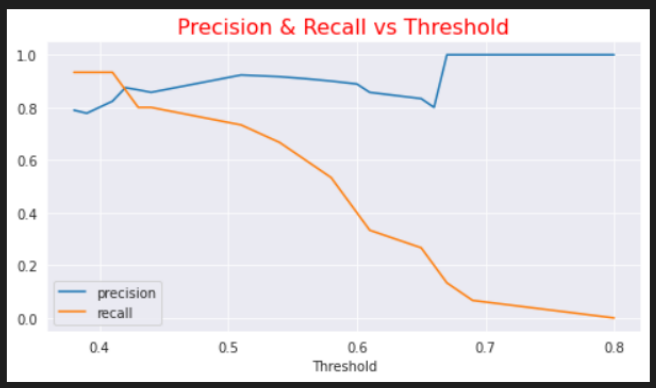
```
cross_entropy_loss = metrics.log_loss(y_test, prob_test)
cross_entropy_loss
```

```
0.4812316995495089
```

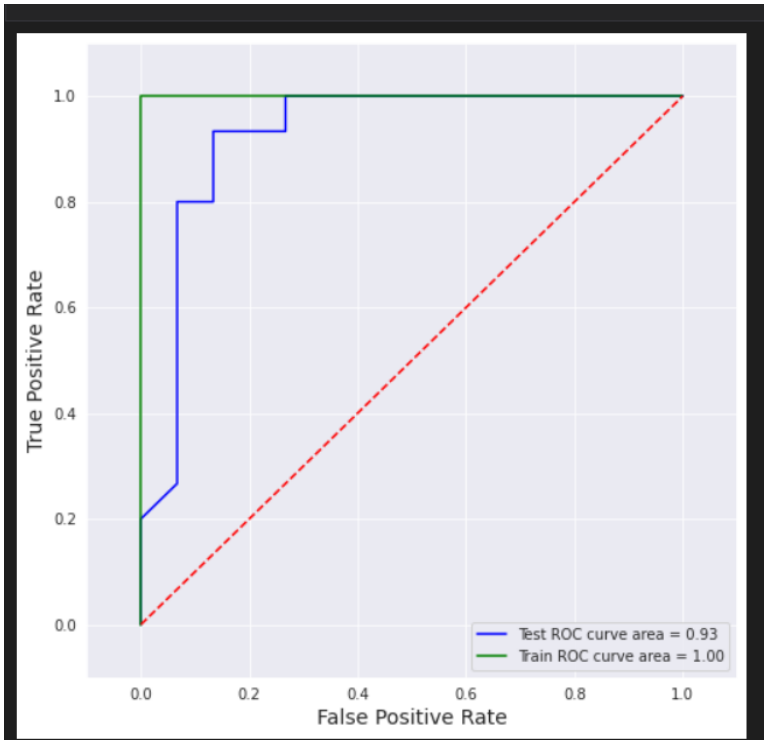
Precision -recall curve:

```
pre, rec, thr = metrics.precision_recall_curve(y_test, prob_test)
plt.figure(figsize=(8,4))
plt.plot(thr, pre[:-1], label='precision')
plt.plot(thr, rec[1:], label='recall')
plt.xlabel('Threshold')
plt.title('Precision & Recall vs Threshold', c='r', size=16)
plt.legend()
plt.show()
```

[27]



ROC score:



F1 – score:

```
f1_positive = metrics.f1_score(y_test, predictions, pos_label=1)
f1_negative = metrics.f1_score(y_test, predictions, pos_label=0)
f1_positive, f1_negative
```

```
(0.8571428571428571, 0.8749999999999999)
```

Recall :

```
from sklearn import metrics
recall_sensitivity = metrics.recall_score(y_test, predictions,pos_label=0)
recall_specificity = metrics.recall_score(y_test, predictions, pos_label=0)
recall_sensitivity, recall_specificity
```

```
(0.9333333333333333, 0.9333333333333333)
```

ADVANTAGES:

- Greater cost reduction in hospitals for testing
- Helps in early diagnosis of the disease
- Chances of recovery is higher

DISADVANTAGES:

- Chances of prediction to be wrong for least number of time which can cause problems
- Vast feature in dataset on discovery of time for the disease making the model inefficient to keep up the metrics
- Since its a web application it requires scaling of web application to handle concurrent requests after certain threshold

Test report :

IBM_Testing_report.html

Report generated on 16-Nov-2022 at 13:03:10 by pytest-html v3.1.1

Environment

Packages	{"pluggy": "1.0.0", "py": "1.11.0", "pytest": "7.1.3"}
Platform	Windows-10-10.0.22621-SP0
Plugins	{"html": "3.1.1", "metadata": "2.0.2"}
Python	3.7.0

Summary

3 tests ran in 26.01 seconds.

(Un)check the boxes to filter the results.

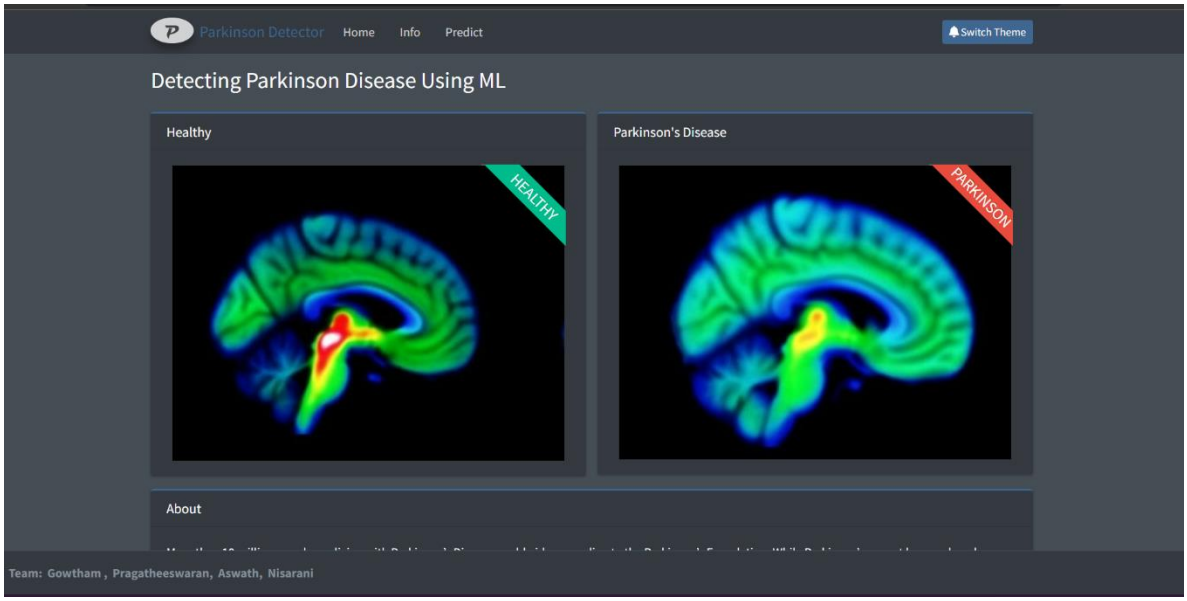
☑ 3 passed, ☐ 0 skipped, ☐ 0 failed, ☐ 0 errors, ☐ 0 expected failures, ☐ 0 unexpected passes

