PROJECT REPORT

Team ID	**PNT2022TMID40374**
Project Name	**DETECTING PARKINSON'S DISEASE USING MACHINE LEARNING**
Team Members	○ **YESWANTH D** ○ **ABIMANYU K** ○ **NIMALRAJ V** ○ **RAJKUMAR P** ○ **SIVAGURUSAKTHIVEL B**

DETECTION OF PARKINSON'S DISEASE USING MACHINE LEARNING

1. INTRODUCTION

A research paper in 2017 by Zham et al explained how to detect parkinson disease without doing any MRI scans on the brain just by detecting the speed and pen pressure of sketching a spiral or a wave. So by this we can automate the process and detect the disease by scanning the pattern of the patient and by using simple computer vision algorithms and machine learning we can train a model to classify between a patient and a healthy person.

a. PROJECT OVERVIEW

Biomarkers derived from human voice can offer in-sight into neurological disorders, such as Parkinson's disease (PD), because of their underlying cognitive and neuromuscular function. PD is a progressive neurodegenerative disorder that affects about one million people in the United States, with approximately sixty thousand new clinical diagnoses made each year . Historically, PD has been difficult to quantity and doctors have tended to focus on symptoms while ignoring others, relying primarily on subjective rating scales. Due to the decrease in motor control that is the hallmark of the disease, voice can be used as a means to detect and diagnose PD

b. PURPOSE

The ML based diagnosis of this subjective disease can be achieved by using symptoms as an attribute for the algorithm.

The ML algorithm is used for better understand and diagnose PD, develop new treatments and ultimately prevent PD.

2. LITERATURE SURVEY

Team	:	PNT2022TMID40374
Team Size	:	5
Team Leader	:	YESWANTH D
Team Member:		ABIMANYU K
Team Member:		NIMALRAJ V
Team Member:		RAJKUMAR P
Team Member:		SIVAGURUSAKTHIVEL B

1	PAPER TITLE	Rahul R Zaveri, Prof. Pramila M. Chawan "The International Research Journal of Engineering and Technology" vol. 07, issue:10, 2020.Dr. Anupam Bhatia and RaunakSulekh [1]
	PROBLEM DEFINITION	In this study, Naive Bayes was applied topredict the performance of the dataset. Rapid miner 7.6.001 is a tool, which was used to explore, statistically analyze, and mine the data.
	METHODOLOGY/ ALGORITHM	“Predictive Modelfor Parkinson’s Disease through Naive Bayes Classification”
	ADVANTAGES	The Naive Bayes modelperforms with 98.5 % accuracy, and 99.75% of precision.
	DISADVANTAGES	The SNN Algorithm is Best for Different classes , but formore accuracy need to be

		used more Best algorithm

2	PAPER TITLE	Carlo Ricciardi, et al "Using gait analysis' parameters to classify Parkinsonism: A data mining approach"
	PROBLEM DEFINITION	In this system, Random Forest is used for classification along with comparing it with Gradient Boosted Trees. These results are being categorized into 3 different categories namely PSP
	METHODOLOGY/ALGORITHM	Random Forest algorithm and Gradient Boosted Tress are used

	ADVANTAGES	De Novo Parkinson's Disease and Stable Parkinson's Diseasewith their accuracybeing as high as 86.4% as compared toGradient Boosted Trees which were accurateto a meagre 70%. Also the precision rate of Random Forest was maximum of 90 % against Gradient Boosted Trees whichwere around maximum of 85%.
	DISADVANTAGES	The accuracy is not necessary to predict the Parkinson's Disease

3	PAPER TITLE	MehrbakhshNilashi et al [3] “A hybrid intelligent system for the prediction ofParkinson’s Disease progression using Machine Learning techniques”
	PROBLEM DEFINITION	It was intended to understand how the different types of preprocessing steps could affect the prediction accuracy of the Classifier
	METHODOLOGY/ALGORITHM	In this system a method was proposed for the UPDRS (Total-UPDRS and Motor-UPDRS)prediction using machine learning. ISVR was used to predict the Total-UPDRS and Motor-UPDRS. SOM and NIPALSwere used for clustering and data dimensionality reduction
	ADVANTAGES	The results show that the method combiningSOM, NIPALS, and ISVR techniques was effective in predicting the Total-UPDRS and Motor-UPDRS.
		The Algorithm used is difficult

	DISADVANTAGES	to predict and edit for future use

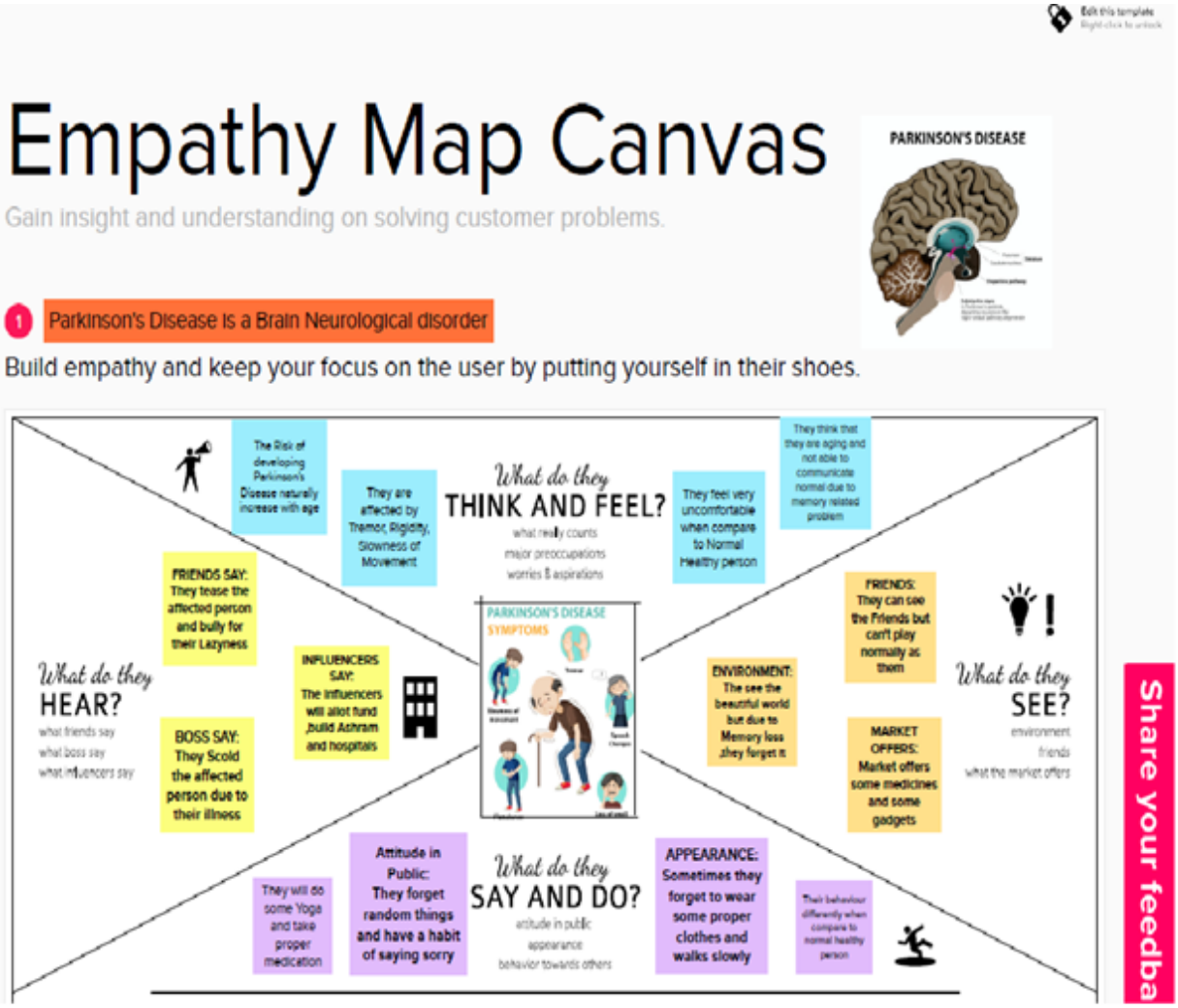
4	PAPER TITLE	Aravind Kumar Tiwari “Machine Learning based Approaches for Predictionof Parkinson’s Disease,”

	PROBLEM DEFINITION	In this system, minimum redundancy maximum relevance feature selection algorithms were used to select the most important feature among all the features to predict Parkinson's disease.
	METHODOLOGY/ALGORITHM	Minimum Redundancy MaximumRelevance Feature Selection Algorithms were used
	ADVANTAGES	This system of feature selection along withRandom Forests provided an accuracy of 90.3% and precision of 90.2%.
	DISADVANTAGES	The accuracy is not neccesaary to predict the Parkinson's Disease more properly

5	PAPER TITLE	DraganaMiljkovicet al [6] "Machine Learning and Data Mining Methods for Managing Parkinson's Disease"
	PROBLEM DEFINITION	In this system, based on the initial patientsexamination and medications taken, the Predictor part was able to predict each Parkinson's Disease symptom separately covering 15 different Parkinson's Disease symptoms in total.
	METHODOLOGY/ALGORITHM	In the process of classifying the Parkinson's Disease dataset using the ANN based MLPclassifier
	ADVANTAGES	The accuracy of prediction ranges from 57.1% to 77.4% depending on the symptom where thehighest accuracy is achieved from tremor detection.
	DISADVANTAGES	Data Mining using those algorithm is very difficult and Accuracy is poor

