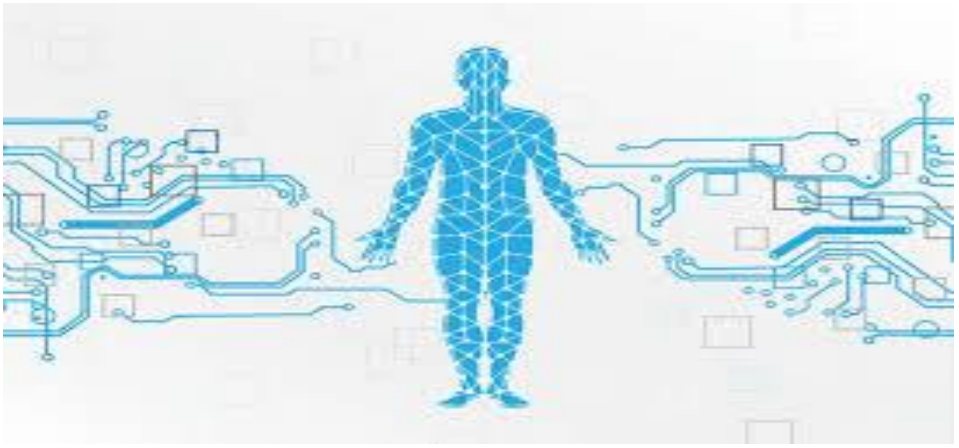
Disease Prediction System for Parkinson Disease using Voice Command



[PROJECT REPORT]

Ashutosh Kumar Yadav	**504121011006**
Mujahid Ali Ansari	**504121011025**
Shahil Kumar Chourasia	**504121011045**
Sirsha Majumder	**504121021050**
Sonu Routh	**504121011051**

in partial fulfillment for the award of the degree

of

MASTER OF COMPUTER APPLICATIONS

IN

BRANCH OF COMPUTER APPLICATIONS

GURUNANAK INSTITUTE OF TECHNOLOGY, KOLKATA

[MAY, 2023]

TABLE OF CONTENTS

CERTIFICATE OF APPROVAL

This is to certify that the major project entitled "Disease Prediction System for Parkinson Disease and Breast Cancer using Voice Command" submitted towards the partial fulfilment of MCA 4th Semester, Department of Computer Application of Guru Nanak Institute of Technology(GNIT), was carried out by Mr. Mujahid Ali Ansari (Roll No - 504121011025), Mr. Shahil Kumar Chourasia (Roll No - 504121011045), Ms. Sirsha Majumder (Roll No - 504121021050), Mr. Sonu Routh (Roll No - 504121011051), Mr. Ashutosh Kumar Yadav (Roll No - 504121011006), under the supervision of Prof. **Dr. Ananjan Maity** , Asst. Prof. Dept. of Computer Application, Guru Nanak Institute of Technology. The matter embodied in this project work is genuine and motivating for future work.

Wish them all success.

(Mr. Chiranjib Dutta)

Head, Asst. Prof.

Dept. of Computer Applications

Guru Nanak Institute of Technology.

(Mr. Dr. Ananjan Maity)

Project Guide, Asst. Prof.

Dept. of Computer Applications

Guru Nanak Institute of Technology.

(Mrs. Dola Saha)

Project Coordinator, Asst. Prof.,

Dept. of Computer Application,

Guru Nanak Institute of Technology.

External Examiner

ACKNOWLEDGEMENT

We, all the team members, would like to take this opportunity to express our profound gratitude and kind regards to our guide **Dr. Ananjan Maity** (Assistant Professor – CA) for his exemplary guidance, monitoring, and constant encouragement throughout this project work. The blessings, help and guidance given by him to us shall carry us to a long way in our journeys of life on which we are about to embark.

Signature of Students **Signature of Mentor**

ABSTRACT

This project introduces a System that can be used to predict if a patient is affected by **Parkinson Disease**. Even today, this diseases is very serious health issues that are persisting.

This system is quite ready to be used by the administration of hospitals as well – the staffs can use it to register new patients and predict if they have symptoms of Parkinson disease, send the patients' health reports to their respective mail ids, retrieve health reports of existing patients in database and to discharge patients. Having all these registrations, predictions, etc. done before getting checked by the main doctor cuts off time consumption and makes check-ups easy for doctors as well.

Also, this system can be used by the Staff Head (admin) to access the data of all the working staffs from the database along with all the patients' details and delete staff data if required. The admin only would have a very confidential password for this special access.

The add-on feature of this system is that, we **can use voice commands** to get the actions of the system done. This system is available to be used 24X7 by staffs and admin.

OVERVIEW

Giovanni Costantini [5] This study aims to develop a voice-based Parkinson's disease diagnosis system using artificial intelligence (AI) techniques. Parkinson's disease is a neurological disorder that affects a person's movement, causing tremors, stiffness, and difficulty with coordination. Early detection of Parkinson's disease is important for effective treatment and management of the condition. However, traditional diagnosis methods can be time-consuming and have limitations in accuracy. The voice-based diagnosis system developed in this study uses machine learning algorithms to analyse speech data collected from individuals with Parkinson's disease. The dataset was pre-processed and feature extraction techniques were used to extract relevant features for diagnosis. A machine learning model was trained and tested using the collected data to evaluate the accuracy of the diagnosis system. The results of this study indicate that the voice-based Parkinson's disease diagnosis system achieved high accuracy in identifying individuals with Parkinson's disease. The system showed promise in providing an early, non-invasive, and low-cost diagnosis method for Parkinson's disease. This study highlights the potential of AI and voice-based diagnosis techniques in healthcare and opens up new opportunities for early detection and management of Parkinson's disease.

INTRODUCTION

- Brief explanation of Parkinson's Disease

G. J. Canter[6] Parkinson disease is a neurological disorder that affects human movements. The condition occurs due to the degeneration of dopamine-producing neurons in a specific area of the brain called the substantia nigra. Dopamine is a neurotransmitter that plays a critical role in regulating movement, motivation, and reward. The loss of dopamine-producing neurons leads to a reduction in dopamine levels in the brain, resulting in the hallmark symptoms of Parkinson's disease. The primary symptoms of Parkinson's disease include tremors, rigidity, bradykinesia (slowness of movement), and postural instability. Other non-motor symptoms, such as depression, anxiety, sleep disturbances, and cognitive impairment, may also occur. Although there is currently no cure for Parkinson's disease, treatments such as medication, surgery, and therapy can help manage symptoms and improve quality of life for those affected by the condition.

- The importance of early diagnosis of Parkinson’s disease

Z. Karapinar Senturk[7] Early diagnosis is critical in the treatment and management of many medical conditions, including Parkinson's disease. Early diagnosis of Parkinson's disease can lead to early intervention and treatment, which can help improve symptoms, slow the progression of the disease, and improve quality of life.

1. E. Waubant,[20] Early Intervention and Treatment: Early diagnosis of Parkinson's disease can lead to early intervention and treatment, which can help improve symptoms, slow the progression of the disease, and improve quality of life. Medication, surgery, physical therapy, and occupational therapy are all treatment options for Parkinson's disease, and early diagnosis can help people with Parkinson's disease start these treatments sooner.

2. D. Mellick,[21] Improved Quality of Life: Parkinson's disease can significantly impact a person's quality of life, and early diagnosis and treatment can help people with Parkinson's disease maintain their quality of life and continue to participate in daily activities. Early intervention and treatment can help improve mobility, reduce muscle stiffness, and improve balance, making it easier for people with Parkinson's disease to perform daily tasks and remain independent.

3. Better Management of Symptoms: Parkinson's disease is a progressive disease, and symptoms can worsen over time. Medication and physical therapy can help reduce tremors,

stiffness, and balance problems, and occupational therapy can help people with Parkinson's disease learn new ways to perform daily tasks.

4. E. R. Dorsey *et al.*,[22] Increased Access to Clinical Trials: Early diagnosis of Parkinson's disease can also increase access to clinical trials. By participating in clinical trials, people with Parkinson's disease can access new treatments and therapies that may not yet be widely available.

5. W. Wang, J. Lee, F. Harrou[8] Improved Mental Health: Parkinson's disease can also impact a person's mental health, leading to depression, anxiety, and other emotional issues. Early diagnosis and treatment can help manage these symptoms, improve mental health, and improve overall quality of life.

6. Family Planning: Parkinson's disease can have a genetic component, and early diagnosis can help families plan for the future. Genetic testing can help determine if family members are at risk for developing Parkinson's disease, and early diagnosis can lead to early intervention and treatment, which can help reduce the risk of developing symptoms.

In conclusion, early diagnosis is critical in the treatment and management of many medical conditions, including Parkinson's disease. If you or someone you know is experiencing symptoms of Parkinson's disease, it is important to seek medical attention as soon as possible.