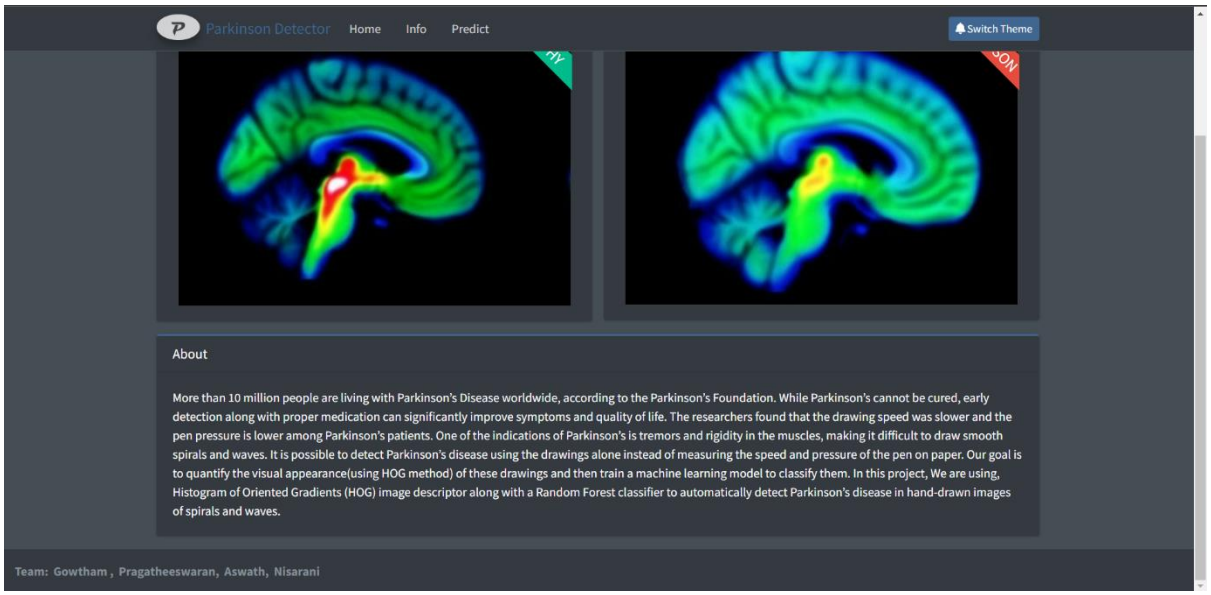
Results

Show all details / Hide all details

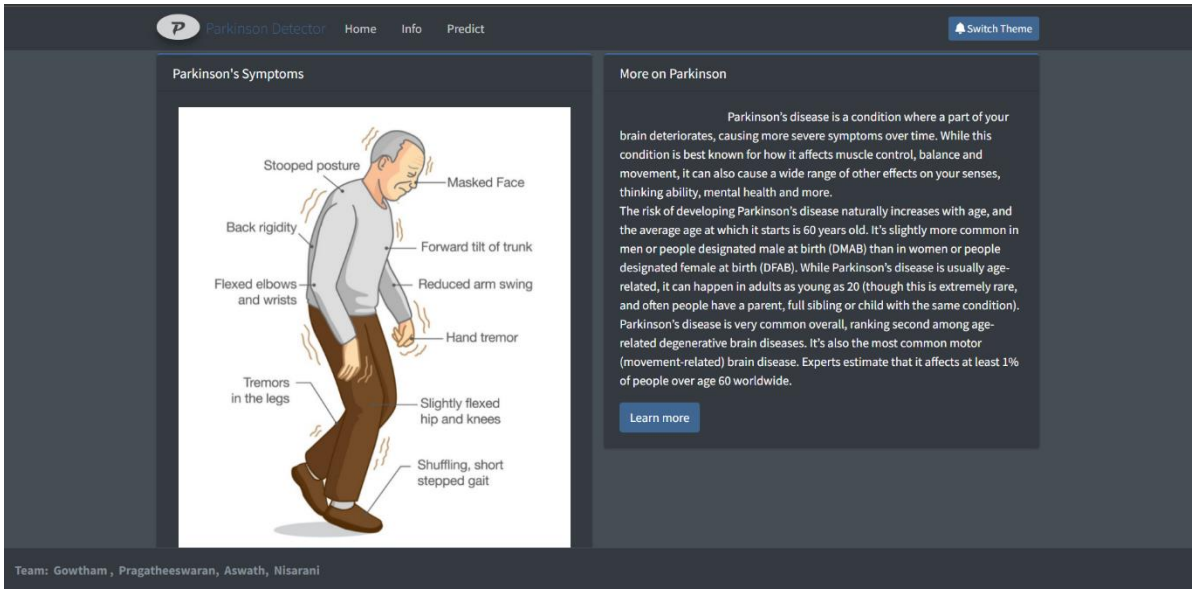
Result	Test	Duration	Links
Passed	test_ibm.py::test_selenium	8.35	
No log output captured.			
Passed	test_ibm.py::test_sum	8.02	
No log output captured.			
Passed	test_ibm.py::test_sum_tuple	9.31	
No log output captured.			

i) Home page of the application :