3. IDEATION & PROPOSED SOLUTION

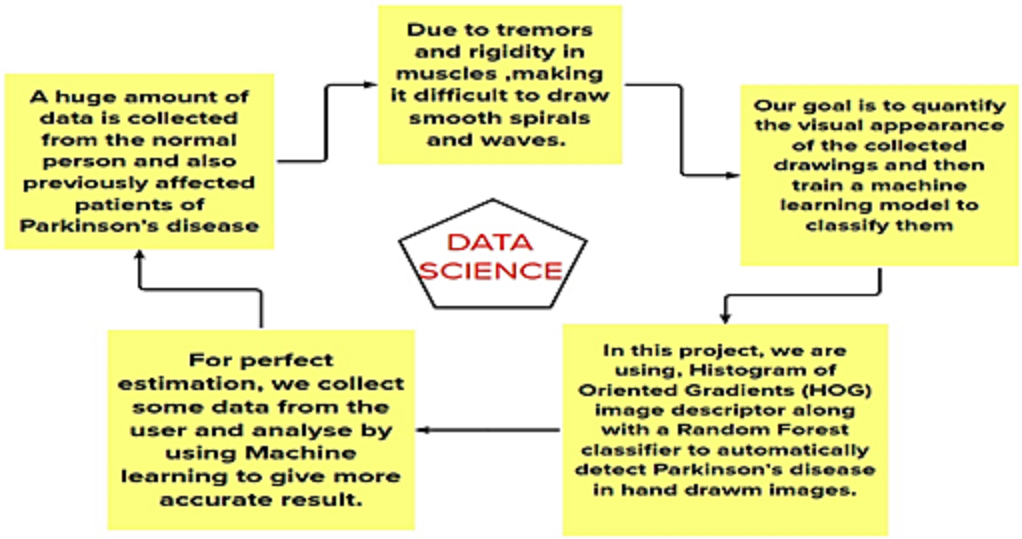
3.1) Empathy Map



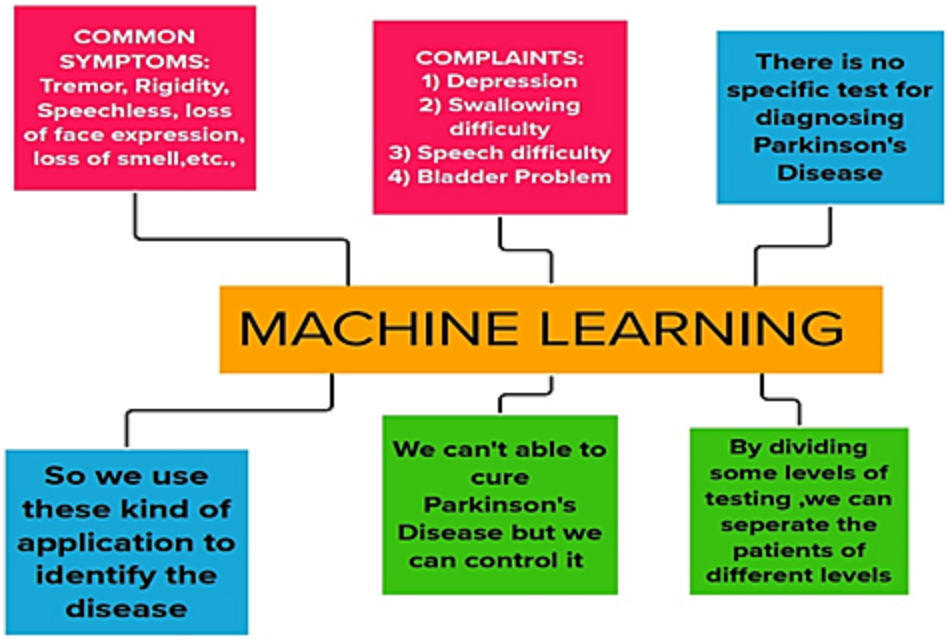
IBM-Project-48457-1660807433/Detecting Parkinson's Disease using Machine Learning.pdf at main · IBM-EPBL/IBM-Project-48457-1660807433 (github.com)

3.2) IDEATION & BRAINSTROMING

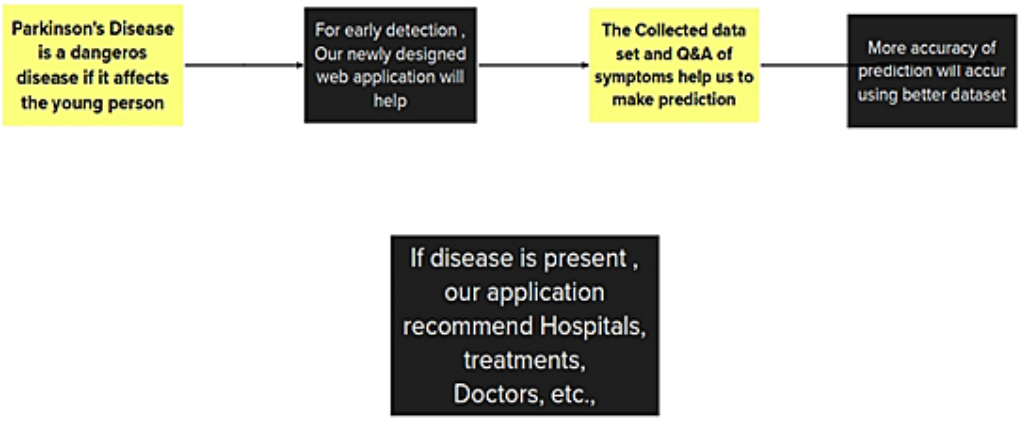
D.YESWANTH :



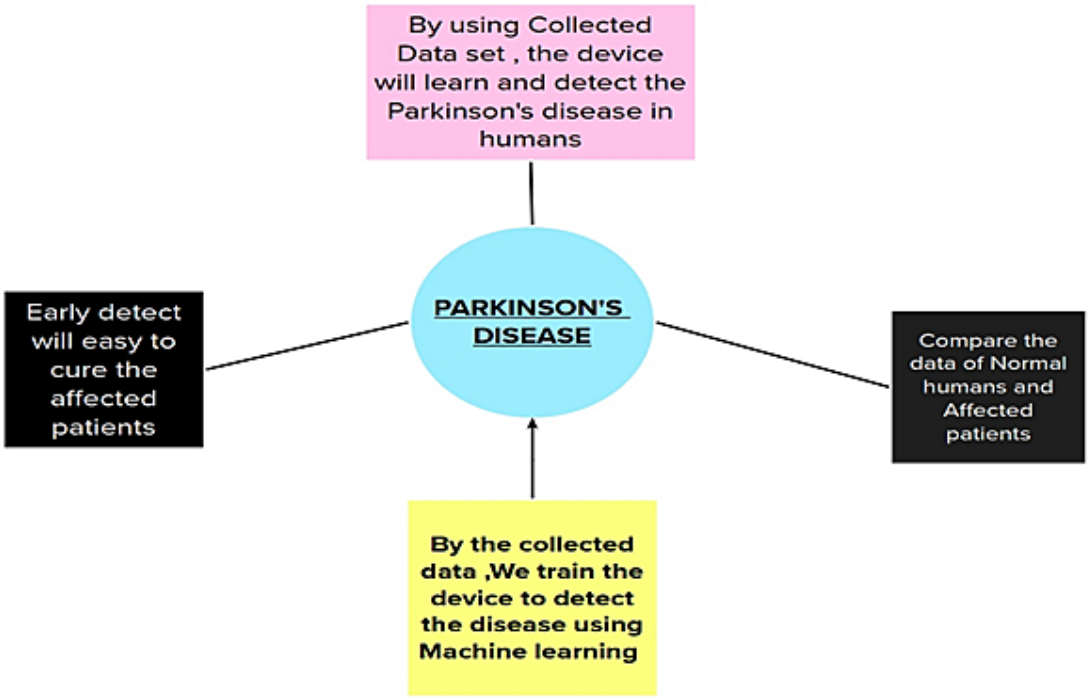
P.RAJKUMAR



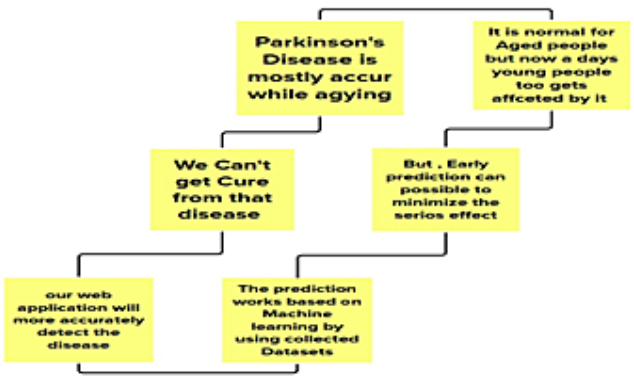
K.ABIMANYU



B.SIVAGURUSAKTHIVEL



V.NIMALRAJ



Parkinson's disease (PD) is a long term degenerative disorder of the central nervous system that mainly affects the motor system.

The symptoms of Parkinson's disease will occur slowly, the symptoms include tremor, shaking, rigidity, slowness of movement and difficulty with walking, thinking and behavior change, depression and anxiety, loss of smell, some visionary problem ,etc

In our model, a huge amount of data is collected from the normal person and also previously affected person by Parkinson's disease , these data is trained using Machine Learning Algorithm.