- The potential of AI and voice-based diagnosis

Artificial intelligence (AI) and voice-based diagnosis have shown tremendous potential in the early detection and diagnosis of a wide range of medical conditions, including Parkinson's disease.

AI and Parkinson's Disease Diagnosis:

I. Nissar, D. Rizvi, S. Masood[9] AI-based algorithms have shown promising results in detecting and diagnosing Parkinson's disease using a wide range of data sources, including voice samples, gait analysis, and digital biomarkers. Voice-based diagnosis is a non-invasive and cost-effective approach that has shown particular promise in early diagnosis of Parkinson's disease.

Studies have shown that people with Parkinson's disease often exhibit changes in their voice, such as decreased loudness, monotone pitch, and altered speech rate. These changes are often subtle and difficult to detect, even by experienced healthcare professionals. AI-based algorithms have been developed that can analyze voice samples and identify these changes, providing a reliable and accurate tool for early diagnosis of Parkinson's disease.

G. Solana-Lavalle[23] Voice-based diagnosis is particularly useful for remote monitoring of Parkinson's disease patients, as it enables healthcare professionals to monitor patients' symptoms and disease progression without the need for frequent in-person appointments. This approach can be genuinely beneficial during pandemic situations like the Covid-19, as it can

help reduce the risk of exposure to the virus for both patients and healthcare professionals.

AI and voice-based diagnosis offer several benefits for Parkinson's disease diagnosis, including:

1.G. Costantini *et al.*, [24] Early Detection and Diagnosis: AI and voice-based diagnosis can help detect Parkinson's disease at an early stage, which can lead to early intervention and treatment. Early intervention can help slow the progression of the disease and improve quality of life for people with Parkinson's disease.

2. S. Marziyeh Ghoreshi Beyrami [25] Non-invasive and Cost-effective: Voice-based diagnosis is a non-invasive and cost-effective approach that can be easily implemented in a wide range of healthcare settings. This approach can help reduce the cost and burden of Parkinson's disease diagnosis and monitoring, making it more accessible to a wider range of patients.

3. D. Gatsios *et al.*,[26] Remote Monitoring: AI and voice-based diagnosis enable remote monitoring of Parkinson's disease patients, which can help improve patient outcomes and reduce the burden on healthcare systems. This approach can help reduce the need for frequent in-person appointments, making healthcare more convenient and accessible for patients.

4. Personalized Treatment: AI-based algorithms can analyze large amounts of data to provide personalized treatment recommendations for people with Parkinson's disease. This approach can help healthcare professionals tailor treatment plans to the individual needs of each patient, improving treatment outcomes and quality of life.

5. W. G. Meissner *et al.*,[27] Improved Research: AI and voice-based diagnosis can also improve research into Parkinson's disease by providing large amounts of data for analysis. This approach can help researchers identify new biomarkers, treatments, and therapies for Parkinson's disease, leading to better treatment options for patients.

Conclusion:

In conclusion, AI and voice-based diagnosis offer tremendous potential for early detection and diagnosis of Parkinson's disease. Voice-based diagnosis is a non-invasive and cost-effective approach that can be easily implemented in a wide range of healthcare settings, enabling remote monitoring of patients and personalized treatment recommendations. AI-based algorithms can analyze large amounts of data to provide accurate and reliable diagnosis, improving treatment outcomes and quality of life for people with Parkinson's disease. As research in this area continues to advance, AI and voice-based diagnosis will likely play an increasingly important role in the early detection and management of Parkinson's disease.

The Science behind Parkinson's Disease Diagnosis

- Traditional diagnosis methods of Parkinson's Disease

J. Mei, C. Desrosiers, and J. Frasnelli,[12] There are several traditional diagnosis methods that can be used to diagnose Parkinson's disease. There are some common methods of diagnosis:

1. Clinical assessment: A clinical assessment involves a physical examination and neurological evaluation by a trained healthcare professional, such as a neurologist. The assessment may include tests of motor function, such as assessing gait, balance, and hand coordination, as well as assessments of non-motor symptoms, such as mood, cognition, and sleep.

2. C. G. Goetz[28] Medical history: Taking a detailed medical history is an important component of Parkinson's disease diagnosis. A healthcare professional may ask about symptoms, family history of Parkinson's disease, and exposure to environmental toxins or medications that may increase the risk of Parkinson's disease.

4. J. Santiago and J. Potashkin[47], Blood tests: While there is no specific blood test for Parkinson's disease, blood tests can be used to rule out other conditions that may cause similar symptoms, such as thyroid disorders or vitamin deficiencies.

5. M. D. Hssayeni, [29] Response to medication: Parkinson's disease symptoms often respond well to medications that increase dopamine levels in the brain. A positive response to medication, such as levodopa, can be a diagnostic indicator of Parkinson's disease.

It is important to note that no single diagnostic test can definitively diagnose Parkinson's disease. A combination of clinical assessments, medical history, imaging studies, and response to medication is typically used to make a diagnosis. Additionally, early diagnosis and treatment are important for managing symptoms and improving quality of life.

- Symptoms of Parkinson's Disease

Symptoms: The symptoms of Parkinson's disease usually develop gradually and worsen over time. The primary symptoms of Parkinson's disease are motor symptoms, which include:

1. S. Sveinbjornsdottir[10] Tremors: Tremors or shaking in the hands, arms, legs, jaw, or face is one of the most recognizable symptoms of Parkinson's disease.
2. Bradykinesia: Slowness of movement or difficulty initiating movement is another common

symptom of Parkinson's disease. It can make simple tasks like getting dressed or brushing teeth more challenging.
3. Rigidity: Stiffness in the arms, legs, or torso can make movement uncomfortable and painful.
4. Postural instability: Difficulty maintaining balance and coordination, which can lead to falls.

Other symptoms of Parkinson's disease may include:

W. C. Koller[11] 1. Difficulty in swallowing: Parkinson's disease can cause difficulty in swallowing, which can lead to choking or aspiration pneumonia.
2. Loss of sense of smell: Many people with Parkinson's disease experience a loss of sense of smell.
3. S. Ray and P. Agarwal[46], Depression and anxiety: Parkinson's disease can lead to depression and anxiety, which can worsen the symptoms of the disease.
Cure: Currently, there is no cure for Parkinson's disease, but several treatments can help manage symptoms and improve quality of life. The primary treatment for Parkinson's disease is medication, which works by increasing dopamine levels in the brain. The most commonly used medication is levodopa, which is converted into dopamine in the brain. Other medications used to treat Parkinson's disease include dopamine agonists, which mimic the action of dopamine in the brain, and MAO-B inhibitors, which help prevent the breakdown of dopamine.
Physical therapy can also be helpful for people with Parkinson's disease. Physical therapy can help improve balance, reduce muscle stiffness, and increase mobility. Occupational therapy can also be helpful in teaching people with Parkinson's disease how to perform daily tasks more easily.
In conclusion, Parkinson's disease is a chronic and progressive neurological disorder that primarily affects movement. The exact cause of Parkinson's disease is not fully understood, but it is believed to be caused by a combination of genetic and environmental factors. Early diagnosis and treatment can help people with Parkinson's disease maintain their quality of life and continue to participate in daily activities.

- Limitations of traditional diagnosis methods of Parkinson Disease

S. E. Lacy[13] While traditional diagnosis methods for Parkinson's disease, such as clinical assessments and imaging studies, can be effective, they have some limitations that can affect their accuracy and reliability. Here are some of the limitations of traditional diagnosis methods:

Subjectivity: Clinical assessments rely on the subjective observations of clinicians and may be influenced by their experience and training. This can lead to variability in diagnoses and may affect the accuracy of diagnosis.

A. Schrag[48], Late diagnosis: Traditional diagnosis methods may not be able to detect Parkinson's disease in its early stages, when symptoms may be mild or non-specific. This can

lead to delayed diagnosis and treatment, which can impact the effectiveness of treatment and quality of life.

Cost: Some traditional diagnosis methods, such as imaging studies, can be expensive and may not be accessible to all patients. This can limit their usefulness in certain populations.

Invasive: Some diagnostic tests, such as lumbar puncture, require invasive procedures that can be uncomfortable or risky for patients.

D. Aarsland,[30] Limited information: Traditional diagnostic methods may not provide a complete picture of the underlying pathology of Parkinson's disease, which can limit the ability to develop personalized treatment plans.

To overcome these limitations, researchers are exploring new diagnostic tools and technologies, such as biomarker tests, wearable sensors, and artificial intelligence algorithms, that may offer more objective, accurate, and non-invasive methods of diagnosis. These technologies have the potential to improve early detection, personalize treatment, and improve patient outcomes in Parkinson's disease and other neurological disorders.