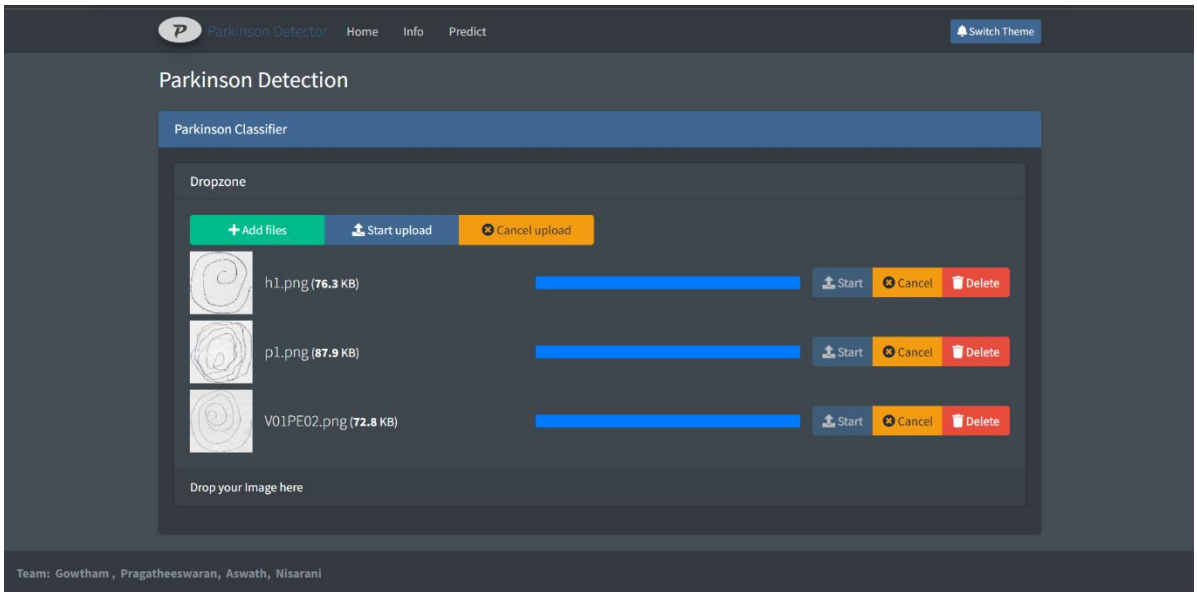

Parkinson Detector
Home
Info
Predict
Switch Theme
Detecting Parkinson Disease Using ML
Healthy
HEALTHY
Parkinson's Disease
PARKINSON
About
Team: Gowtham , Pragatheeswaran, Aswath, Nisarani

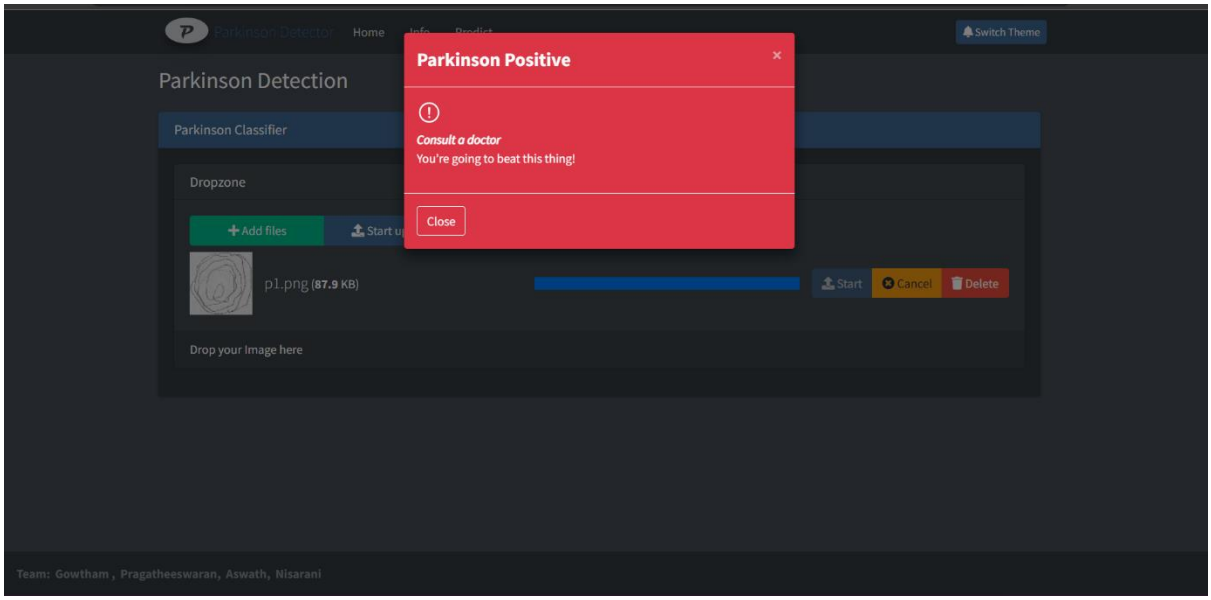

Parkinson Detector
Home
Info
Predict
Switch Theme
About
More than 10 million people are living with Parkinson's Disease worldwide, according to the Parkinson's Foundation. While Parkinson's cannot be cured, early detection along with proper medication can significantly improve symptoms and quality of life. The researchers found that the drawing speed was slower and the pen pressure is lower among Parkinson's patients. One of the indications of Parkinson's is tremors and rigidity in the muscles, making it difficult to draw smooth spirals and waves. It is possible to detect Parkinson's disease using the drawings alone instead of measuring the speed and pressure of the pen on paper. Our goal is to quantify the visual appearance(using HOG method) of these drawings and then train a machine learning model to classify them. In this project, We are using, Histogram of Oriented Gradients (HOG) image descriptor along with a Random Forest classifier to automatically detect Parkinson's disease in hand-drawn images of spirals and waves.
Team: Gowtham , Pragatheeswaran, Aswath, Nisarani

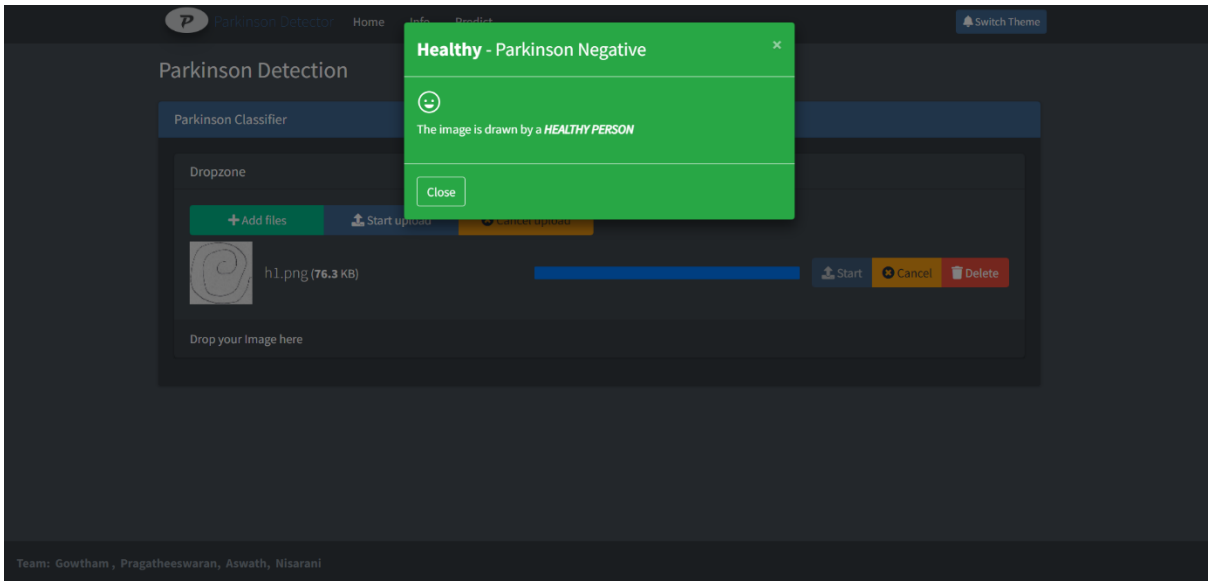
ii) Info Page :



iii) Prediction Page :







10.CONCLUSION

The detection of parkinson disease using machine learning methods has been implemented. Random forest classifier have been trained and tested on the same data in order acquire the comparison between the classifiers. Utilising these machine learning techniques, a high amount of accuracy can be obtained.Compared to other research methods, this method focuses on which classifier works better by improving the accuracy of classification models by more than 82%.