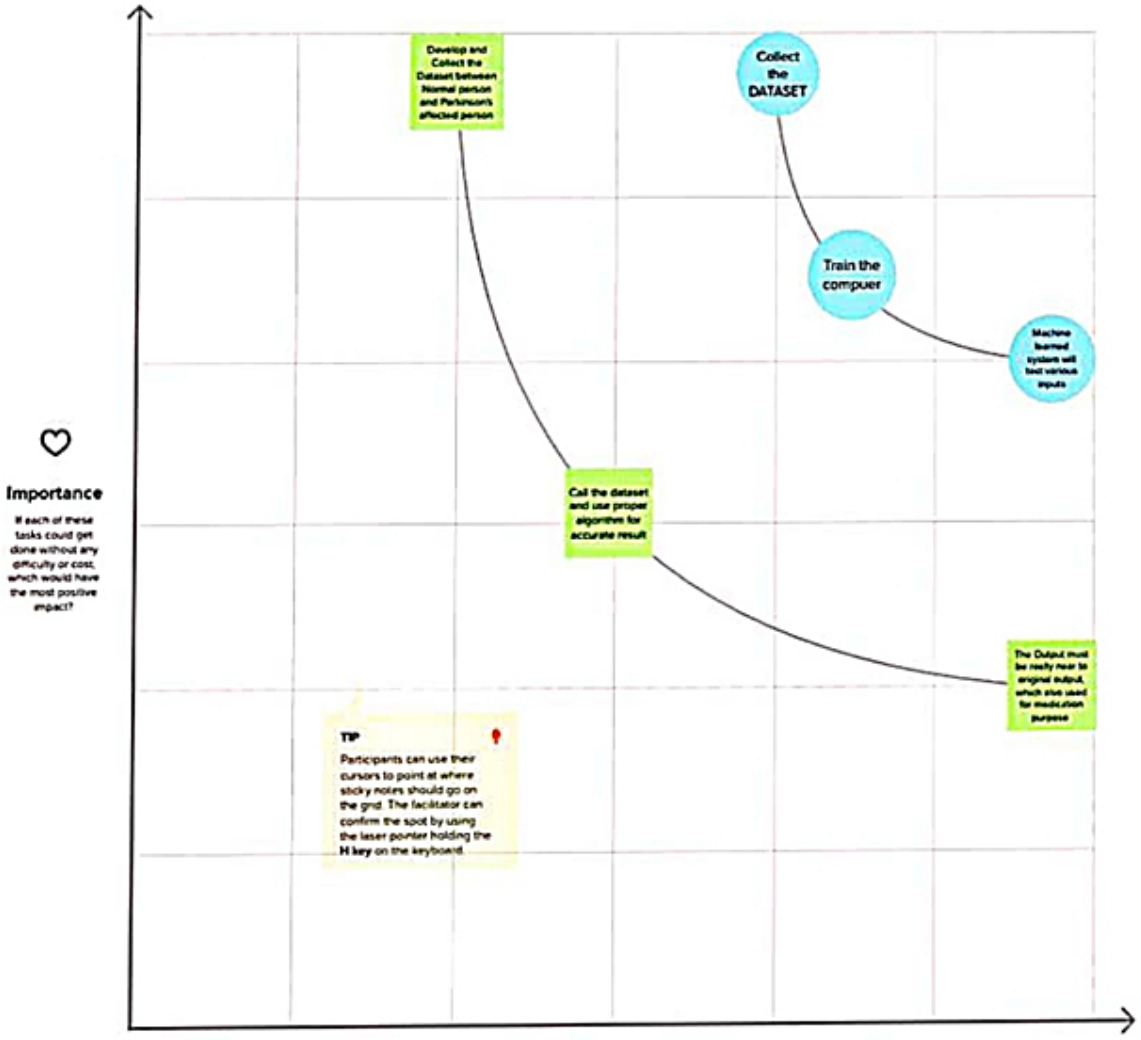
The drawing speed was slower and the pen pressure is lower among Parkinson's patients. Due to tremors and rigidity in muscles ,making it difficult to draw smooth spirals and waves.

Our goal is to quantify the visual appearance 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.

For perfect estimation, we collect some data from the user and analyse by using Machine learning to give more accurate result.
We will build a model using which we can accurately detect the presence of Parkinson's disease in one's body.

The Risk of developing Parkinson's disease naturally increase with age, and the average age at which it starts is 60 years old
Parkinson's disease is chronic and has no cure yet. But, Early detection along with proper medication can significantly improve the quality of life .

Due to improvement in accuracy of detection, user will trust the application and very helpful for Doctors to treat the patients at earlier stage.
To help in early detection of Parkinson's disease which would help in early diagnosis thus slowing down disease progression


Develop and Collect the Dataset between Normal person and Parkinson's affected person
Collect the DATASET
Train the compuer
Machine learned system will test various inputs
Importance
If each of these tasks could get done without any difficulty or cost, which would have the most positive impact?
Call the dataset and use proper algorithm for accurate result
The Output must be really near to original output, which also used for medication purpose
TIP
Participants can use their cursors to point at where sticky notes should go on the grid. The facilitator can confirm the spot by using the laser pointer holding the H key on the keyboard.
Feasibility
Regardless of their importance, which tasks are more feasible than others? (Cost, time, effort, complexity, etc.)

IBM-Project-48457-1660807433/Brainstorm and Ideation (1).pdf at main · IBM-EPBL/IBM-Project-48457-1660807433 (github.com)

3.3) PROPOSED SOLUTION

PROPOSED SOLUTION TEMPLATE

Date	24 September 2022
Team ID	PNT2022TMID40374
Project Name	Detecting Parkinson's Disease using Machine Learning
Maximum Marks	2 Marks

Project team shall fill the following information in proposed solution template.

S.No.	Parameter	Description
1.	Problem Statement (Problem to be solved)	• Parkinson's disease is a progressive disorder of the central nervous system affecting movements and inducing tremors and stiffness • 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.

		● If laterly predicted , it results in cause several threatens to attected persons
2.	Idea / Solution description	● For perfect estimation, we collect some data from the user and analyse by using Machine learning to give more accurate result. ● Using better Machine Learning Algoritm will result in perfect estimation.
3.	Novelty / Uniqueness	● Due to tremors and rigidity in muscles ,making it difficult to draw smooth spirals and waves. ● Our goal is to quantify the visual appearance 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 drawm images. ● For perfect estimation, we collect some data from the user and analyse by using Machine learning to give more accurate result.

4.	Social Impact / Customer Satisfaction	• The main aim of this application is early prediction and proper treatments can possibly stop or slow progression of this disease to end stage. • Hence Customer will not suffer from serious effects of Parkinson's disease
5.	Business Model (Revenue Model)	• Can generate revenue through direct customer. • Can collaborate with health care sector and generate revenue from their customers. • Can make an advertisement of hospitals that have special treatment of Parkinson's Disease, and if patients allow ,we share the details of the patients to that hospital for medication.
6.	Scalability of the Solution	• In order to get user friendly and more accuracy , we additionally add some data collection from user and analyse easily by draw smooth spirals and waves. • An Application is developed where data about patients are recorded and Suggest some medication if they have Parkinson's Disease. • Due to improvement in accuracy of detection, user will trust the application and very helpful for Doctors to treat the patients at earlier stage. • To help in early detection of Parkinson's disease which would help in early diagnosis thus slowing down disease progression.

3.4) PROBLEM SOLUTION FIT

IBM-Project-48457-1660807433/Problem_solution_fit .pdf at main · IBM-EPBL/IBM-Project-48457-1660807433 (github.com)

4. REQUIREMENT ANALYSIS

FUNCTIONAL REQUIREMENT & NON- FUNCTIONAL REQUIREMENTS

Date	12 October 2022
Team ID	PNT2022TMID40374
Project Name	Detecting Parkinson's Disease using Machine Learning
Maximum Marks	4 Marks

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 Registration	Registration through Form Registration through Gmail Registration through Phone Number
FR-2	User Confirmation	Confirmation via Email Confirmation via OTP
FR-3	User permission	Allow the Camera

FR-4	User Details	Get user Name Get user Age Get user Gender Get user Address Get user Location Get user Mobile Number Get user Gmail id
FR-5	User Known value (Accurate/Approximate)	Get user BP level Get user sugar level Get user Physical Health Condition Get user Disorder / Disabilities detail
FR-6	User Motion Capture	The Motion of the User is recorded and the result will be added in Final Overall result

FR-6	User Drawings (Spiral / Circle)	The Drawing pattern and speed was noted and result will be added
FR-7	User Demand	After Getting the Result , the Affected person need to consult a Doctor. The Non-Affected person need some medication which will show in same web application itself

4.2) Non-functional Requirements:

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

FR No.	Non-Functional Requirement	Description

NFR-1	**Usability**	Our Web application is User Friendly which show them demonstration skippable video
NFR-2	**Security**	We don’t share the image and user details
NFR-3	**Reliability**	Our web application have high accuracy which can be utilized multiple times
NFR-4	**Performance**	By using better algorithm and collecting lot of details, Our web application provide more accuracy than all.
NFR-5	**Availability**	Our web application can be used by using Mobile phones itself. It doesn’t need any external material
NFR-6	**Scalability**	Our web applicaton can be used by world wide with easy to access it .