The Role of Artificial Intelligence in Parkinson's Disease Diagnosis

- Advantages of AI diagnosis of Parkinson's Disease

M. Belić, V. Bobić, [14] Artificial intelligence (AI) has the potential to revolutionize the diagnosis of Parkinson's disease by offering several advantages over traditional diagnostic methods. Here are some of the advantages of AI diagnosis:

1. Objective and standardized: AI algorithms can provide an objective and standardized analysis of diagnostic data, eliminating the subjectivity and variability that can be present in traditional diagnostic methods.

2. Early detection: AI algorithms can detect Parkinson's disease at an earlier stage when symptoms may be subtle or non-specific, allowing for earlier intervention and better treatment outcomes.

3. Non-invasive: Many AI-based diagnostic tools, such as voice-based analysis or wearable sensors, are non-invasive and can be easily integrated into routine clinical care without requiring uncomfortable or invasive procedures.

4. Cost-effective: Some AI-based diagnostic tools are relatively inexpensive compared to traditional diagnostic methods such as imaging studies, making them accessible to a wider range of patients.

5. T. Stoddard-Bennett[49], Personalized treatment: AI algorithms can provide more detailed information about the underlying pathology of Parkinson's disease, allowing for more personalized treatment plans tailored to the individual patient's needs.

6. Remote monitoring: AI-based diagnostic tools can enable remote monitoring of patients, allowing for more frequent and timely assessments of symptoms, which can improve disease management and reduce hospitalizations.

Overall, AI-based diagnostic tools have the potential to significantly improve the accuracy, efficiency, and accessibility of Parkinson's disease diagnosis, leading to earlier intervention, better treatment outcomes, and improved quality of life for patients.

- Machine Learning algorithms used in Parkinson's Disease diagnosis

I. Mandal and N. Sairam,[15] Machine learning algorithms are increasingly being used to improve the accuracy and efficiency of Parkinson's disease diagnosis. Here are some of the machine learning algorithms commonly used in Parkinson's disease diagnosis:

1. M. Shahbakhi[31] Support Vector Machines (SVMs): SVMs are supervised learning algorithms that can be used to classify patients with Parkinson's disease based on input data. SVMs work by finding the optimal hyperplane that separates data points into different classes.

2. R. Rone Sarra, [32] Artificial Neural Networks (ANNs): ANNs are a type of machine learning algorithm that mimic the structure and function of the human brain. ANNs can be used to analyze complex data sets and identify patterns that may not be apparent to humans.

3. T. P. Exarchos *et al.*, [33] Decision Trees: Decision trees are a type of machine learning algorithm that uses a tree-like structure to model decisions and their possible consequences. Decision trees can be used to identify the most important features or variables that contribute to Parkinson's disease diagnosis.

4. K. Polat[50], Random Forests: Random forests are an ensemble learning method that combines multiple decision trees to improve accuracy and reduce overfitting. Random forests can be used to classify patients with Parkinson's disease based on multiple input variables.

5. H. W. Loh *et al.*,[34]. Deep learning: Deep learning is a type of machine learning that uses artificial neural networks with multiple layers to analyze complex data sets. Deep learning can be used to classify patients with Parkinson's disease based on input data from imaging studies or other sources.

6. V. Despotovic,Gaussian[51], Process Regression (GPR): GPR is a type of machine learning algorithm that can be used for regression analysis. GPR can be used to predict disease progression in Parkinson's disease based on input data, such as motor function assessments or biochemical markers.

These machine learning algorithms have shown promise in improving the accuracy and efficiency of Parkinson's disease diagnosis, as well as predicting disease progression and response to treatment. However, further research is needed to validate the use of these algorithms in clinical practice and to optimize their performance for Parkinson's disease diagnosis and management.

- Voice-based diagnosis using AI

Karaman[16] Voice-based diagnosis using AI is a cutting-edge technology that is transforming the healthcare industry. By analyzing voice patterns, AI algorithms can detect potential medical conditions or diseases, allowing doctors and healthcare professionals to make quicker and more accurate diagnoses. In this article, we will explore the process of voice-based diagnosis using AI and its potential applications.

Data Collection: The first step in the voice-based diagnosis using AI is data collection. Large datasets of voice recordings are needed to train machine learning models to identify patterns

and make accurate diagnoses. These datasets are collected from patients who have been diagnosed with various medical conditions, including respiratory illnesses, neurological disorders, and mental health conditions.

S. M. van Rooden,[35] Data Preprocessing: Once the datasets are collected, they are preprocessed to extract meaningful features. This involves analyzing the voice recordings to identify patterns and extract relevant features that can be used to train the AI models.

A. Zhan *et al.*[36] Machine Learning: The preprocessed data is then used to train machine learning models. There are various machine learning algorithms that can be used for voice-based diagnosis, including neural networks, decision trees, and support vector machines. These algorithms are trained to identify patterns in the voice recordings and make accurate diagnoses.

In conclusion, voice-based diagnosis using AI is a rapidly advancing technology that has the potential to revolutionize the way medical conditions are diagnosed and managed.

OBJECTIVE

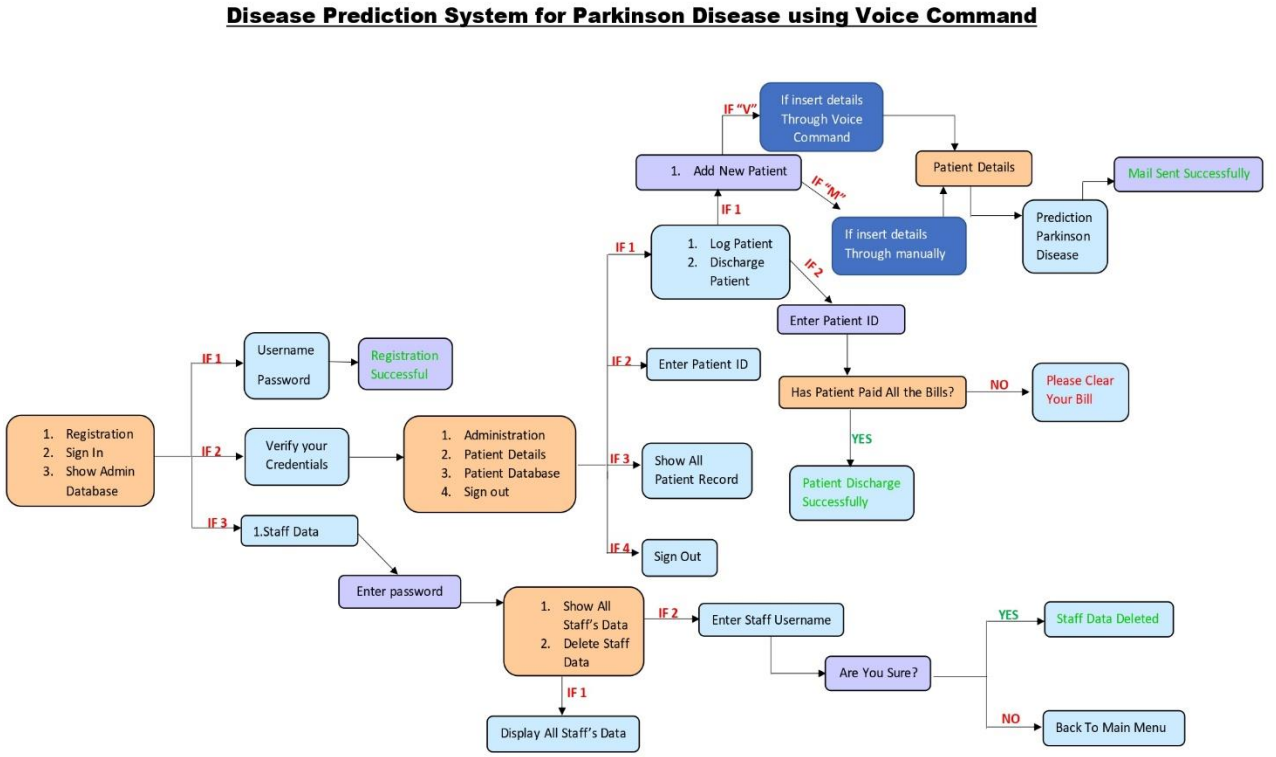
The primary aim of our project this project is to **predict the Parkinson** from the given symptoms by patients and create and monitor a health profile of every individual patient.

Health is the most important factor for everyone. But unfortunately it has been neglected today for many reasons. Absence of doctor due to some reason during emergency may result in loss of life. Not only that sometimes patient often feel hesitant to go to hospital for minor symptoms. These may prompt into major illness. With the proliferation of technology in health care becomes easier to diagnosis any disease – even the deadliest ones. It is applied in healthcare to identify the clusters of patients, diseases, and future predictions using different machine learning tools. So, this project work proposed a Disease Prediction System that will help user to receive immediate guidance regarding their health issues.