11. Future scope

The proposed system takes images as input. The same system with further modifications and improvements in the dataset and the model can be used to build for predicting different disease which can be take image as an input and also improve the accuracy of the model to above 90 percent by using XGBoost algorithm.

12. APPENDIX

Python:

Python is an interpreted, high-level, general purpose programming language created by Guido Van Rossum and first released in 1991, Python's design philosophy emphasizes code Readability with its notable use of significant White space. Its language constructs and object oriented approach aim to help programmers write clear, logical code for small and large-scale projects. Python is dynamically typed and garbage collected. It supports multiple programming paradigms, including procedural, objectoriented, and functional programming

Numpy:

NumPy is a Python library used for working with arrays. It also has functions for working in domain of linear algebra, Fourier transform, and matrices. Numpy which stands for Numerical Python, is a library consisting of multidimensional array objects and a collection of routines for processing those arrays. Using NumPy, mathematical and logical operations on arrays can be performed. This tutorial explains the basics of NumPy such as its architecture and environment. It also discusses the various array functions, types of indexing, etc. It is an open source project and you can use it freely. NumPy stands for Numerical Python. NumPy aims to provide an array object that is up to 50x faster than traditional Python lists. The array object in NumPy is called ndarray, it provides a lot of supporting functions that make working with ndarray very easy. Arrays are very frequently used in data science, where speed and resources are very important.

Jupyter Notebook:

JupyterLab is a web-based interactive development environment for Jupyter notebooks, code, and data. JupyterLab is flexible: configure and arrange the user interface to support a wide range of workflows in data science, scientific computing, and machine learning. JupyterLab is extensible and modular: write plugins that add new components and integrate with existing ones.

Machine Learning:

Machine learning is a method of data analysis that automates analytical model building. It is a branch of artificial intelligence based on the idea that systems can learn from data, identify patterns and make decisions with minimal human intervention.

SOURCE CODE

app.py

```
import pickle
```

```
import cv2
from skimage import feature
from flask import Flask, request, render_template
import os.path

app = Flask(__name__)
@app.route("/test")
def test():
   return render_template("predict.html")

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

@app.route("/info")
def information():
   return render_template("info.html")
@app.route('/predict', methods=['GET', 'POST'])
def upload():
   if request.method == 'POST':
      f=request.files['file'] #requesting the file
      basepath=os.path.dirname(__file__)#storing the file directory
      filepath=os.path.join(basepath, "uploads", f.filename) #storing the file in uploads folder
      f.save(filepath) #saving the file
      #Loading the saved model
      print("[INFO] loading model...")
          model=pickle.loads(open("C:/Users/Admin/Desktop/Parkinson's
           disease/Training/parkinson.pkl", "rb").read())
      # 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])
      ls=["healthy","parkinson"]
      result = ls[preds[0]]
      # # draw the colored class label on the output image and add it to the set of output images
      # 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(host='0.0.0.0', port=8000, debug=False)
```