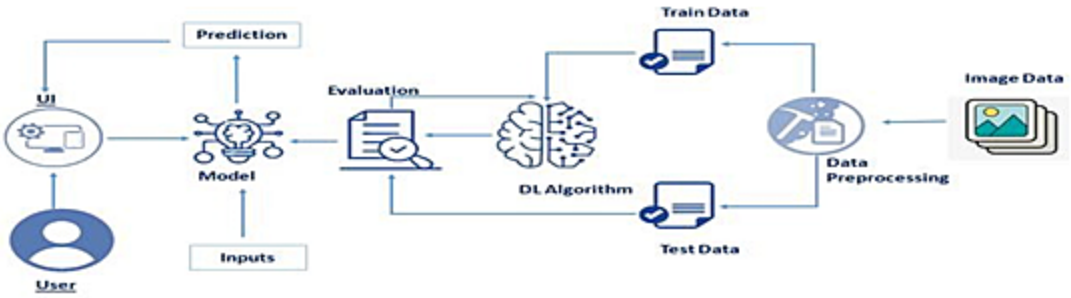
5. **PROJECT DESIGN**

5.1) DATA FLOW DIAGRAMS

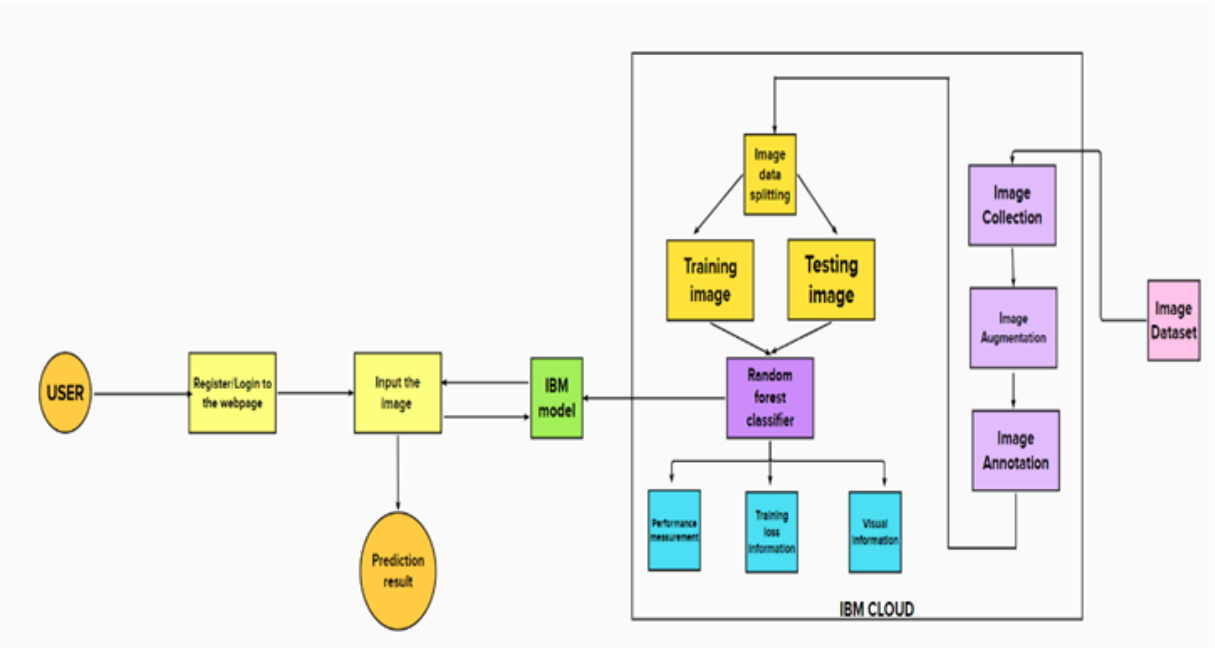
Date	14 October 2022
Team ID	PNT2022TMID40374
Project Name	Detecting Parkinson’s Disease using Machine Learning
Maximum Marks	4 Marks

Data Flow Diagrams:

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



DFD Level 0 (Industry Standard):



5.2) USER STORIES

Use the below template to list all the user stories for the product.

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
Customer (Web user)	Home Page	USN-1	Description about Parkinson's disease.	I can get an idea about the disease.	Low	Sprint-3
		USN-2	Details about the test vitals required for the testing.		Low	Sprint-3
	Registration	USN-3	As a user, I can register for the application by entering my username, email, phone number, and password, and confirming my password.	I can access my account.	Moderate	Sprint-3
		USN-4	As a user, I will receive a confirmation mail once I have registered for the application.	I can receive a confirmation OTP upon registration for verification.	High	Sprint-3
	Login	USN-5	As a user, I can log in to the web application by entering my email id & password.	I can log in successfully.	High	Sprint-2
	Main Page(Test vitals)	USN-6	As a user, I submit the required image for the prediction.	I can access the page and can submit the input.	Moderate	Sprint-4
	Results	USN-7	Results will be displayed along with their accuracy.	I got my results successfully and accurately.	High	Sprint-4
Admin	Data collection	USN-8	Collect the required data for the detection of Parkinson's disease		High	Sprint-1
	Data preprocessing	USN-9	Clean and analyze the data to avoid noise and duplications	As a result I get the desired dataset to get trained.	High	Sprint-1
	Model Building	USN-10	Build the model using a Random forest classifier to classify the images.	Successfully trained the model.	High	Sprint-1
	Deploy the model	USN-11	Deployment of ML model using IBMWatson Studio, object storage.	Deployed successfully.	High	Sprint-2
	Integrate the web app with the IBM Model	USN-12	Use flask for the integration purpose.	Created the web app successfully.	Moderate	Sprint-2

5.3) TECHNOLOGY STACK (ARCHITECTURE & STACK)

Date	13 October 2022
Team ID	PNT2022TMID40374
Project Name	Detecting Parkinson's Disease Machine Learning
Maximum Marks	4 Marks

Technical Architecture:

The Deliverable shall include the architectural diagram as below and the information as per the table1 & table 2

Example: Order processing during pandemics for offline mode

Reference: https://link.springer.com/chapter/10.1007/978-981-33-6912-2_22

Guidelines:

1. Get the Image from the user (From Storage / Live Drawing)
2. Use HOG or Deep features method for better classification
3. Use Several best Algorithm techniques like KNN , SVM , Random Forest, Multilayer Perception , Native Bayes etc., to get accurate result
4. By Testing and get compared with trained result it predict and detect the Diagram Pattern
5. Finally the Machine Learning will predict with more efficiency

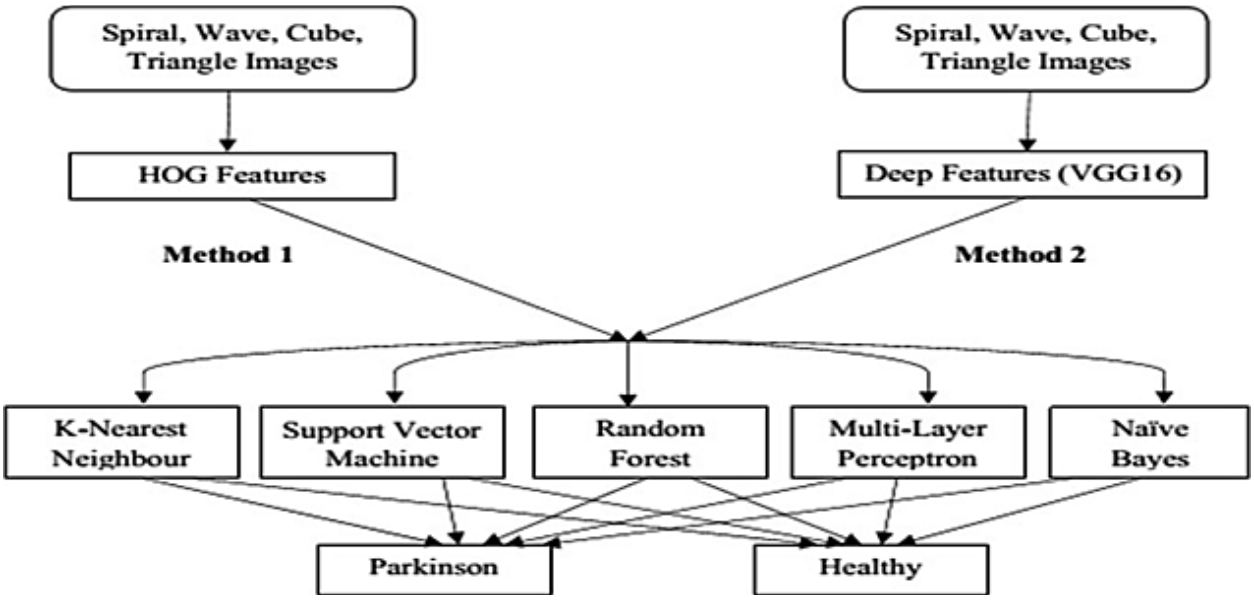


TABLE-1 : COMPONENTS & TECHNOLOGIES:

S.No	Component	Description	Technology
1.	User Interface	How user interacts with application e.g. Web UI, Mobile App, Chatbot etc.	HTML, CSS, JavaScript / Angular Js / React Js etc.
2.	Application Logic-1	Logic for a process in the application	Java / Python
3.	Application Logic-2	Logic for a process in the application	IBM Watson STT service
4.	Application Logic-3	Logic for a process in the application	IBM Watson Assistant
5.	Database	Data Type, Configurations etc.	MySQL, NoSQL, etc.
6.	Cloud Database	Database Service on Cloud	IBM DB2, IBM Cloudant etc.
7.	File Storage	File storage requirements	IBM Block Storage or Other Storage Service or Local

			Filesystem
8.	External API-1	Purpose of External API used in the application	IBM Weather API, etc.
9.	External API-2	Purpose of External API used in the application	Aadhar API, etc.
10.	Machine Learning Model	Purpose of Machine Learning Model	Object Recognition Model, etc.
11.	Infrastructure (Server / Cloud)	Application Deployment on Local System / Cloud Local Server Configuration: Cloud Server Configuration :	Local, Cloud Foundry, Kubernetes, etc.

Table-2: Application Characteristics:

S.No	Characteristics	Description	Technology
1.	Open-Source Frameworks	List the open-source frameworks used	Technology of Opensource framework
2.	Security Implementations	List all the security / access controls implemented, use of firewalls etc.	e.g. SHA-256, Encryptions, IAM Controls, OWASP etc.
3.	Scalable Architecture	Justify the scalability of architecture (3 – tier, Micro-services)	Technology used
4.	Availability	Justify the availability of application (e.g. use of load balancers, distributed servers etc.)	Technology used
5.	Performance	Design consideration for the performance of The application (number of requests per sec, use of Cache, use of CDN's) etc.	Technology used

References:

1. DraganaMiljkovic et al, "Machine Learning and Data Mining Methods for Managing Parkinson's Disease" LNAI 9605, pp 209-220, 2016.
2. Arvind Kumar Tiwari, "Machine Learning based Approaches for Prediction of Parkinson's Disease," Machine Learning and Applications- An International Journal (MLAU) vol. 3, June 2016.