To diagnose any disease, doctor initially analyses the symptoms of the patient and after that the result is predicted. Similarly, machine diagnose the diseases based on the symptoms just like the doctor does. The system is fetched with various symptoms and their disease related with it.

This system aims to improve disease treatment and its diagnosis in early stages for a faster and better treatment. Therefore, it is an attempt to make a faster and more accurate disease prediction and help the physicians for making a reliable decision in a short span of time by increasing efficiency and quality in health management system.

FLOW CHART DIAGRAM



The above flow-chart diagram represents the total workflow of the hospital management system including all operations of patients' database as well as staffs' database and predicts if the patient is suffering from Parkinson Disease.

TOOLS AND TECHNOLOGY

Tools and Software

- Visual Studio Code:

  Visual Studio Code is a code editor that is redefined and optimized for building and debugging modern web and cloud applications. Visual it is free and also available on our favourite platforms – Linux, macOS, and Windows.

- MySql :

  MySQL Workbench is a graphical tool that works with MySQL. MySQL Workbench offers an easy-to-use interface to perform multiple tasks involved when working with databases. It integrates SQL development, administration, database design, creation, and maintenance into one visual integrated development environment. MySQL Workbench is similar to SQL Server's SSMS, which is used for administering SQL Server.

Programming Language :

- Python :

Python is a high-level programming language, that precludes the need to compile code before executing a program because Python does the compilation in the background. Because Python is a high-level programming language, it abstracts many sophisticated details from the programming code.

Skill-based Technology :

- Machine Learning :

Machine learning is a subfield of artificial intelligence which includes the development of algorithms and statistical models that enable computers to improve their performance in tasks through experience. These algorithms and models are designed to learn from data and make predictions or decisions without explicit instructions.

APPROACH

- **Database**

➢ Patient_db.py
  - insert_patient()
  - show_patientdb()
  - delete_patientdb()
  - show_all_patientdb()

➢ Staff_db.py
  - insert_staff()
  - show_staff()
  - show_all_staffdb()
  - delete_staff()

- **Predicted Disease**

➢ Parkinson.py
  - Parkinson()

- **Send Mail**

➢ Email.py
  - PDF()
  - Send_mail()

- **Voice Command**

➢ Command.py
  - takeCommand()

➢ Speak.py
  - Speak()

➢ Wish.py
  - wishMe()

- **main.py**

APPENDIX

1. Database

➢ Patient_db.py

- **insert_patient()** : This function is used to insert patient details in a database, especially when a new patient is being registered.

```
def insert_patient(patient_id, name, age, sex, address, contact, mail, disease, f_pred, p_pred):

    mycursor.execute("create database if not exists city_hospitals")
    mycursor.execute("use city_hospitals")

     # creating the tables for storing patient details.
    mycursor.execute("create table if not exists patient_detail(patient_id int(4) primary key, name varchar(30) ,sex
varchar(15),age int(3),address varchar(50),contact varchar(15),mail varchar(40), disease varchar(80), breasr_cancer_prediction
varchar(20), parkinson_disease_prediction varchar(20))")

    # Inserting Patient Details
    mycursor.execute("insert into patient_detail values('" + patient_id + "','" + name + "','" + sex + "','" + age + "','" +
address + "','" + contact + "','" + mail + "','" + disease + "','" + f_pred + "','" + p_pred + "')")
    mysql.commit()
```

- **show_patientdb()** : This function is used to retrieve patients' records. Staff members can get specific patient details by using their patient ID to keep the track of their health issues.

```
def show_patientdb(patient_id):
    mycursor.execute("SELECT * FROM city_hospitals.patient_detail where patient_id ='" + patient_id + "'")
    row = mycursor.fetchall()
    return row
```

- **delete_patientdb()** : This function is used to delete the patient details from the database if it's no longer required.

```
def delete_patientdb(patient_id):
    mycursor.execute("delete from city_hospitals.patient_detail where patient_id='" + patient_id + "'")
    mysql.commit()
```

- **show_all_patientdb()** : This function is used to show all the patients' records present in the database.

```
def show_all_patientdb():
    mycursor.execute("SELECT * FROM city_hospitals.patient_detail")
    row1 = mycursor.fetchall()
    return row1
```

➢ Staff_db.py

- **insert_staff()** : This function is used to insert staff details in the database like their username and password.

```
def insert_staff(u, p):



    mycursor.execute("create database if not exists city_hospitals")

    mycursor.execute("use city_hospitals")

    # creating table for storing the username and password of the user
    mycursor.execute("create table if not exists user_data(username varchar(30) primary key,password varchar(30) default'000')")
    mycursor.execute("insert into user_data values('" + u + "','" + p + "')")
    mysql.commit()
```

- **show_staff()** : This function is used to retrieve all the Staff details from the database.

```
def show_staffdb(un):
    mycursor.execute("select password from city_hospitals.user_data where username='" + un + "'")
    row = mycursor.fetchall()
    return row
```

- **show_all_staffdb()** : This function is used to show all the staff records present in the database.

```
def show_all_staffdb():
    mycursor.execute("select * from city_hospitals.user_data")
    row1 = mycursor.fetchall()
    return row1
```

- **delete_staff()** : This function is used to delete the staff details from the database if it's no longer required.

```
def delete_stafftdb(un):
    mycursor.execute("delete from city_hospitals.user_data where username='" + un + "'")
    mysql.commit()
```

---

2. Predicted Disease

➢ Parkinson.py

- **Parkinson()** : This function is used to predict the Parkinson Disease.

Python-Final-project > disease_pred > parkinson.py > parkinson

You, 1 hour ago | 2 authors (Shahil-1999 and others)

```python
import os
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
import voice.speak as sp
import voice.command as cmd
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler
from sklearn import svm
from sklearn.metrics import accuracy_score
from sklearn.tree import DecisionTreeClassifier
import warnings
warnings.filterwarnings("ignore", category=DeprecationWarning)
warnings.filterwarnings("ignore", category=UserWarning)

def parkinson():



    df = pd.read_csv('C:/Users/kshah/OneDrive/Desktop/test_major_project/Python-Final-project/disease_pred/parkinsons.csv')
    # print(df.info())
    # print(df.describe())
    df.isnull().sum()#checking for missing values


    #dropping column axis = 1; dropping row then axis = 0
    #Data Pre-Processing - Seperating Features and Target variables according to their Correlation

    df.drop(["name",'spread1', 'MDVP:Flo(Hz)','MDVP:Fhi(Hz)','MDVP:Fo(Hz)'], axis=1, inplace=True)
    columns = list(df.columns)
    for column in columns:
        if column == "status":
            continue
```

Python-Final-project > disease_pred > parkinson.py > parkinson

```python
            filtered_columns = [column]
            for col in df.columns:
                if (column == col) | (column == "status"):
                    continue
                cor_val = df[column].corr(df[col])
                if cor_val > 0.75:
                    columns.remove(col)
                    continue
                else:
                    filtered_columns.append(col)
            df = df[filtered_columns]


        df.isnull().sum() #checking null value


        # converting Data in the form of hundred
        # df.iloc[:,:8] = (df.iloc[:, 8:9])









        #Plotting Heatmap
        # plt.figure(figsize=(25, 7))
        # p = sns.heatmap(df.corr(), annot=True)
        # plt.show()


        # plotting bar figure on STATUS column
        # sns.set_style('whitegrid')
        # sns.set_context('paper')
        # sns.set_palette('GnBu_d')
        # a = sns.catplot(x='status', data=df, kind='count')
        # plt.title('Number of Samples in Each Class')
        # a.set(ylabel='Number of Samples', xlabel='Have Parkinson')
```

Python-Final-project > disease_pred > parkinson.py > parkinson

```
75  # plt.show()
76
77  #histogram
78  # df.hist(figsize=(25,7))
79  # plt.show()
80  #We can see some of the data is normally distributed and most of the attributes are right skewed
81
82
83
84  # Splitting the data into testing and training set
85  x_train, x_test, y_train, y_test = train_test_split(df.drop(columns=['status']), df['status'], test_size=0.2, random_state = 42)
86
87  # Model Training (DecisionTreeClassifier)
88  clf = DecisionTreeClassifier()
89  clf.fit(x_train, y_train)
90
91  # # Model Evaluation
92  # # Accuracy Score
93
94  # # Accuracy Score on training data
95  # x_train_pred = clf.predict(x_train)
96  # training_data_accuracy = accuracy_score(y_train, x_train_pred)
97
98  # print('Accuracy (Training Data) :', training_data_accuracy*100,'%')
99
100
101 # # Accuracy Score on test data
102 # x_test_pred = clf.predict(x_test)
103 # testing_data_accuracy = accuracy_score(y_test, x_test_pred)
104
105
106 # print('Accuracy (Testing Data) :', testing_data_accuracy*100,'%')
107
108
109
110 print("Enter your First nonlinear dynamical complexity measures (1.42 - 3.67)")
```