Home.html :

```
<!DOCTYPE html>
<!--
This is a starter template page. Use this page to start your new project from
scratch. This page gets rid of all links and provides the needed markup only.
-->
<html lang="en">
  <head>
    <meta charset="utf-8" />
    <meta name="viewport" content="width=device-width, initial-scale=1" />
    <title>Parkinson Detection</title>

    <!-- Google Font: Source Sans Pro -->
    <link
      rel="stylesheet"
      href="https://fonts.googleapis.com/css?family=Source+Sans+Pro:300,400,400i,700&dis
play=fallback"
    />
    <!-- Font Awesome Icons -->
    <link
      rel="stylesheet"
      href="../static/plugins/fontawesome-free/css/all.min.css"
    />
    <!-- Theme style -->
    <link rel="stylesheet" href="../static/dist/css/adminlte.min.css" />
    <link
      rel="stylesheet"
      href="https://cdn.jsdelivr.net/npm/admin-lte@3.1/dist/css/adminlte.min.css"
    />
  </head>
  <body
    class="hold-transition layout-top-nav layout-footer-fixed layout-navbar-fixed dark-mode"
  >
    <div class="wrapper">
      <!-- Navbar -->
      <nav
        class="main-header navbar navbar-expand-md navbar-dark accent-primary"
      >
        <div class="container">
          <a href="/" class="navbar-brand">
            <img
              src="../static/dist/img/plogo.jpg"
              alt="Logo"
              class="brand-image img-circle elevation-3"
              style="opacity: 0.8"
            />
            <span class="brand-text font-weight-light"
              ><b>Parkinson Detector</b></span
            >
```

```
    </a>

    <button
      class="navbar-toggler order-1"
      type="button"
      data-toggle="collapse"
      data-target="#navbarCollapse"
      aria-controls="navbarCollapse"
      aria-expanded="false"
      aria-label="Toggle navigation"
    >
      <span class="navbar-toggler-icon"></span>
    </button>

    <div class="collapse navbar-collapse order-3" id="navbarCollapse">
      <!-- Left navbar links -->
      <ul class="navbar-nav">
        <li class="nav-item">
          <a href="/" class="nav-link">Home</a>
        </li>
        <li class="nav-item">
          <a href="/info" class="nav-link">Info</a>
        </li>
        <li class="nav-item">
          <a href="/test" class="nav-link">Predict</a>
        </li>
      </ul>
    </div>

    <!-- Right navbar links -->
    <ul class="order-1 order-md-3 navbar-nav navbar-no-expand ml-auto">
      <li class="nav-item">
        <button
          type="button"
          onclick="switchTheme()"
          class="btn btn-primary btn-block btn-sm"
        >
          <i class="fa fa-bell"></i> Switch Theme
        </button>
      </li>
    </ul>
  </div>
</nav>
<!-- /.navbar -->

<!-- Content Wrapper. Contains page content -->
<div class="content-wrapper">
  <!-- Content Header (Page header) -->
  <div class="content-header">
    <div class="container">
      <div class="row mb-2">
        <div class="col-sm-6">
```

```
      <h1 class="m-0">Detecting Parkinson Disease Using ML</h1>
     </div>
     <!-- /.col -->

     <!-- /.col -->
    </div>
    <!-- /.row -->
   </div>
   <!-- /.container-fluid -->
</div>
<!-- /.content-header -->

<!-- Main content -->
<div class="content">
  <div class="container">
    <div class="row">
     <div class="col-lg-6">
      <div class="card card-primary card-outline">
       <div class="card-header">
        <h5 class="card-title m-0">Healthy</h5>
       </div>
       <div class="card-body">
        <div class="col-sm-12">
         <div class="position-relative">
          <img
           src="../static/dist/img/healthy.jpg"
           alt="Photo 1"
           class="img-fluid"
           width="650"
           height="400"
          />
          <div class="ribbon-wrapper ribbon-lg">
           <div class="ribbon bg-success text-lg">Healthy</div>
          </div>
         </div>
        </div>
       </div>
      </div>
      <!-- /.card -->
     </div>
     <!-- /.col-md-6 -->
     <div class="col-lg-6">
      <div class="card card-primary card-outline">
       <div class="card-header">
        <h5 class="card-title m-0">Parkinson's Disease</h5>
       </div>
       <div class="card-body">
        <div class="col-sm-12">
         <div class="position-relative">
          <img
           src="../static/dist/img/parkinson_affected.jpg"
           alt="Photo 1"
```

```
                class="img-fluid"
                width="650"
                height="300"
               />
               <div class="ribbon-wrapper ribbon-lg">
                <div class="ribbon bg-danger text-lg">Parkinson</div>
               </div>
              </div>
             </div>
            </div>
           </div>
          </div>
          <!-- /.col-md-6 -->
        </div>
        <div class="row">
          <div class="col-lg-12">
            <div class="card card-primary card-outline">
              <div class="card-header">
                <h5 class="card-title m-0">About</h5>
              </div>
              <div class="card-body">
                <p class="card-text">
                  More than 10 million people are living with Parkinson’s
                  Disease worldwide, according to the Parkinson’s
                  Foundation. While Parkinson’s cannot be cured, early
                  detection along with proper medication can significantly
                  improve symptoms and quality of life. The researchers
                  found that the drawing speed was slower and the pen
                  pressure is lower among Parkinson’s patients. One of the
                  indications of Parkinson’s is tremors and rigidity in the
                  muscles, making it difficult to draw smooth spirals and
                  waves. It is possible to detect Parkinson’s disease using
                  the drawings alone instead of measuring the speed and
                  pressure of the pen on paper. Our goal is to quantify the
                  visual appearance(using HOG method) of these drawings and
                  then train a machine learning model to classify them. In
                  this project, We are using, Histogram of Oriented
                  Gradients (HOG) image descriptor along with a Random
                  Forest classifier to automatically detect Parkinson’s
                  disease in hand-drawn images of spirals and waves.
                </p>
              </div>
            </div>
          </div>
          <!-- /.row -->
        </div>
        <!-- /.container-fluid -->
      </div>
      <!-- /.content -->
    </div>
    <!-- /.content-wrapper -->
```

```
    <!-- Control Sidebar -->
    <aside class="control-sidebar control-sidebar-dark">
      <!-- Control sidebar content goes here -->
    </aside>
    <!-- /.control-sidebar -->

    <!-- Main Footer -->
    <footer class="main-footer">
      <!-- To the right -->
      <div class="float-right d-none d-sm-inline"></div>
      <!-- Default to the left -->
      <strong
        >Team:  <a
          >Gowtham <S></S></a
        >,  <a
          >Pragatheeswaran</a
        >,  <a
          >Aswath</a
        >,  <a
          >Nisarani</a
        ></strong
      >
    </footer>
  </div>
  <!-- ./wrapper -->

  <!-- REQUIRED SCRIPTS -->

  <!-- jQuery -->
  <script src="../static/plugins/jquery/jquery.min.js"></script>
  <!-- Bootstrap 4 -->
  <script src="../static/plugins/bootstrap/js/bootstrap.bundle.min.js"></script>
  <!-- AdminLTE App -->
  <script src="../static/dist/js/adminlte.min.js"></script>
  <!-- AdminLTE for demo purposes -->
  <script src="../static/dist/js/demo.js"></script>
  <script src="https://cdn.jsdelivr.net/npm/admin-
lte@3.1/dist/js/adminlte.min.js"></script>
  <script>
    sessionStorage.setItem("theme", "dark");

    var currentTheme = sessionStorage.getItem("theme");
    var mainHeader = document.querySelector(".main-header");

    if (currentTheme) {
      if (currentTheme === "dark") {
        if (!document.body.classList.contains("dark-mode")) {
          document.body.classList.add("dark-mode");
        }
        if (mainHeader.classList.contains("navbar-light")) {
          mainHeader.classList.add("navbar-dark");
```

```
          mainHeader.classList.remove("navbar-light");
        }
        toggleSwitch.checked = true;
      }
    }

    function switchTheme() {
      if (sessionStorage.getItem("theme") === "dark") {
        if (document.body.classList.contains("dark-mode")) {
          document.body.classList.remove("dark-mode");
        }

        sessionStorage.setItem("theme", "light");
      } else {
        if (!document.body.classList.contains("dark-mode")) {
          document.body.classList.add("dark-mode");
        }

        sessionStorage.setItem("theme", "dark");
      }
    }
  </script>
 </body>
</html>
```