3. Dr. Anupam Bhatiaand RaunakSulekh, “Predictive Model for Parkinson’s Disease through Naive Bayes Classification” International Journal of Computer Science & Communication vol. 9, March 2018..
4. M. Abdar and M. Zomorodi-Moghadam, “Impact of Patients’ Genderon Parkinson’s diseaseusing Classification Algorithms” Journal of AI and Data Mining,vol.6, 2018.
5. Md. Redone Hassan et al, “A Knowledge Base Data Mining based on Parkinson’s Disease” International Conference on System Modelling& Advancement in Research Trends, 2019

6. PROJECT PLANNING & SCHEDULING

Project Planning Phase

Project Planning Template (Product Backlog, Sprint Planning, Stories, Storypoints)

Date	21 October 2022
Team ID	PNT2022TMID40374
Project Name	Detecting Parkinson’s Disease Using Machine Learning
Maximum Marks	8 Marks

6.1) Sprint Planning & Estimation

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-1	Registration	USN-1	As a user, I can register for the web application by entering my email, password, and confirming my password.	2	High	Yeswanth D Sivagurusakthivel B
Sprint-1	Registration	USN-2	As a user, I will receive confirmation email once I have registered for the application	1	High	Rajkumar P Nimalraj V
Sprint-2	Registration	USN-3	As a user, I can register for the application through Facebook	2	Low	Abimanyu K
Sprint-2	Registration	USN-4	As a user, I can register for the application through	2	Low	Sivagurusakthivel B Rajkumar P

			Google Account			
Sprint-1	Registration	USN-5	As a user, I can register for the application through Gmail	2	Medium	Abimanyu K Nimalraj V
Sprint-1	Login	USN-6	As a user, I can log into the application by entering email & password	1	High	Yeswanth D
Sprint-3	Dashboard	USN-7	As a user, I can see the description about the option in web application	2	Low	Abimanyu K Nimalraj V
Sprint-3	Dashboard	USN-8	As a user, I can provide the necessary details and answer the question in that website honestly	2	High	Yeswanth D Sivagurusakthivel B Rajkumar P
Sprint-3	Dashboard	USN-9	As a user, I upload the drawn images and get the desired result along with the accuracy	2	High	Yeswanth D Sivagurusakthivel B
Sprint-4	Dashboard	USN-10	As a user, I can able to see the Nearby located Hospitals, and some medication advices	1	Low	Rajkumar P Nimalraj V Abimanyu K
Sprint-4	Management	USN-11	As a Administrator, I will update our web application with additional features.	2	Low	Yeswanth D Sivagurusakthivel B
Sprint-4	Management	USN-12	As a Administrator, I can maintain third party Services	1	High	Rajkumar P Nimalraj V Abimanyu K

Sprint	**Functional Requirement (Epic)**	**User Story Number**	**User Story / Task**	**Story Points**	**Priority**	**Team Members**

Sprint-4	Management	USN-13	As a Administrator, I will collect new dataset and keep our model well trained to avoid noise and duplication.	2	High	Yeswanth D Sivagurusakthivel B Rajkumar P
Sprint-2	Management	USN-14	As a Administrator, I can maintain the information about the user and provide privacy to them	2	High	Abimanyu K Nimalraj V

6.2) Sprint Delivery Schedule:

Sprint	**Total Story Points**	**Duration**	**Sprint Start Date**	**Sprint End Date (Planned)**	**Story Points Completed (as on Planned End Date)**	**Sprint Release Date (Actual)**
Sprint-1	6	6 Days	24 Oct 2022	29 Oct 2022	6	29 Oct 2022
Sprint-2	6	6 Days	31 Oct 2022	05 Nov 2022	6	05 Nov 2022
Sprint-3	6	6 Days	07 Nov 2022	12 Nov 2022	5	12 Nov 2022
Sprint-4	6	6 Days	14 Nov 2022	19 Nov 2022	6	19 Nov 2022

Velocity:

Imagine we have a 10-day sprint duration, and the velocity of the team is 20 (points per sprint). Let's calculate the team's average velocity (AV) per iteration unit (story points per day)

AV (Sprint 1) = 6/6 = 1

AV (Sprint 2) = 6/6 = 1

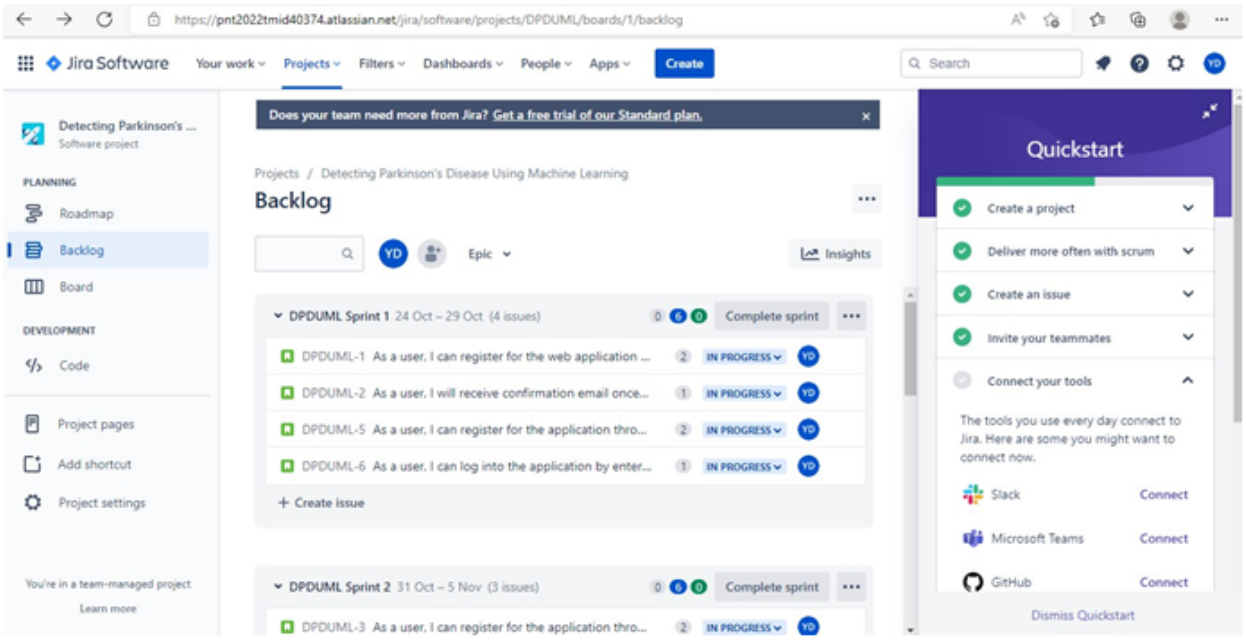
AV (Sprint 3) = 6/6 = 1

AV (Sprint 4) = 6/6 = 1

AV (Total) = 24/24 = 1

Burndown Chart:

A burn down chart is a graphical representation of work left to do versus time. It is often used in agile software development methodologies such as Scrum. However, burn down charts can be applied to any project containing measurable progress over time.



https://www.visual-paradigm.com/scrum/scrum-burndown-chart/

https://www.atlassian.com/agile/tutorials/burndown-charts

REFERENCE

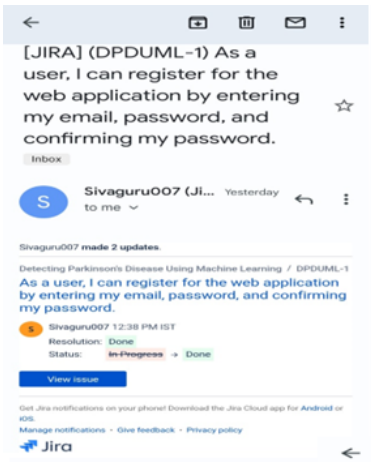
https://www.atlassian.com/agile/project-managementhttps://www.atlassian.com/agile/tutorials/how-to-do-scrum-with-jira-software https://www.atlassian.com/agile/tutorials/epics

https://www.atlassian.com/agile/tutorials/sprints

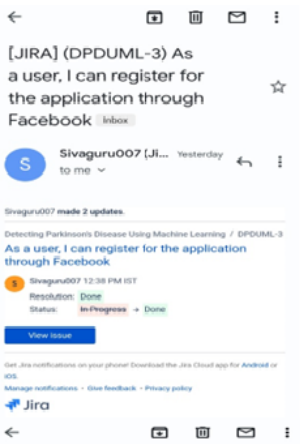
https://www.atlassian.com/agile/project-management/estimation

https://www.atlassian.com/agile/tutorials/burndown-charts

6.3) REPORTS FROM JIRA


[JIRA] (DPDUML-1) As a user, I can register for the web application by entering my email, password, and confirming my password.
Inbox
Sivaguru007 (Ji... Yesterday
to me
Sivaguru007 made 2 updates.
Detecting Parkinson's Disease Using Machine Learning / DPDUML-1
As a user, I can register for the web application by entering my email, password, and confirming my password.
Sivaguru007 12:38 PM IST
Resolution: Done
Status: In Progress → Done
View issue
Get Jira notifications on your phone! Download the Jira Cloud app for Android or iOS.
Manage notifications · Give feedback · Privacy policy
Jira


[JIRA] (DPDUML-2) As a user, I will receive confirmation email once I have registered for the application
Inbox
Sivaguru007 (Ji... Yesterday
to me
Sivaguru007 made 2 updates.
Detecting Parkinson's Disease Using Machine Learning / DPDUML-2
As a user, I will receive confirmation email once I have registered for the application
Sivaguru007 12:38 PM IST
Resolution: Done
Status: In Progress → Done
View issue
Get Jira notifications on your phone! Download the Jira Cloud app for Android or iOS.
Manage notifications · Give feedback · Privacy policy
Jira


[JIRA] (DPDUML-3) As a user, I can register for the application through Facebook
Inbox
Sivaguru007 (Ji... Yesterday
to me
Sivaguru007 made 2 updates.
Detecting Parkinson's Disease Using Machine Learning / DPDUML-3
As a user, I can register for the application through Facebook
Sivaguru007 12:38 PM IST
Resolution: Done
Status: In Progress → Done
View issue
Get Jira notifications on your phone! Download the Jira Cloud app for Android or iOS.
Manage notifications · Give feedback · Privacy policy
Jira