Python-Final-project > disease_pred > parkinson.py > parkinson

```
111     sp.speak("Enter your First nonlinear dynamical complexity measures (1.42 - 3.67)")
112     D2 = cmd.takeCommand().lower()
113     print(D2)
114     sp.speak(D2)        Shahil-1999, 2 months ago • Disease Prediction System
115
116     print("if you satisfy with this value then press Y, else press N")
117     sp.speak("if you satisfy with this value then press Y, else press N")
118
119     D2_satisfaction = input("Enter your value : ")
120     if D2_satisfaction.lower() == "n":
121         sp.speak("Enter New Value: ")
122         new_input_D2 = input("Enter New Value : ")
123         D2 = (new_input_D2)
124         print(D2)
125         sp.speak(D2)
126     # else:
127     #     os.system("pause")
128
129
130
131
132
133
134     print("Enter your second nonlinear dynamical complexity measures (0.25 - 0.68)")
135     sp.speak("Enter your second nonlinear dynamical complexity measures (0.25 - 0.68)")
136     RPDE = cmd.takeCommand().lower()
137     print(RPDE)
138     sp.speak(RPDE)
139
140     sp.speak("if you satisfy with this value then press Y, else press N")
141     RPDE_satisfaction = input("Enter your value : ")
142     if RPDE_satisfaction.lower() == "n":
143         sp.speak("Enter New Value: ")
144         new_input_RPDE = input("Enter New Value : ")
145         RPDE = (new_input_RPDE)
146         print(RPDE)
147         sp.speak(RPDE)
```

Python-Final-project > disease_pred > parkinson.py > parkinson

```
    # else:
    #     os.system("pause")


    print('Enter your third nonlinear measures of fundamental frequency variation (0.04 - 0.52)')
    sp.speak('Enter your third nonlinear measures of fundamental frequency variation (0.04 - 0.52)')
    PPE = cmd.takeCommand().lower()
    print(PPE)
    sp.speak(PPE)

    sp.speak("if you satisfy with this value then press Y, else press N")
    PPE_satisfaction = input("Enter your value : ")
    if PPE_satisfaction.lower() == "n":
        sp.speak("Enter New Value: ")
        new_input_PPE = input("Enter New Value : ")
        PPE = (new_input_PPE)
        print(PPE)
        sp.speak(PPE)
    # else:
    #     os.system("pause")

    print("Enter your nonlinear fundamental frequency variation (0.00 - 0.45)")
    sp.speak("Enter your nonlinear fundamental frequency variation (0.00 - 0.45)")
    spread2 = cmd.takeCommand().lower()
    print(spread2)
    sp.speak(spread2)

    sp.speak("if you satisfy with this value then press Y, else press N")
    spread2_satisfaction = input("Enter your value : ")
    if spread2_satisfaction.lower() == "n":
        sp.speak("Enter New Value: ")
        new_input_spread2 = input("Enter New Value : ")
        spread2 = (new_input_spread2)
        print(spread2)
        sp.speak(spread2)
    # else:
    #     os.system("pause")
```

Python-Final-project > disease_pred > parkinson.py > parkinson

```
      print("Enter your Signal fractal scaling exponent (0.57 - 0.82)")
      sp.speak("Enter your Signal fractal scaling exponent (0.57 - 0.82)")
      DFA = cmd.takeCommand().lower()
      print(DFA)
      sp.speak(DFA)


      sp.speak("if you satisfy with this value then press Y, else press N")
      DFA_satisfaction = input("Enter your value : ")
      if DFA_satisfaction.lower() == "n":
          sp.speak("Enter New Value: ")
          new_input_DFA = input("Enter New Value : ")
          DFA = (new_input_DFA)
          print(DFA)
          sp.speak(DFA)
      # else:
      #     os.system("pause")



      print("Enter your ratio of noise to tonal components in the voice (8.44 - 33.04)")
      sp.speak("Enter your ratio of noise to tonal components in the voice (8.44 - 33.04)")
      HNR = cmd.takeCommand().lower()
      print(HNR)
      sp.speak(HNR)


      sp.speak("if you satisfy with this value then press Y, else press N")
      HNR_satisfaction = input("Enter your value : ")
      if HNR_satisfaction.lower() == "n":
          sp.speak("Enter New Value: ")
          new_input_HNR = input("Enter New Value : ")
          HNR = (new_input_HNR)
          print(HNR)
          sp.speak(HNR)
```

Python-Final-project > disease_pred > parkinson.py > parkinson

```
221         # else:
222         #     os.system("pause")
223 
224 
225 
226 
227         print("Enter your Several measures of variation in amplitude(0.00 - 0.05)")
228         sp.speak("Enter your Several measures of variation in amplitude(0.00 - 0.05)")
229         Shimar = cmd.takeCommand().lower()
230         print(Shimar)
231         sp.speak(Shimar)
232 
233         sp.speak("if you satisfy with this value then press Y, else press N")
234         Shimar_satisfaction = input("Enter your value : ")
235         if Shimar_satisfaction.lower() == "n":
236             sp.speak("Enter New Value: ")
237             new_input_Shimar = input("Enter New Value : ")
238             Shimar = (new_input_Shimar)
239             print(Shimar)
240             sp.speak(Shimar)
241         # else:
242         #     os.system("pause")
243 
244 
245 
246         print("Enter your Several measures of variation in fundamental frequency (0.00 - 0.03)")
247         sp.speak("Enter your Several measures of variation in fundamental frequency (0.00 - 0.03)")
248         Jitter = cmd.takeCommand().lower()
249         print(Jitter)
250         sp.speak(Jitter)
251 
252         sp.speak("if you satisfy with this value then press Y, else press N")
253         Jitter_satisfaction = input("Enter your value : ")
254         if Jitter_satisfaction.lower() == "n":
255             sp.speak("Enter New Value: ")
256             new_input_Jitter = input("Enter New Value : ")
257             Jitter = (new input Jitter)
```

Python-Final-project > disease_pred > parkinson.py > parkinson

```
258             print(Jitter)
259             sp.speak(Jitter)
260         # else:
261         #     os.system("pause")
262 
263 
264 
265 
266 
267         p_pred = clf.predict([[D2, RPDE, PPE, spread2, DFA, HNR,Shimar, Jitter]])
268 
269 
270         predicted = ""
271 
272         if p_pred == 0:
273             predicted = 'Not Affected'
274 
275 
276         else:
277             p_pred == 1
278             predicted = 'Affected'
279 
280 
281             return predicted
282 
```

3. Send Mail

➢ Email.py

- **PDF()** : This function is used to create patient reports in pdf format.

```
class PDF(FPDF):
    def header(self):
        #logo
        self.image('C:/Users/kshah/OneDrive/Desktop/test_major_project/mail/code.jpg', 8, 8, 15)
        self.image('C:/Users/kshah/OneDrive/Desktop/test_major_project/mail/R.jpg', 187, 7, 15)
        #font
        self.add_font('Lucida Bright','',r'C:/Windows/Fonts/LCALLIG.TTF', uni=True) # uni = True does true type font subset embedding
        self.set_font('Lucida Bright','U',20)
        self.set_text_color(134,108,15)
        # Title
        self.cell(0,10,'Disease prediction System for',ln=True, align='C',)
        self.cell(0,10,'Breast Cancer and Parkinson ',ln=True, align='C',)
        self.cell(0,10,'Disease Using Voice Command',ln=True, align='C',)
        #line break
        self.ln(20)
```

```
# Create Object
pdf = PDF('P','mm')
# Add a page
pdf.add_page()

#Specify Fonts ('times','courier' etc)
# 'B'(bold), 'U' (underline), ' I'(Italics), ''(regular),'BU' (combination)
# Font size
pdf.set_font('courier', 'U', 16)

pdf.add_font('Goudy Old Style','',r'C:/Windows/Fonts/GOUDOSB.TTF', uni=True) # uni = True does true type font subset embedding
pdf.set_font('Goudy Old Style', '', 16)
# pdf.set_text_color(0,0,0)
# Add Text
# w = 'width'
# h = 'height'
pdf.cell(0,10, f'Patient ID: {patient_id}', ln=1)
pdf.cell(0,10, f'Name: {name}', ln=1)
pdf.cell(0,10, f'Age: {age}', ln=1)
```

mail > email.py > send_mail

```
60      pdf.cell(0,10, f'Gender: {sex}', ln=1)
61      pdf.cell(0,10, f'Address: {address}', ln=1)
62      pdf.cell(0,10, f'Contact: +91{contact}', ln=1)
63      pdf.cell(0,10, f'Mail: {mail}', ln=1)
64      pdf.cell(0,10, f'Disease: {desease}', ln=1)
65      pdf.cell(0,10, f'Breast Cancer Prediction: {f_pred}', ln=1)
66      pdf.cell(0,10, f'Parkinson Disease Prediction: {p_pred}', ln=1)
67      y_axis_initial = 255
68      pdf.set_y(y_axis_initial)
69      pdf.set_font('times', 'U', 16)
70      pdf.cell(0,10,"Doctor's Sign with Date", ln = 1, align='R')
71      a = "C:/Users/kshah/OneDrive/Desktop/test_major_project/Paitent_Details_PDF/"
72      pdf.output(f"{a}{name}.pdf")
```