Info.html

```
<!DOCTYPE html>
<!--
This is a starter template page. Use this page to start your new project from
scratch. This page gets rid of all links and provides the needed markup only.
-->
<html lang="en">
 <head>
  <meta charset="utf-8" />
  <meta name="viewport" content="width=device-width, initial-scale=1" />
  <title>Parkinson Detection</title>

  <!-- Google Font: Source Sans Pro -->
  <link
   rel="stylesheet"
   href="https://fonts.googleapis.com/css?family=Source+Sans+Pro:300,400,400i,700&dis
play=fallback"
  />
  <!-- Font Awesome Icons -->
  <link
   rel="stylesheet"
   href="../static/plugins/fontawesome-free/css/all.min.css"
  />
  <!-- Theme style -->
  <link rel="stylesheet" href="../static/dist/css/adminlte.min.css" />
 </head>
```

```
<body
  class="hold-transition layout-top-nav layout-footer-fixed layout-navbar-fixed dark-mode"
>
  <div class="wrapper">
    <!-- Navbar -->
    <nav
      class="main-header navbar navbar-expand-md navbar-dark accent-primary"
    >
      <div class="container">
        <a href="/" class="navbar-brand">
          <img
            src="../static/dist/img/plogo.jpg"
            alt="Logo"
            class="brand-image img-circle elevation-3"
            style="opacity: 0.8"
          />
          <span class="brand-text font-weight-light">Parkinson Detector</span>
        </a>

        <button
          class="navbar-toggler order-1"
          type="button"
          data-toggle="collapse"
          data-target="#navbarCollapse"
          aria-controls="navbarCollapse"
          aria-expanded="false"
          aria-label="Toggle navigation"
        >
          <span class="navbar-toggler-icon"></span>
        </button>

        <div class="collapse navbar-collapse order-3" id="navbarCollapse">
          <!-- Left navbar links -->
          <ul class="navbar-nav">
            <li class="nav-item">
              <a href="/" class="nav-link">Home</a>
            </li>
            <li class="nav-item">
              <a href="/info" class="nav-link">Info</a>
            </li>
            <li class="nav-item">
              <a href="/test" class="nav-link">Predict</a>
            </li>
          </ul>
        </div>

        <!-- Right navbar links -->
        <ul class="order-1 order-md-3 navbar-nav navbar-no-expand ml-auto">
          <li class="nav-item">
            <button
              type="button"
              onclick="switchTheme()"
```

```
          class="btn btn-primary btn-block btn-sm"
        >
          <i class="fa fa-bell"></i> Switch Theme
        </button>
      </li>
    </ul>
  </div>
</nav>
<!-- /.navbar -->

<!-- Content Wrapper. Contains page content -->
<div class="content-wrapper">
  <!-- Content Header (Page header) -->
  <div class="content-header">
    <div class="container">
      <div class="row mb-2">
        <div class="col-sm-6">
          <h1 class="m-0"></h1>
        </div>
        <!-- /.col -->

        <!-- /.col -->
      </div>
      <!-- /.row -->
    </div>
    <!-- /.container-fluid -->
  </div>
  <!-- /.content-header -->

  <!-- Main content -->
  <div class="content">
    <div class="container">
      <div class="row">
        <div class="col-lg-6">
          <div class="card card-primary card-outline">
            <div class="card-header">
              <h5 class="card-title m-0">Parkinson's Symptoms</h5>
            </div>
            <div class="card-body">
              <div class="col-sm-12">
                <div class="position-relative">
                  <img
                    src="../static/dist/img/parkinson_symptoms.png"
                    alt="Symptoms"
                    class="img-fluid"
                  />
                </div>
              </div>
            </div>
          </div>
          <!-- /.card -->
        </div>
```

```
        <!– /.col-md-6 –>
        <div class="col-lg-6">
          <div class="card card-primary card-outline">
            <div class="card-header">
              <h5 class="card-title m-0">More on Parkinson</h5>
            </div>
            <div class="card-body">
              <p class="card-text">
                           &nbs
p;             &
nbsp;            &nbs
p;    Parkinson’s
                disease is a condition where a part of your brain
                deteriorates, causing more severe symptoms over time.
                While this condition is best known for how it affects
                muscle control, balance and movement, it can also cause a
                wide range of other effects on your senses, thinking
                ability, mental health and more.
                <br />The risk of developing Parkinson’s disease naturally
                increases with age, and the average age at which it starts
                is 60 years old. It’s slightly more common in men or
                people designated male at birth (DMAB) than in women or
                people designated female at birth (DFAB). While
                Parkinson’s disease is usually age-related, it can happen
                in adults as young as 20 (though this is extremely rare,
                and often people have a parent, full sibling or child with
                the same condition).
                <br />
                Parkinson’s disease is very common overall, ranking second
                among age-related degenerative brain diseases. It’s also
                the most common motor (movement-related) brain disease.
                Experts estimate that it affects at least 1% of people
                over age 60 worldwide.
              </p>
              <a
                href="https://en.wikipedia.org/wiki/Parkinson%27s_disease"
                class="btn btn-primary"
                >Learn more</a
              >
            </div>
          </div>
        </div>
        <!– /.col-md-6 –>
      </div>

      <!– /.row –>
    </div>
    <!– /.container-fluid –>
  </div>
  <!– /.content –>
</div>
<!– /.content-wrapper –>
```

```
  <!-- Control Sidebar -->
  <aside class="control-sidebar control-sidebar-dark">
    <!-- Control sidebar content goes here -->
  </aside>
  <!-- /.control-sidebar -->

  <!-- Main Footer -->
  <footer class="main-footer">
    <!-- To the right -->
    <div class="float-right d-none d-sm-inline"></div>
    <!-- Default to the left -->
    <strong
      >Team:  <a
        >Gowtham <S></S></a
      >,  <a
        >Pragatheeswaran</a
      >,  <a
        >Aswath</a
      >,  <a
        >Nisarani</a
      ></strong
    >
  </footer>
</div>
<!-- ./wrapper -->

<!-- REQUIRED SCRIPTS -->

<!-- jQuery -->
<script src="../static/plugins/jquery/jquery.min.js"></script>
<!-- Bootstrap 4 -->
<script src="../static/plugins/bootstrap/js/bootstrap.bundle.min.js"></script>
<!-- AdminLTE App -->
<script src="../static/dist/js/adminlte.min.js"></script>
<!-- AdminLTE for demo purposes -->
<script src="../static/dist/js/demo.js"></script>
<script>
  sessionStorage.setItem("theme", "dark");

  var currentTheme = sessionStorage.getItem("theme");
  var mainHeader = document.querySelector(".main-header");

  if (currentTheme) {
    if (currentTheme === "dark") {
      if (!document.body.classList.contains("dark-mode")) {
        document.body.classList.add("dark-mode");
      }
      if (mainHeader.classList.contains("navbar-light")) {
        mainHeader.classList.add("navbar-dark");
        mainHeader.classList.remove("navbar-light");
      }
```

```
        toggleSwitch.checked = true;
      }
    }

    function switchTheme() {
      if (sessionStorage.getItem("theme") === "dark") {
        if (document.body.classList.contains("dark-mode")) {
          document.body.classList.remove("dark-mode");
        }

        sessionStorage.setItem("theme", "light");
      } else {
        if (!document.body.classList.contains("dark-mode")) {
          document.body.classList.add("dark-mode");
        }

        sessionStorage.setItem("theme", "dark");
      }
    }
  </script>
 </body>
</html>
```