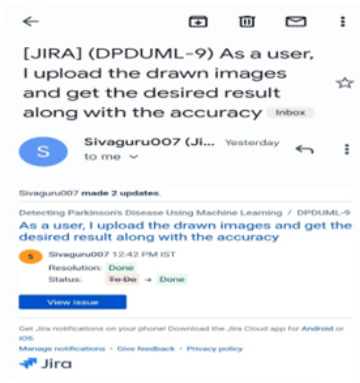

[JIRA] (DPDUML-4) As a user, I can register for the application through Google Account
Inbox
Sivaguru007 (Ji... Yesterday
to me
Sivaguru007 made 2 updates.
Detecting Parkinson's Disease Using Machine Learning / DPDUML-4
As a user, I can register for the application through Google Account
Sivaguru007 12:38 PM IST
Resolution: Done
Status: In Progress → Done
View issue
Get Jira notifications on your phone! Download the Jira Cloud app for Android or iOS.
Manage notifications · Give feedback · Privacy policy
Jira


[JIRA] (DPDUML-5) As a user, I can register for the application through Gmail
Inbox
Sivaguru007 (Ji... Yesterday
to me
Sivaguru007 made 2 updates.
Detecting Parkinson's Disease Using Machine Learning / DPDUML-5
As a user, I can register for the application through Gmail
Sivaguru007 12:38 PM IST
Resolution: Done
Status: In Progress → Done
View issue
Get Jira notifications on your phone! Download the Jira Cloud app for Android or iOS.
Manage notifications · Give feedback · Privacy policy
Jira

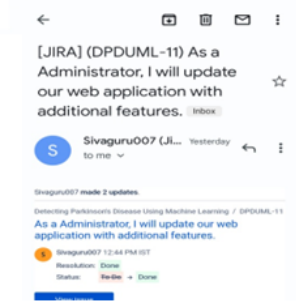

[JIRA] (DPDUML-6) As a user, I can log into the application by entering email & password
Inbox
Sivaguru007 (Ji... Yesterday
to me
Sivaguru007 made 2 updates.
Detecting Parkinson's Disease Using Machine Learning / DPDUML-6
As a user, I can log into the application by entering email & password
Sivaguru007 12:38 PM IST
Resolution: Done
Status: In Progress → Done
View issue
Get Jira notifications on your phone! Download the Jira Cloud app for Android or iOS.
Manage notifications · Give feedback · Privacy policy
Jira

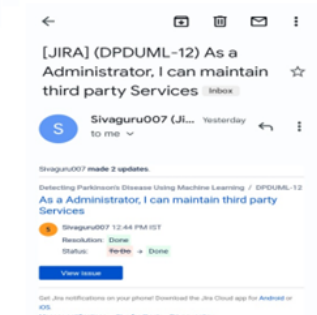

[JIRA] (DPDUML-7) As a user, I can see the description about the option in web application
Inbox
Sivaguru007 (Ji... Yesterday
to me
Sivaguru007 made 2 updates.
Detecting Parkinson's Disease Using Machine Learning / DPDUML-7
As a user, I can see the description about the option in web application
Sivaguru007 12:42 PM IST
Resolution: Done
Status: To Do → Done
View issue
Get Jira notifications on your phone! Download the Jira Cloud app for Android or iOS.
Manage notifications · Give feedback · Privacy policy
Jira


[JIRA] (DPDUML-8) As a user, I can provide the necessary details and answer the question in that website honestly
Inbox
Sivaguru007 (Ji... Yesterday
to me
Sivaguru007 made 2 updates.
Detecting Parkinson's Disease Using Machine Learning / DPDUML-8
As a user, I can provide the necessary details and answer the question in that website honestly
Sivaguru007 12:42 PM IST
Resolution: Done
Status: To Do → Done
View issue
Get Jira notifications on your phone! Download the Jira Cloud app for Android or iOS.
Manage notifications · Give feedback · Privacy policy
Jira


[JIRA] (DPDUML-9) As a user, I upload the drawn images and get the desired result along with the accuracy
Inbox
Sivaguru007 (Ji... Yesterday
to me
Sivaguru007 made 2 updates.
Detecting Parkinson's Disease Using Machine Learning / DPDUML-9
As a user, I upload the drawn images and get the desired result along with the accuracy
Sivaguru007 12:42 PM IST
Resolution: Done
Status: To Do → Done
View issue
Get Jira notifications on your phone! Download the Jira Cloud app for Android or iOS.
Manage notifications · Give feedback · Privacy policy
Jira


[JIRA] (DPDUML-10) As a user, I can able to see the Nearby located Hospitals, and some medication advices
Inbox
Sivaguru007 (Ji... Yesterday
to me
Sivaguru007 made 2 updates.
Detecting Parkinson's Disease Using Machine Learning / DPDUML-10
As a user, I can able to see the Nearby located Hospitals, and some medication advices
Sivaguru007 12:44 PM IST
Resolution: Done
Status: To Do → Done


[JIRA] (DPDUML-11) As a Administrator, I will update our web application with additional features.
Inbox
Sivaguru007 (Ji... Yesterday
to me
Sivaguru007 made 2 updates.
Detecting Parkinson's Disease Using Machine Learning / DPDUML-11
As a Administrator, I will update our web application with additional features.
Sivaguru007 12:44 PM IST
Resolution: Done
Status: To Do → Done


[JIRA] (DPDUML-12) As a Administrator, I can maintain third party Services
Inbox
Sivaguru007 (Ji... Yesterday
to me
Sivaguru007 made 2 updates.
Detecting Parkinson's Disease Using Machine Learning / DPDUML-12
As a Administrator, I can maintain third party Services
Sivaguru007 12:44 PM IST
Resolution: Done
Status: To Do → Done
View issue
Get Jira notifications on your phone! Download the Jira Cloud app for Android or iOS.
Manage notifications · Give feedback · Privacy policy

7. CODING & SOLUTION

7.1)FEATURES 1

Importing The Necessary Libraries

The first step is usually importing the libraries that will be needed in the program.

The required libraries to be imported to Python script are:

OpenCV:

OpenCV is a library of programming functions mainly aimed at real-time computer vision. Here, OpenCV is used for resizing. Rescaling, thresholding the image.

Imutils:

Imutils package has a series of convenience functions to make basic image processing functions such as translation, rotation, resizing, and displaying Matplotlib images and video frames easier with OpenCV.

We will build_montages for visualization. Our paths import will help us to extract the file paths to each of the images in our dataset.

sklearn.metrics:
The module implements several loss, score, and utility functions to measure classification performance.

sklearn.preprocessing:
This package provides several common utility functions and transformer classes to change raw feature vectors into a representation that is more

suitable for the downstream estimators.

Sklearn.ensemble
This package contains RandomForestClassifier and many more inbuilt algorithms.

Scikit-image

Scikit-image, or skimage, is an open-source Python package designed for image preprocessing.

Histogram of Oriented Gradients (HOG) will come from the feature import of scikit-image.

Pickle:
Python pickle module is used for serializing and de-serializing python object structures. The process to converts any kind of python objects (list, dict, etc.) into byte streams.

7.2) FEATURES 2

- Once after splitting the data into train and test, the data should be fed to an algorithm to build a model.
- There are several Machine learning algorithms to be used depending on the data you are going to process such as images, sound, text, and numerical values. The algorithms that you can choose according to the objective that you might have it may be Classification algorithms are Regression algorithms.

1.Logistic Regression

2.Decision Tree Classifier

3.Random Forest Classifier

4.KNN

Random Forest classifier

Initialize our **Random Forest classifier** and train the model using a number of estimators as 100
After training the model, the model should be tested by using the test data which is been separated while splitting the data for checking the functionality of the model.

Here we are selecting 25 images from the test data and initialize the output images for montage

we're going to create a montage so that we can share our work visually

- First, we randomly sample images from our testing set
- Our images list will hold each spiral image along with annotations added via OpenCV drawing functions .
- We proceed to loop over the random image indices.
- Inside the loop, each image is processed in the same manner as during training(convert to gray scale, resize, threshold) .
- From there we'll automatically classify the image using our new HOG + Random Forest based classifier and add color-coded annotation
- Each image is quantified with HOG features.
- Then the image is classified bypassing those features to model.predict .
- The class label is colored green for "healthy" and red otherwise .The label is drawn in the top left corner of the image using cv2.putText function .

- Each output image is then appended to an images list so that we can develop a montage

- The montage is then displayed until a key is pressed

The **cv2.imshow()** function always takes two more functions to load and close the image. **cv2.waitKey()** function, you can provide any value to close the image and continue with further lines of code.
Evaluation is a process during the development of the model to check whether the model is the best fit for the given problem and corresponding data.

Classification Evaluation Metrics:

These model evaluation techniques are used to find out the accuracy of models built in the classification type of machine learning models. We have three types of evaluation methods.

- Accuracy_score
- Confusion matrix
- Roc- Auc Curve

Confusion Matrix
It is a matrix representation of the results of any binary testing.

Fig: Confusion Matrix of prediction of a disease

1. True Positive: 12 (You have predicted the positive case correctly!)
2. True Negative: 77 (You have predicted negative case correctly!)
3. False Positive: 8 (You have predicted these people as having disease, but in actual they do not have.)

4. False Negative: 3 (Wrong predictions)

We can use the **predict** method on the model and pass **X_test** as a parameter to get the output as predictions.