- **Send_mail()** : This function is used to automatically send the PDF report to the patient via email.

```
email_sender = "shahil.official.college@gmail.com"
pwd_sender = os.environ.get("Email_Password")
receiver = mail
# msg = MIMEMultipart()

subject = f'{name} Health Report'

body = ''
em = EmailMessage()
em['From'] = email_sender

em['To'] = receiver
em['Subject'] = subject

em.set_content(body)
context = ssl.create_default_context()


files = f"C:/Users/kshah/OneDrive/Desktop/test_major_project/Paitent_Details_PDF/{name}.pdf"
with open (files, 'rb') as m:
    file_data = m.read()
    file_name = name + ".pdf"
em.add_attachment(file_data, maintype='pdf', subtype = 'octet-stream', filename = file_name)
with smtplib.SMTP_SSL('smtp.gmail.com', 465) as smtp:
    smtp.login(email_sender, pwd_sender)
    smtp.sendmail(
        email_sender, receiver, em.as_string())
```

---

4. Voice Command

➢ Command.py

- **takeCommand()** : This function is used to take Voice Commands from the user.

```
import speech_recognition as sr  # pip install speechRecognition


def takeCommand():
# It takes microphone input from the user and returns string output

    r = sr.Recognizer()
    with sr.Microphone() as source:
        print("Listening...",source)
        r.pause_threshold = 1
        audio = r.listen(source)

    try:
        print("Recognizing...")
        query = r.recognize_google(audio, language='en - in')


    except Exception as e:
        # print(e)
        print("Say that again please...")
        return "None"

    return query
```

➢ Speak.py

- **Speak()** :

```
import pyttsx3  # pip install pyttsx3


engine = pyttsx3.init('sapi5')
voices = engine.getProperty('voices')
# print(voices[1].id)
engine.setProperty('voice', voices[1].id)

def speak(audio):
    engine.say(audio)
    engine.runAndWait()
```

- **Wish.py**
  - **wishMe()** : This function is used to wish or greet.

```
import datetime
import voice.speak as sp

def wishMe():
    hour = int(datetime.datetime.now().hour)
    if hour >= 0 and hour < 12:
        sp.speak("Good Morning!")

    elif hour >= 12 and hour < 18:
        sp.speak("Good Afternoon")

    else:
        sp.speak("Good Evening")
```

---

5. main.py

Python-Final-project > main.py > ...

You, 3 hours ago | 2 authors (Shahil-1999 and others)

```python
import os
import voice.speak as sp
import voice.command as cmd
import voice.wish as ws
import mail.email as email
import disease_pred.parkinson as park
import database.patient_db as p_details
import database.staff_db as sd


while (True):


    Admin_passwd = os.environ.get("Admin_Password")


    while (True):
        ws.wishMe()
        print("Disease Prediction System for Parkinson Disease using Voice Command")
        sp.speak("Disease Prediction System for Parkinson Disease using Voice Command")

        print("""
                        1. Registration
                        2. Sign In
                        3. Show Admin database
                        """)
        sp.speak("""press 1 for Registration
                    press 2 for Sign In
                    press 3 for Show Admin database""")


        r = int(input("enter your choice: "))
        if r == 1:
            print("""
                !!!!!!!!!!Register Yourself!!!!!!!!""")
```

Shahil-1999, 2 months ago • Disease Prediction System

Python-Final-project > main.py > ...

```python
            sp.speak(("Please Enter username..."))
            u = input("Enter username!!:")


            sp.speak("Please Enter password(Password Must Be Strong)...")
            p = input("Enter Your password (Password must be strong!!!): ")

            show_userdb = sd.show_staffdb(u)

            if len(show_userdb):
                print("Username Already Registered choose different username")
                sp.speak("Username Already Registered choose different username")
            else:
                sd.insert_staff(u, p)

                print("""
                ============================================
                !!Well Done!!Registration Done Successfully!!
                ============================================
                                                """)
                sp.speak("Registration Done Sucessfully...")

            sp.speak("Press any key to continue")
            os.system("pause")


        # IF USER WANTS TO LOGIN
        elif r == 2:
            print("""
                    ==================================
                    !!!!!!!!  {{Sign In}}  !!!!!!!!!!
                    ==================================
                                                """)

            print("Please Verify Your Credentials")
            sp.speak("Please Verify Your Credentials")
```

Python-Final-project > main.py > ...

```
75                  print("Please Enter Your Username")
76                  sp.speak("Please Enter Your Username")
77                  un = input("Username!!: ")
78
79
80                  print("Please Enter Your Password")
81                  sp.speak("Please Enter Your Password")
82                  ps = input("Password!!: ")
83
84
85
86                  row = sd.show_staffdb(un)
87                  for i in row:
88                      a = list(i)
89                      if a[0] == str(ps):
90                          while(True):
91                              print("""
92                                          1. Administration
93                                          2. Patient(Details)
94                                          3. Show patient database
95                                          4. Sign Out
96                                                      """)
97                              sp.speak("""
98                                          press 1 for Administration
99                                          press 2 for Patient(Details)
100                                         press 3 for Show patient database
101                                         press 4 for Sign Out
102                                                     """)
103
104                             a = int(input("ENTER YOUR CHOICE: "))
105                             if a == 1:
106                                 print("""
107                                         1. Log patient Record
108                                         2. Discharge Summary
109                                         """)
110
111                                 sp.speak("""
```

Python-Final-project > main.py > ...

```
                    press 1 for Log patient Record
                    press 2 for Discharge Summary
                    """)



                x = int(input("ENTER YOUR CHOICE: "))
                if x == 1:

                    print("""
                            1. Add New Patient

                            """)
                    sp.speak("press 1 Add New Patient")
                    b = int(input("Enter Your Choice: "))

                    # adding new patient
                    if b == 1:
                        print("If You Want to insert patient details manually then press M, or If you want to add patient details through voice then press V")
                        sp.speak("If You Want to insert patient details manually then press M, or If you want to add patient details through voice then press V")
                        p_d = input("Enter Your response: ")


                        if p_d.lower() == 'm':
                            print("Please Enter Your patient ID")
                            sp.speak("Please Enter Your patient ID")
                            patient_id = input("Patient ID: ")

                            print("Please Enter Your Name")
                            sp.speak("Please Enter Your Name")
                            name = input("Name: ")

                            print("Please Enter Your Gender")
                            sp.speak("Please Enter Your Gender")
                            sex = input("Gender: ")
```

Python-Final-project > main.py > ...

```
                            print("Please Enter your Age")
                            sp.speak("Please Enter your Age")
                            age = input("Age: ")

                            print("Please Enter Your Address")
                            sp.speak("Please Enter Your Address")
                            address = input("Address: ")

                            print("Please Enter Your Contact Number")
                            sp.speak("Please Enter Your Contact Number")
                            contact = input("Contact Details: ")

                            print("Please Enter Your Email")
                            sp.speak("Please Enter Your Email")
                            mail = input("Mail Id: ")

                            print("Please Enter Your Disease")
                            sp.speak("Please Enter Your Disease")
                            disease = input("Disease: ")

                        elif p_d.lower() == "v":




                            print("Please Enter Your patient ID")
                            sp.speak("Please Enter Your patient ID")
                            patient_id = cmd.takeCommand().lower()
                            print(patient_id)
                            sp.speak(patient_id)

                            print("if you satisfy with this value then press Y, else press N")
                            sp.speak("if you satisfy with this value then press Y, else press N")
                            patient_id_satisfaction = input("Enter your value : ")
                            if patient_id_satisfaction.lower() == "n":
```