Predict.html

```
<!DOCTYPE html>
<!--
This is a starter template page. Use this page to start your new project from
scratch. This page gets rid of all links and provides the needed markup only.
-->
<html lang="en">
 <head>
  <meta charset="utf-8" />
  <meta name="viewport" content="width=device-width, initial-scale=1" />
  <title>Parkinson Detection</title>

  <!-- Google Font: Source Sans Pro -->
  <link
   rel="stylesheet"
   href="https://fonts.googleapis.com/css?family=Source+Sans+Pro:300,400,400i,700&dis
play=fallback"
  />
  <!-- Font Awesome Icons -->
  <link
   rel="stylesheet"
   href="../static/plugins/fontawesome-free/css/all.min.css"
  />
  <!-- Theme style -->
  <link rel="stylesheet" href="../static/dist/css/adminlte.min.css" />
  <!-- dropzonejs -->
  <link
   rel="stylesheet"
   href="../static/plugins/dropzone/min/dropzone.min.css"
```

```
  />
</head>
<body
  class="hold-transition layout-top-nav layout-footer-fixed layout-navbar-fixed dark-mode"
>
  <div class="wrapper">
    <!-- Navbar -->
    <nav
      class="main-header navbar navbar-expand-md navbar-dark accent-primary"
    >
      <div class="container">
        <a href="/" class="navbar-brand">
          <img
            src="../static/dist/img/plogo.jpg"
            alt="Logo"
            class="brand-image img-circle elevation-3"
            style="opacity: 0.8"
          />
          <span class="brand-text font-weight-light">Parkinson Detector</span>
        </a>

        <button
          class="navbar-toggler order-1"
          type="button"
          data-toggle="collapse"
          data-target="#navbarCollapse"
          aria-controls="navbarCollapse"
          aria-expanded="false"
          aria-label="Toggle navigation"
        >
          <span class="navbar-toggler-icon"></span>
        </button>

        <div class="collapse navbar-collapse order-3" id="navbarCollapse">
          <!-- Left navbar links -->
          <ul class="navbar-nav">
            <li class="nav-item">
              <a href="/" class="nav-link">Home</a>
            </li>
            <li class="nav-item">
              <a href="/info" class="nav-link">Info</a>
            </li>
            <li class="nav-item">
              <a href="/test" class="nav-link">Predict</a>
            </li>
          </ul>
        </div>

        <!-- Right navbar links -->
        <ul class="order-1 order-md-3 navbar-nav navbar-no-expand ml-auto">
          <li class="nav-item">
            <button
```

```
          type="button"
          onclick="switchTheme()"
          class="btn btn-primary btn-block btn-sm"
        >
          <i class="fa fa-bell"></i> Switch Theme
        </button>
      </li>
    </ul>
  </div>
</nav>
<!-- /.navbar -->

<!-- Content Wrapper. Contains page content -->
<div class="content-wrapper">
  <!-- Content Header (Page header) -->
  <div class="content-header">
    <div class="container">
      <div class="row mb-2">
        <div class="col-sm-6">
          <h1 class="m-0">Parkinson Detection</h1>
        </div>
        <!-- /.col -->

        <!-- /.col -->
      </div>
      <!-- /.row -->
    </div>
    <!-- /.container-fluid -->
  </div>
  <!-- /.content-header -->

  <!-- Main content -->
  <div class="content">
    <div class="container">
      <div class="row">
        <div class="col-lg-12">
          <div class="card card-primary">
            <div class="card-header">
              <h5 class="card-title m-0">Parkinson Classifier</h5>
            </div>
            <div class="card-body">
              <div class="row">
                <div class="col-md-12">
                  <div class="card card-default">
                    <div class="card-header">
                      <h3 class="card-title">Dropzone</h3>
                    </div>
                    <div class="card-body">
                      <div id="actions" class="row">
                        <div class="col-lg-6">
                          <div class="btn-group w-100">
                            <span
```

```
      class="btn btn-success col fileinput-button"
     >
      <i class="fas fa-plus"></i>
      <span>Add files</span>
     </span>
     <button
      type="submit"
      class="btn btn-primary col start"
     >
      <i class="fas fa-upload"></i>
      <span>Start upload</span>
     </button>
     <button
      type="reset"
      class="btn btn-warning col cancel"
     >
      <i class="fas fa-times-circle"></i>
      <span>Cancel upload</span>
     </button>
    </div>
   </div>
   <div class="col-lg-6 d-flex align-items-center">
    <div class="fileupload-process w-100">
     <div
      id="total-progress"
      class="progress progress-striped active"
      role="progressbar"
      aria-valuemin="0"
      aria-valuemax="100"
      aria-valuenow="0"
     >
      <div
       class="progress-bar progress-bar-success"
       style="width: 0%"
       data-dz-uploadprogress
      ></div>
     </div>
    </div>
   </div>
  </div>
  <div
   class="table table-striped files"
   id="previews"
  >
   <div id="template" class="row mt-2">
    <div class="col-auto">
     <span class="preview"
      ><img src="data:," alt="" data-dz-thumbnail
     /></span>
    </div>
    <div class="col d-flex align-items-center">
     <p class="mb-0">
```

```
          <span class="lead" data-dz-name></span>
          (<span data-dz-size></span>)
        </p>
        <strong
          class="error text-danger"
          data-dz-errormessage
        ></strong>
      </div>
      <div class="col-4 d-flex align-items-center">
        <div
          class="progress progress-striped active w-100"
          role="progressbar"
          aria-valuemin="0"
          aria-valuemax="100"
          aria-valuenow="0"
        >
          <div
            class="progress-bar progress-bar-success"
            style="width: 0%"
            data-dz-uploadprogress
          ></div>
        </div>
      </div>
      <div class="col-auto d-flex align-items-center">
        <div class="btn-group">
          <button class="btn btn-primary start">
            <i class="fas fa-upload"></i>
            <span>Start</span>
          </button>
          <button
            data-dz-remove
            class="btn btn-warning cancel"
          >
            <i class="fas fa-times-circle"></i>
            <span>Cancel</span>
          </button>
          <button
            data-dz-remove
            class="btn btn-danger delete"
          >
            <i class="fas fa-trash"></i>
            <span>Delete</span>
          </button>
        </div>
      </div>
    </div>
  </div>
  <!-- /.card-body -->
  <div class="card-footer">Drop your Image here</div>
</div>
<!-- /.card -->
```

```
        </div>
       </div>
      </div>
     </div>
    </div>
   </div>
  </div>
 </div>
 <!-- /.row -->
 <button
  type="button"
  class="btn btn-danger"
  data-toggle="modal"
  data-target="#modal-danger"
  id="dangerModel"
  hidden
 >
  Launch danger Modal
 </button>