8. TESTING

-

8.1) Test Case

Section	Total Cases	Not Tested	Fail	Pass
Print Engine	7	0	0	7
Client Application	195	0	0	195
Security	0	0	0	0
Outsource Shipping	0	0	0	0
Exception Reporting	9	0	0	9
Final Report Output	4	0	0	4
Version Control	2	0	0	2

8.2) User Acceptance Testing

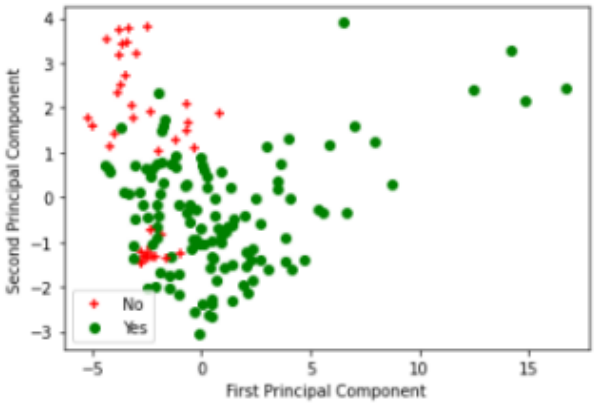
Resolution	Severity 1	Severity 2	Severity 3	Severity 4	Subtotal
By Design	5	3	2	5	15
Duplicate	1	0	3	0	4
External	2	3	0	1	6
Fixed	11	2	4	20	37
Not Reproduced	0	0	0	0	0
Skipped	0	0	0	0	0
Won't Fix	0	5	2	0	7
Totals	19	13	11	25	69

9. RESULTS

9.1) Performance Metrics

PROJECT : Detecting Parkinson’s Disease Using Machine Learning

TEAM ID : PNT2022TMID40374



In [ ]:

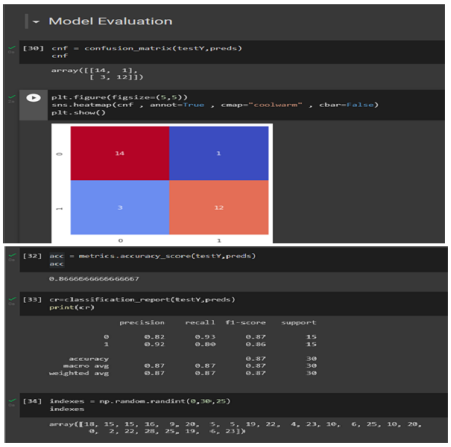
```
input_data = (197.07600,206.89600,192.05500,0.00289,0.00001,0.00166,0.00168,0.00498,0.01098,0.09700,0.00563,0.00680,0.00802,0.01689,0.00339,26.77500,0

# changing input data to a numpy array
input_data_as_numpy_array = np.asarray(input_data)

# reshape the numpy array
input_data_reshaped = input_data_as_numpy_array.reshape(1,-1)

# standardize the data
std_data = scaler.transform(input_data_reshaped)

prediction = model.predict(std_data)
print(prediction)


if (prediction[0] == 0):
  print("The Person does not have Parkinsons Disease")

else:
  print("The Person has Parkinsons")
```

```
[0]
The Person does not have Parkinsons Disease
```

S.No.	Parameter	Values	Screenshot
1.	Metrics	**Classification Model:** Confusion Matrix, Accuracy Score & Classification Report	
2.	Tune the Model	Hyperparameter Tuning - GridSearchCV	

BRIEF DETAILED SCREENSHOTS:

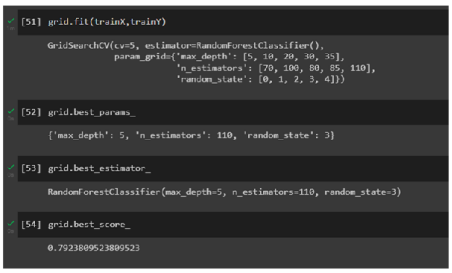
METRICS:



```
Model Evaluation

[30] cnf = confusion_matrix(testY,preds)
     cnf

     array([[14,  1],
            [ 3, 12]])

     plt.figure(figsize=(5,5))
     sns.heatmap(cnf , annot=True , cmap="coolwarm" , cbar=False)
     plt.show()

[32] acc = metrics.accuracy_score(testY,preds)
     acc

     0.8666666666666667

[33] cr=classification_report(testY,preds)
     print(cr)

                  precision    recall  f1-score   support

               0       0.82      0.93      0.87        15
               1       0.92      0.80      0.86        15

        accuracy                           0.87        30
       macro avg       0.87      0.87      0.87        30
    weighted avg       0.87      0.87      0.87        30

[34] indexes = np.random.randint(0,30,25)
     indexes

     array([18, 15, 15, 16,  9, 20,  5,  5, 19, 22,  4, 23, 10,  6, 25, 10, 20,
             0,  2, 22, 28, 25, 19,  6, 21])
```

TUNE THE MODEL:

```
Tuning the model

[47] from sklearn.model_selection import GridSearchCV

[48] model=RandomForestClassifier()

[49] parameters={
         'max_depth': [5,10,20,30,35],
         'random_state':[0,1,2,3,4],
         'n_estimators':[20,100,80,85,110]
     }

[50] grid=GridSearchCV(model,parameters,cv=5)

[51] grid.fit(trainX,trainY)
```



```
[51] grid.fit(trainX,trainY)

     GridSearchCV(cv=5, estimator=RandomForestClassifier(),
                  param_grid={'max_depth': [5, 10, 20, 30, 35],
                              'n_estimators': [70, 100, 80, 85, 110],
                              'random_state': [0, 1, 2, 3, 4]})

[52] grid.best_params_

     {'max_depth': 5, 'n_estimators': 110, 'random_state': 3}

[53] grid.best_estimator_

     RandomForestClassifier(max_depth=5, n_estimators=110, random_state=3)

[54] grid.best_score_

     0.7923809523809523
```

10. ADVANTAGES AND DISADVANTAGES

The Advantage is that the RandomForest model to detect parkinson disease from the images was developed successfully with an accuracy of 88%. This model is now used to classify the images, as healthy or parkinson, uploaded in a web page which is integrated with Flask.

The Disadvantages is that the Predict may sometime rarely mismatch with original Disease symptoms.

Due to some comman symptoms of other disease, it may overcome by updating the dataset and trained itself.

11. CONCLUSION

a. Parkinson's disease is a progressive disorder of the central nervous system affecting movements and inducing tremors and stiffness.
b. According to Parkinson's Foundation, more than 10 million people are living with Parkinson's disease.
c. The Risk of developing Parkinson's disease naturally increase with age, and the average age at which it starts is 60 years old
d. Parkinson's disease is chronic and has no cure yet. But, Early detection along with proper medication can significantly improve the quality of life
e. As soon as early predicted will reduce the risk factor of life.

12. FUTURE SCOPE

Machine Learning is the Future of our World . It Simplify the testing , and save money and Time. It will replace the MRI scan and CT scan etc.,

The Test is done at home itself

Every smart gadgets user will predominantly prefer this type of application

13. APPENDIX

Algorithm Used

I've used HOG (Histogram of oriented gradients) to extract features from the dataset and then passed these features to a random forest classifier to train the model on classifying patterns of patients and healthy drawings.

Software Designing

- Jupyter Notebook Environment
- Machine Learning Algorithms
- Python
- Visual studio code
- HTML
- Flask

We developed this Image detection by using the Python language, which is a high level programming language along with Machine Learning Algorithm such as Random Forest Classifier. For coding we used the Jupyter Notebook of Anaconda distributions and Visual Studio Code in python language. Flask is used as a user interface for the prediction. Hypertext Markup Language (*HTML*) is the standard markup language for documents designed to be

displayed in a web browser.

SOURCE CODE:

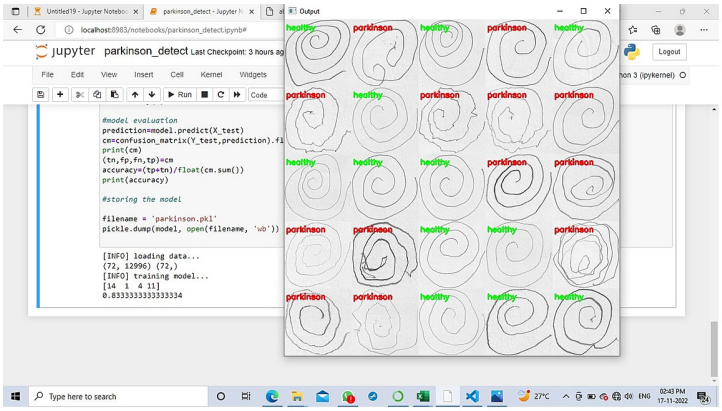
```
#import the pakages

from sklearn.ensemble import RandomForestClassifier
from sklearn.preprocessing import LabelEncoder
from sklearn.metrics import confusion_matrix
from skimage import feature
from imutils import build_montages
from imutils import paths
import numpy as np
import cv2
import os
import pickle

#Quatifying image
def quantify_image(image):
    #compute histogram of oriented gradients feature vector for
the input image

features=feature.hog(image,orientations=9,pixels_per_cell=(10,10
),cells_per_block=(2,2),transform_sqrt=True,block_norm="L1")
    return features

def load_split(path):
    #grab list of images in the input dir,then initialize the
list of data and class labels

    imagepaths=list(paths.list_images(path))
    data,labels=[],[]

    #loop over the image path
    for imagepath in imagepaths:
        #extract the class label from the filename
        label=imagepath.split(os.path.sep)[-2]

        #load the input image
        image=cv2.imread(imagepath)
```

```
        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
        features=quantify_image(image)

        #update the data and labels
        data.append(features)
        labels.append(label)

    return (np.array(data),np.array(labels))

# define path to train and test dir

trainingpath= r"Desktop/dataset/spiral/training"
testingpath=r"Desktop/dataset/spiral/testing"

#loading train and test data

print("[INFO] loading data...")
(X_train,Y_train)=load_split(trainingpath)
(X_test,Y_test)=load_split(testingpath)



#Label Encoding
le=LabelEncoder()
Y_train=le.fit_transform(Y_train)
Y_test=le.transform(Y_test)
print(X_train.shape,Y_train.shape)

#Training The Model

print("[INFO] training model...")
model=RandomForestClassifier(n_estimators=100)
model.fit(X_train,Y_train)

#testing the model
testingpath=list(paths.list_images(testingpath))
```

```
idxs=np.arange(0,len(testingpath))
idxs=np.random.choice(idxs,size=(25,),replace=False)
images=[]

#loop over the testing samples
for i in idxs:
    image=cv2.imread(testingpath[i])
    output=image.copy()

    # load the input image,convert 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 feature using last trained random forest
    features=quantify_image(image)
    preds=model.predict([features])
    label=le.inverse_transform(preds)[0]
    #the set of output images
    if label=="healthy":
        color=(0,255,0)
    else:
        color=(0,0,255)


cv2.putText(output,label,(3,20),cv2.FONT_HERSHEY_SIMPLEX,0.5,col
or,2)
    images.append(output)

#creating a montage
montage=build_montages(images,(128,128),(5,5))[0]
cv2.imshow("Output",montage)
cv2.waitKey(0)

#model evaluation
prediction=model.predict(X_test)
cm=confusion_matrix(Y_test,prediction).flatten()
```