patient_db.py | main.py M | parkinson.py

Python-Final-project > main.py > ...

```
184                     sp.speak("Enter New Value: ")
185                     new_input_patient_id = input("Enter New Value : ")
186                     patient_id = (new_input_patient_id)
187                     print(patient_id)
188                     sp.speak(patient_id)
189
190
191
192
193                 print("Please Enter Your Name")
194                 sp.speak("Please Enter Your Name")
195                 name = cmd.takeCommand().lower()
196                 print(name)
197                 sp.speak(name)
198
199                 print("if you satisfy with this value then press Y, else press N")
200                 sp.speak("if you satisfy with this value then press Y, else press N")
201                 name_satisfaction = input("Enter your value : ")
202                 if name_satisfaction.lower() == "n":
203                     sp.speak("Enter New Value: ")
204                     new_input_name = input("Enter New Value : ")
205                     name = (new_input_name)
206                     print(name)
207                     sp.speak(name)
208
209
210                 print("Please Enter Your Gender")
211                 sp.speak("Please Enter Your Gender")
212                 sex = cmd.takeCommand().lower()
213                 print(sex)
214                 sp.speak(sex)
215
216                 print("if you satisfy with this value then press Y, else press N")
217                 sp.speak("if you satisfy with this value then press Y, else press N")
218                 gender_satisfaction = input("Enter your value : ")
219                 if gender_satisfaction.lower() == "n":
220                     sp.speak("Enter New Value: ")
```

patient_db.py | main.py M | parkinson.py

Python-Final-project > main.py > ...

```
221                     new_input_gender = input("Enter New Value : ")
222                     sex = (new_input_gender)
223                     print(sex)
224                     sp.speak(sex)
225
226
227
228                 print("Please Enter Your Age")
229                 sp.speak("Please Enter Your Age")
230                 age = cmd.takeCommand().lower()
231                 print(age)
232                 sp.speak(age)
233
234                 print("if you satisfy with this value then press Y, else press N")
235                 sp.speak("if you satisfy with this value then press Y, else press N")
236                 age_satisfaction = input("Enter your value : ")
237                 if age_satisfaction.lower() == "n":
238                     sp.speak("Enter New Value: ")
239                     new_input_age = input("Enter New Value : ")
240                     age= (new_input_age)
241                     print(age)
242                     sp.speak(age)
243
244                 print("Please Enter Your Address")
245                 sp.speak("Please Enter Your Address")
246                 address = cmd.takeCommand().lower()
247                 print(address)
248                 sp.speak(address)
249
250                 print("if you satisfy with this value then press Y, else press N")
251                 sp.speak("if you satisfy with this value then press Y, else press N")
252                 address_satisfaction = input("Enter your value : ")
253                 if address_satisfaction.lower() == "n":
254                     sp.speak("Enter New Value: ")
255                     new_input_address = input("Enter New Value : ")
256                     address = (new_input_address)
257                     print(address)
```

Python-Final-project > main.py > ...

```
258                     sp.speak(address)
259
260                 print("Please Enter Your Contact number")
261                 sp.speak("Please Enter Your contact Number")
262                 contact = cmd.takeCommand().lower()
263                 print(contact)
264                 sp.speak(contact)
265
266                 print("if you satisfy with this value then press Y, else press N")
267                 sp.speak("if you satisfy with this value then press Y, else press N")
268                 contact_satisfaction = input("Enter your value : ")
269                 if contact_satisfaction.lower() == "n":
270                     sp.speak("Enter New Value: ")
271                     new_input_contact = input("Enter New Value : ")
272                     contact = (new_input_contact)
273                     print(contact)
274                     sp.speak(contact)
275
276                 print("Please Enter Your Email")
277                 sp.speak("Please Enter Your Email")
278                 mail = cmd.takeCommand().lower()
279                 print(mail)
280                 sp.speak(mail)
281
282                 print("if you satisfy with this value then press Y, else press N")
283                 sp.speak("if you satisfy with this value then press Y, else press N")
284                 mail_satisfaction = input("Enter your value : ")
285                 if mail_satisfaction.lower() == "n":
286                     sp.speak("Enter New Value: ")
287                     new_input_mail = input("Enter New Value : ")
288                     mail = (new_input_mail)
289                     print(mail)
290                     sp.speak(mail)
291
292
293                 print("Please Enter Your Disease")
294                 sp.speak("Please Enter Your disease")
```

patient_db.py | main.py M | parkinson.py

Python-Final-project > main.py > ...

```
295                     disease = cmd.takeCommand().lower()
296                     print(disease)
297                     sp.speak(disease)
298
299                     print("if you satisfy with this value then press Y, else press N")
300                     sp.speak("if you satisfy with this value then press Y, else press N")
301                     disease_satisfaction = input("Enter your value : ")
302                     if disease_satisfaction.lower() == "n":
303                         sp.speak("Enter New Value: ")
304                         new_input_disease = input("Enter New Value : ")
305                         patient_id = (new_input_disease)
306                         print(disease)
307                         sp.speak(disease)
308
309             else:
310                 print("Choose Valid Option")
311                 sp.speak("Choose Valid Option")
312
313
314             # print("Prediction Breast Cancer?")
315             # sp.speak("Prediction Breast Cancer?")
316             # br_cancer = input("prediction breast cancer? (y/n): ")
317             # f_pred = "NA"
318
319             # if br_cancer == "y":
320             #     f_pred = b_cancer.breast_cancer()
321             #     print(f_pred)
322
323             print("Prediction parkinson Disease")
324             sp.speak('prediction parkinson Disease')
325             p_cancer = input("prediction parkinson disease? (y/n): ")
326             p_pred = "NA"
327
328
329             if p_cancer == "y":
330                 # print("parkinson abcd")
331                 p_pred = park.parkinson()
```

Python-Final-project > main.py > ...

```
                print(p_pred)



            p_details.insert_patient(patient_id, name, sex ,age , address, contact, mail, disease, p_pred)


            print("""
                  ==================================
                  !!!!!!!Registered Successfully!!!!!!
                  ==================================
                        """)
            sp.speak("Patient Registered Sucessfully")

            sp.speak("press Any Key To Continue")
            os.system("pause")



            print("p_pred", p_pred)
            email.send_mail(patient_id, name, age, sex, address, contact, mail, disease, p_pred)
            sp.speak("Mail Sent Sucessfully")

        else:
            print("please Choose Valid Option")
            sp.speak("Please Choose Valid Option")

    # dischare process
    elif x == 2:

        print("Please Enter The Patient Name")
        sp.speak("Please Enter The Patient ID")
        patient_id = input("Enter The Patient ID: ")
```

Python-Final-project > main.py > ...

```
            row = p_details.show_patientdb(patient_id)
            if len(row):
                for i in row:
                    b1 = 0
                    v1 = list(i)
                    k1 = ["PATIENT ID","NAME", "SEX", "AGE", "ADDRESS", "CONTACT",
                          "MAIL", "DISEASE", "PARKINSON DISEASE PRIDICTION"]
                    d1 = dict(zip(k1, v1))
                    print(d1)
            else:
                print("patient Dose not Exist")
                sp.speak("patient Dose not Exist")

                sp.speak("press any key to continue")
                os.system("pause")
                break


            print("Has Patient Paid all the bills")
            sp.speak("Has Patient Paid all the Bills?")
            bill = input("Has he paid all the bills? (y/n):")

            if bill == "y":
                p_details.delete_patientdb(patient_id)
                print("Patient Discharged Sucessfully")
                sp.speak("Patient Discharged Sucessfully")
            else:

                print("please clear your bill")
                sp.speak("Please Clear Your Bill")

        else:

            print("Please Choose Valid Option")
            sp.speak("Please Choose Valid Option")

            sp.speak("Press Any Key To Continue")
```

Python-Final-project > main.py > ...

```
406                         os.system("pause")
407
408
409                     # if user wants to see the details of PATIENT
410                     elif a == 2:
411
412
413                         sp.speak("please Enter patient ID")
414                         patient_id = input("Enter The Patient ID: ")
415
416
417                         row = p_details.show_patientdb(patient_id)
418                         if len(row):
419                             for i in row:
420                                 b = 0
421                                 v = list(i)
422                                 k = ["PATIENT ID", "NAME", "SEX", "AGE", "ADDRESS", "CONTACT",
423                                     "MAIL", "DISEASE", "PARKINSON DISEASE PRIDICTION"]
424                                 d = dict(zip(k, v))
425                                 print(d)
426                         else:
427                             print("patient Dose not Exist in our database")
428                             sp.speak("patient Dose not Exist in our database")
429
430                             os.system("pause")
431                             sp.speak("press any key to continue")
432
433
434                     # if user wants to show all patient records
435                     elif a == 3:
436
437                         row1 = p_details.show_all_patientdb()
438                         for i in row1:
439                             b = 0
440                             v = list(i)
441                             k = ["PATIENT ID","NAME", "SEX", "AGE", "ADDRESS", "CONTACT",
442                                 "MAIL", "DISEASE", "PARKINSON DISEASE PRIDICTION"]
```