 <div class="modal fade" id="modal-danger">
  <div class="modal-dialog">
   <div class="modal-content bg-danger">
    <div class="modal-header">
     <h4 class="modal-title">
      <b>Parkinson Positive</b>
     </h4>
     <button
      type="button"
      class="close"
      data-dismiss="modal"
      aria-label="Close"
     >
      <span aria-hidden="true">&times;</span>
     </button>
    </div>
    <div class="modal-body">
     <ion-icon
      name="alert-circle-outline"
      size="large"
     ></ion-icon>
     <p>
      <b><i> Consult a doctor</i></b>
      <br />
      You’re going to beat this thing!
     </p>
    </div>
    <div class="modal-footer justify-content-between">
     <button
      type="button"
      class="btn btn-outline-light"
      data-dismiss="modal"
     >
      Close
     </button>
```

```
      </div>
    </div>
    <!-- /.modal-content -->
  </div>
  <!-- /.modal-dialog -->
</div>
<button
  type="button"
  class="btn btn-success"
  data-toggle="modal"
  data-target="#modal-success"
  id="successModel"
  hidden
>
  Launch Success Modal
</button>
<div class="modal fade" id="modal-success">
  <div class="modal-dialog">
    <div class="modal-content bg-success">
      <div class="modal-header">
        <h4 class="modal-title">
          <b>Healthy</b> - Parkinson Negative
        </h4>
        <button
          type="button"
          class="close"
          data-dismiss="modal"
          aria-label="Close"
        >
          <span aria-hidden="true">&times;</span>
        </button>
      </div>
      <div class="modal-body">
        <ion-icon name="happy-outline" size="large"></ion-icon>
        <p>
          The image is drawn by a <b><i>HEALTHY PERSON</i></b>
        </p>
      </div>
      <div class="modal-footer justify-content-between">
        <button
          type="button"
          class="btn btn-outline-light"
          data-dismiss="modal"
        >
          Close
        </button>
      </div>
    </div>
    <!-- /.modal-content -->
  </div>
  <!-- /.modal-dialog -->
</div>
```

```
      </div>
      <!-- /.container-fluid -->
    </div>
    <!-- /.content -->
  </div>
  <!-- /.content-wrapper -->

  <!-- Control Sidebar -->
  <aside class="control-sidebar control-sidebar-dark">
    <!-- Control sidebar content goes here -->
  </aside>
  <!-- /.control-sidebar -->

  <!-- Main Footer -->
  <footer class="main-footer">
    <!-- To the right -->
    <div class="float-right d-none d-sm-inline"></div>
    <!-- Default to the left -->
    <strong
      >Team:  <a
        >Gowtham <S></S></a
      >,  <a
        >Pragatheeswaran</a
      >,  <a
        >Aswath</a
      >,  <a
        >Nisarani</a
      ></strong
    >
  </footer>
</div>
<!-- ./wrapper -->

<!-- REQUIRED SCRIPTS -->

<!-- jQuery -->
<script src="../static/plugins/jquery/jquery.min.js"></script>
<!-- Bootstrap 4 -->
<script src="../static/plugins/bootstrap/js/bootstrap.bundle.min.js"></script>
<!-- AdminLTE App -->
<script src="../static/dist/js/adminlte.min.js"></script>
<!-- AdminLTE for demo purposes -->
<script src="../static/dist/js/demo.js"></script>
<!-- dropzonejs -->
<script src="../static/plugins/dropzone/min/dropzone.min.js"></script>
<script>
  // DropzoneJS Demo Code Start
  Dropzone.autoDiscover = false;

  // Get the template HTML and remove it from the doumenthe template HTML and remove
it from the doument
  var previewNode = document.querySelector("#template");
```

```
    previewNode.id = "";
    var previewTemplate = previewNode.parentNode.innerHTML;
    previewNode.parentNode.removeChild(previewNode);

    var myDropzone = new Dropzone(document.body, {
      // Make the whole body a dropzone
      url: "/predict", // Set the url
      thumbnailWidth: 80,
      thumbnailHeight: 80,
      parallelUploads: 20,
      previewTemplate: previewTemplate,
      autoQueue: false, // Make sure the files aren't queued until manually added
      previewsContainer: "#previews", // Define the container to display the previews
      clickable: ".fileinput-button", // Define the element that should be used as click trigger to
select files.
      success: function (file, response) {
        if (response === "healthy") {
          $("#successModel").click();
        } else {
          $("#dangerModel").click();
        }
      },
    });

    myDropzone.on("addedfile", function (file) {
      // Hookup the start button
      file.previewElement.querySelector(".start").onclick = function () {
        myDropzone.enqueueFile(file);
      };
    });

    // Update the total progress bar
    myDropzone.on("totaluploadprogress", function (progress) {
      document.querySelector("#total-progress .progress-bar").style.width =
        progress + "%";
    });

    myDropzone.on("sending", function (file) {
      // Show the total progress bar when upload starts
      document.querySelector("#total-progress").style.opacity = "1";
      // And disable the start button
      file.previewElement
        .querySelector(".start")
        .setAttribute("disabled", "disabled");
    });

    // Hide the total progress bar when nothing's uploading anymore
    myDropzone.on("queuecomplete", function (progress) {
      document.querySelector("#total-progress").style.opacity = "0";
    });

    // Setup the buttons for all transfers
```

```
  // The "add files" button doesn't need to be setup because the config
  // `clickable` has already been specified.
  document.querySelector("#actions .start").onclick = function () {
    myDropzone.enqueueFiles(myDropzone.getFilesWithStatus(Dropzone.ADDED));
  };
  document.querySelector("#actions .cancel").onclick = function () {
    myDropzone.removeAllFiles(true);
  };

  // DropzoneJS Demo Code End
</script>
<script>
  sessionStorage.setItem("theme", "dark");

  var currentTheme = sessionStorage.getItem("theme");
  var mainHeader = document.querySelector(".main-header");

  if (currentTheme) {
    if (currentTheme === "dark") {
      if (!document.body.classList.contains("dark-mode")) {
        document.body.classList.add("dark-mode");
      }
      if (mainHeader.classList.contains("navbar-light")) {
        mainHeader.classList.add("navbar-dark");
        mainHeader.classList.remove("navbar-light");
      }
    }
  }

  function switchTheme() {
    if (sessionStorage.getItem("theme") === "dark") {
      if (document.body.classList.contains("dark-mode")) {
        document.body.classList.remove("dark-mode");
      }

      sessionStorage.setItem("theme", "light");
    } else {
      if (!document.body.classList.contains("dark-mode")) {
        document.body.classList.add("dark-mode");
      }

      sessionStorage.setItem("theme", "dark");
    }
  }
</script>
<script
  type="module"
  src="https://unpkg.com/ionicons@5.5.2/dist/ionicons/ionicons.esm.js"
></script>
<script
  nomodule
  src="https://unpkg.com/ionicons@5.5.2/dist/ionicons/ionicons.js"
```

```
    ></script>
  </body>
</html>
```

GITHUB LINK :

https://github.com/IBM-EPBL/IBM-Project-8291-1658914296