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

#storing the model

filename = 'parkinson.pkl'
pickle.dump(model, open(filename, 'wb'))
[INFO] loading data...
(72, 12996) (72,)
[INFO] training model...
[14  1  4 11]
0.8333333333333334
```

IMAGE PREPROCESSING OUTPUT:



BASE HTML CODES :

```
<html lang="en">

  <head>
```

```
    <meta charset="UTF-8" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0"
/>
    <meta http-equiv="X-UA-Compatible" content="ie=edge" />
    <title>Predict</title>
    <link
      href="https://cdn.bootcss.com/bootstrap/4.0.0/css/bootstrap.min.css"
      rel="stylesheet"
    />
    <script
src="https://cdn.bootcss.com/popper.js/1.12.9/umd/popper.min.js"></script>
    <script src="https://cdn.bootcss.com/jquery/3.3.1/jquery.min.js"></script>
    <script
src="https://cdn.bootcss.com/bootstrap/4.0.0/js/bootstrap.min.js"></script>
    <link
      href="{{ url_for('static', filename='css/main.css') }}"
      rel="stylesheet"
    />
    <style>
      body {
        background-image: url("https://img.freepik.com/free-vector/clean-
```

```
medical-patterned-background-vector_53876-
140867.jpg?w=1060&t=st=1667911964~exp=1667912564~hmac=4298568
f384f42cfc60423d63ac6a8c806e4fe025c1bed2f32ae68b3f15b2139");
    background-position: center;
    background-repeat: no-repeat;
    background-size: cover;
    height: 100%;
    width: 100%;
  }
  h1 {
    font-size: 40px;
    text-align: center;
    color: black;
    font-style: italic;
    font-weight: bolder;
  }
  h2 {
    font-size: 35px;
    text-align: center;
    color: black;
    font-style: italic;
```

```
  font-weight: bolder;
}
h5 {
  font-size: 25px;
  text-align: center;
  color: black;
  font-weight: bolder;
}

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

a:hover {
  background-color: black;
```

```
  color: white;
  font-size: 30px;
  padding-left: 10px;
  border-radius: 5px;
}

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

```
    }

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

```
      <li class="active"><a href="/upload">Predict-Results</a></li>
    </ul>
  </nav>
  <br />
  <h1><b>Prevention is better than cure!</b></h1>
  <br />
  <h2>
    <center>
      ❤Diagnosis is not the end, but the beginning of practice.
    </center>
  </h2>
  <br />
  <h2><center>❤Detect the disease and take measures wisely</center></h2>
  <br />
  <h5>
    NOTE: Upload an spiral drawn by the patient for better Prediction /user in a white
    sheet
  </h5>
```

```
    <div class="container">
      <center>
        <div id="content" style="margin-top: 2em">
          {% block content %}{% endblock %}
        </div>
      </center>
    </div>
  </body>

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

HOME HTML CODES

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

```
h3 {
  text-align: center;
  color: white;
}
.main {
  margin-top: 100px;
}
p {
  color: black;
  text-indent: 10px;
  margin: 10px;
  font-size: 20px;
}

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

```
  padding-right: 20px;
}

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

ul {
  align-items: center;
  display: flex;
  list-style-type: none;
  width: 100%;
  gap: 3rem;
  justify-content: center;
  font-size: 2rem;
```

```
  position: fixed;
  top: 0;
  margin: 0;
  padding: 1rem;
  background-color: white;
}

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

```
    img {
      width: 450px;
      height: 400px;
      padding: 25px;
    }
    img:hover {
      border-color: grey;
    }
    #im {
      width: 1450px;
      height: 700px;
      padding: 25px;
    }
  </style>
</head>
<body>
  <nav>
    <ul>
      <li class="active"><a href="/home">Home</a></li>
```

```
      <li class="active"><a href="/upload">Predict-Results</a></li>
    </ul>
  </nav>
  <br /><br /><br />
  <h1>
    <center>
      <b class="pd"
        ><font color="black" size="15" font-family="Comic Sans MS"
          >Detection of Parkinson's Disease using ML</font
        ></b
      >
    </center>
  </h1>
  <div>
    <center>
      <p style="text-align: left">
        Parkinson disease (PD) is a progressive neuro degenerative disorder
        that impacts more than 6 million people around the world.
Parkinson's
        disease is non-communicable, early-stage detection of Parkinson's
```

can

prevent further damages in humans suffering from it.

However,Nonetheless, non-specialist physicians still do not have a definitive test for PD, similarly in the early stage of the diseased person where the signs may be intermittent and badly characterized. It resulted in a high rate of misdiagnosis (up to 25% among non-specialists) and many years before treatment, patients can have the disorder. A more accurate, unbiased means of early detection is required, preferably one that individuals can use in their home setting.However, it has been observed that PD's presence in a human is related to its hand-writing as well as hand-drawn subjects. From that perspective, several techniques have been proposed by researchers to detect Parkinson's disease from hand-drawn images of suspected people.

But the previous methods have their constraints.

</p>

<h4>

<b class="pd"
><font color="black" size="12" font-family="Comic Sans MS"
>Causes and Symptoms of Parkinson's Disease

</h4>

<img
src="https://www.narayanahealth.org/blog/wp-content/uploads/2015/04/parkinson.png"
title="Disease"
/>

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

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

```
/></span>
<span
  ><img
    src="https://jnnp.bmj.com/content/jnnp/91/8/795/F4.large.jpg"
    title="diagnosis"
/></span>
<h3>
  <center>
    <font color="black" size="12" font-family="Comic Sans MS"
      >Treatment for parkinson disease</font
    >
  </center>
</h3>
<span
  ><img
    src="https://www.mdpi.com/biomolecules/biomolecules-11-
00612/article_deploy/html/images/biomolecules-11-00612-g001.png"
    title="diagnosis"
/></span>
<span
```

```
        ><img

          src="https://media.springernature.com/m685/springer-
static/image/art%3A10.1038%2Fs41401-020-0365-
y/MediaObjects/41401_2020_365_Fig1_HTML.png"

          title="diagnosis"

       /></span>

       <span

         ><img

          src="https://www.verywellhealth.com/thmb/BgjmOKb2W-
7z0gqLZryKBd4FFHs=/1500x0/filters:no_upscale():max_bytes(150000):stri
p_icc()/advanced-parkinsons-disease-5200544_color_text_v1-
3bc74418259340ceaf5f6d407daeff73.jpg"

          title="diagnosis"

       /></span>


       <h3>

         <center>

          <font color="black" size="12" font-family="Comic Sans MS"

            >How brains looks during PD?</font

          >

         </center>
```

```
      </h3>


      <span
        ><img
          id="im"

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

PRED HTML CODES

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


<div>
    <form id="upload-file" method="post" enctype="multipart/form-data">
        <center>
            <label for="imageUpload" class="upload-label">
                Choose...
            </label>
            <input type="file" name="file" id="imageUpload" accept=".png, .jpg,
.jpeg">
        </center>
    </form>


  <center> <div class="image-section" style="display:none;">
        <div class="img-preview">
            <div id="imagePreview">
            </div></center>
        </div>
        <center>
```

```
            <div>
                <button type="button" class="btn btn-primary btn-lg " id="btn-predict">Predict!</button>
            </div>
        </center>
    </div>

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

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

</div>

{% endblock %}
```

main.css CODE :

```
.img-preview {
    width: 256px;
    height: 256px;
```

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

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

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

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

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

.loader {
    border: 8px solid #f3f3f3;
    /* Light grey */
    border-top: 8px solid #3498db;
```

```
    /* Blue */
    border-radius: 50%;
    width: 50px;
    height: 50px;
    animation: spin 1s linear infinite;
}

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

main.js CODE :

```
$(document).ready(function() {
    // Init
    $('.image-section').hide();
```

```
    $('.loader').hide();
    $('#result').hide();

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

```
    $('#result').hide();
    readURL(this);
});

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

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

    // Make prediction by calling api /predict
    $.ajax({
        type: 'POST',
        url: '/predict',
        data: form_data,
        contentType: false,
        cache: false,
```

```
            processData: false,
            async: true,
            success: function(data) {
                // Get and display the result
                $('.loader').hide();
                $('#result').fadeIn(600);
                $('#result').text('Prediction : ' + data);
                console.log('Success!');
            },
        });
    });


});
```

GITHUB LINK : https://github.com/IBM-EPBL/IBM-Project-48457-1660807433

PROJECT DEMO LINK : Parkinson Disease Video.mp4 - Google Drive

PROJECT DEMO LINK IN GITHUB :

IBM-Project-48457-1660807433/Parkinson Disease Video.mp4 at main · IBM-EPBL/IBM-Project-48457-1660807433 (github.com) **(WITHOUT VOICE)**

PROJECT DEMO LINK :
https://drive.google.com/file/d/1AbImdGJX_ebiokaemzgdaiJGemCol

H1q/view?usp=share_link (WITH VOICE)