Python-Final-project > main.py > ...

```
443                                 d = dict(zip(k, v))
444                                 print(d)
445
446                             # SIGN OUT
447                             elif a == 4:
448                                 break
449
450                             else:
451                                 print("Choose Valid Option")
452                                 sp.speak("Choose Valid Option")
453
454                     # IF THE USERNAME AND PASSWORD IS NOT IN THE DATABASE
455                     else:
456                         print("Staff Dosen't Exist in our database")
457                         sp.speak("Staff Dosen't Exist in our database")
458                         break
459
460             elif r == 3:
461
462                 print("""
463                       1. Staff's Data
464                       """)
465                 sp.speak("press 1 for Staff's Data")
466
467                 i = int(input("Enter Your Choice: "))
468                 if i == 1:
469
470
471                     sp.speak("Please Enter Your Password")
472                     pwd = str(input("Enter Your Password: "))
473                     if pwd == Admin_passwd:
474                         while(True):
475                             print("""
476                                                 1. Show Staff's Data
477                                                 2. Delete staff's data
478
479                                                     """)
```

Python-Final-project > main.py > ...

```
481                         sp.speak("""
482                                     press 1 Show Staff's Data
483                                     press 2 Delete staff's data
484 
485                                                 """)
486 
487 
488                         x = int(input("Enter your Choice: "))
489                         if x == 1:
490 
491                             row1 = sd.show_all_staffdb()
492                             for i in row1:
493                                 b = 0
494                                 v = list(i)
495                                 k = ["USERNAME", "PASSWORD"]
496                                 d = dict(zip(k, v))
497                                 print(d)
498                             sp.speak("press any key to continue")
499                             os.system("pause")
500 
501 
502 
503                             break
504 
505                         elif x == 2:
506 
507 
508                             sp.speak("Please Enter The Staff's Username")
509                             un = input("Enter the Staff Username: ")
510 
511 
512                             sp.speak("Are you Sure")
513                             sure = input("Are You Sure? (y/n):")
514 
515                             if sure == "y":
```

Python-Final-project > main.py > ...

```
517                                     sd.delete_stafftdb(un)
518                                     print("Successfully Staff deleted")
519                                     sp.speak("Staff Sucessfully Deleted")
520                                 else:
521                                     print("Staff Not Found")
522                                     sp.speak("Staff Not Found")
523 
524                                 sp.speak("Press Any Key To continue")
525                                 os.system("pause")
526                                 break
527 
528                             else:
529                                 print("Please Choose Valid Option")
530                                 sp.speak("Please Choose valid Option")
531                     else:
532                         print("Invalid Password")
533                         sp.speak("Invalid Password")
534                         break
535                 else:
536                     print("Please Choose Valid Option")
537                     sp.speak("Please Choose Valid Option")
538             else:
539                 print("Please Choose Valid Option")
540                 sp.speak("Please Choose Valid Option")
```

OUTPUTS

```
PS C:\Users\kshah\OneDrive\Desktop\test_major_project> & C:/Users/kshah/AppData/Local/
y
Disease Prediction System for Parkinson Disease using Voice Command

                            1. Registration
                            2. Sign In
                            3. Show Admin database

enter your choice: 1

                    !!!!!!!!!!Register Yourself!!!!!!!!
Enter username!!:Skc
Enter Your password (Password must be strong!!!): Skc@123

                    ============================================
                    !!Well Done!!Registration Done Successfully!!
                    ============================================

Press any key to continue . . .
Disease Prediction System for Parkinson Disease using Voice Command

                            1. Registration
                            2. Sign In
                            3. Show Admin database

enter your choice: 2

                        ====================================
                        !!!!!!!!  {{Sign In}}  !!!!!!!!!!
                        ====================================

Please Verify Your Credentials
Please Enter Your Username
Username!!: Skc
Please Enter Your Password
Password!!: Skc@123

                                        1. Administration
                                        2. Patient(Details)
                                        3. Show patient database
                                        4. Sign Out

ENTER YOUR CHOICE: 1

                                _______ 1. Log patient Record

Enter Your Choice: 1
If You Want to insert patient details manually then press M, or If you want to add patient details through voice then press V
Enter Your response: m
Please Enter Your patient ID
Patient ID: 087
Please Enter Your Name
Name: Rahul
Please Enter Your Gender
Gender: male
Please Enter your Age
Age: 24
Please Enter Your Address
Address: Kolkata
Please Enter Your Contact Number
Contact Details: 123654789
Please Enter Your Email
Mail Id: kshahil1999@gmail.com
Please Enter Your Disease
Disease: Fever, Parkinson
Prediction parkinson Disease
prediction parkinson disease? (y/n): y
Enter your First nonlinear dynamical complexity measures (1.42 - 3.67)
Listening...
Recognizing...
result2:
{   'alternative': [{'confidence': 0.56002802, 'transcript': '2.21'}],
    'final': True}
2.21
if you satisfy with this value then press Y, else press N
Enter your value : y
Enter your second nonlinear dynamical complexity measures (0.25 - 0.68)
Listening...
Recognizing...
result2:
{   'alternative': [{'confidence': 0.97374111, 'transcript': '0.42'}],
    'final': True}
0.42
if you satisfy with this value then press Y, else press N
Enter your value : y
Enter your third nonlinear measures of fundamental frequency variation (0.04 - 0.52)
Listening...
Recognizing...
result2:
{   'alternative': [{'confidence': 0.48600605, 'transcript': '0.21'}],
```

```
0.21
if you satisfy with this value then press Y, else press N
Enter your value : y
Enter your nonlinear fundamental frequency variation (0.00 - 0.45)
Listening...
Recognizing...
result2:
{   'alternative': [{'confidence': 0.98762906, 'transcript': '0.34'}],
    'final': True}
0.34
if you satisfy with this value then press Y, else press N
Enter your value : y
Enter your Signal fractal scaling exponent (0.57 - 0.82)
Listening...
Recognizing...
result2:
{   'alternative': [{'confidence': 0.57697445, 'transcript': '0.69'}],
    'final': True}
0.69
if you satisfy with this value then press Y, else press N
Enter your value : y
Enter your ratio of noise to tonal components in the voice (8.44 - 33.04)
Listening...
Recognizing...
result2:
{   'alternative': [{'confidence': 0.57270861, 'transcript': '11.21'}],
    'final': True}
11.21
if you satisfy with this value then press Y, else press N
Enter your value : y
Enter your Several measures of variation in amplitude(0.00 - 0.05)
Listening...
Recognizing...
result2:
{   'alternative': [{'confidence': 0.85601306, 'transcript': '0.3'}],
    'final': True}
0.3
if you satisfy with this value then press Y, else press N
Enter your value : y
Enter your Several measures of variation in fundamental frequency (0.00 - 0.03)
Listening...
Recognizing...
result2:
{   'alternative': [   {'confidence': 0.85065097, 'transcript': '0.01'},
                       {'transcript': '0 .01'}],
```

```
Enter your Several measures of variation in fundamental frequency (0.00 - 0.03)
Listening...
Recognizing...
result2:
{   'alternative': [   {'confidence': 0.85065097, 'transcript': '0.01'},
                       {'transcript': '0 .01'}],
    'final': True}
0.01
if you satisfy with this value then press Y, else press N
Enter your value : y
pre: Affected
Affected

                                    ==================================
                                    !!!!!!!Registered Successfully!!!!!!
                                    ==================================

Press any key to continue . . .

                                1. Administration
                                2. Patient(Details)
                                3. Show patient database
                                4. Sign Out

ENTER YOUR CHOICE: 2
Enter The Patient ID: 087
{'PATIENT ID': '087', 'NAME': 'Rahul', 'SEX': 'male', 'AGE': '24', 'ADDRESS': 'Kolkata', 'CONTACT': '123654789', 'MAIL': 'kshahil1999@gmail.com', 'DISEASE': 'Fever, Parkinson', 'PARKINSON DISEASE PRIDI
CTION': 'Affected'}

                                1. Administration
                                2. Patient(Details)
                                3. Show patient database
                                4. Sign Out

ENTER YOUR CHOICE: 3
{'PATIENT ID': '012', 'NAME': 'Shahil', 'SEX': 'male', 'AGE': '24', 'ADDRESS': 'bandel', 'CONTACT': '456', 'MAIL': 'kshahil1999@gmail.com', 'DISEASE': 'fever', 'PARKINSON DISEASE PRIDICTION': 'NA'}
{'PATIENT ID': '021', 'NAME': 'asd', 'SEX': 'm', 'AGE': '24', 'ADDRESS': 'a', 'CONTACT': '4', 'MAIL': 'a@gmail.com', 'DISEASE': 'q', 'PARKINSON DISEASE PRIDICTION': 'NA'}
{'PATIENT ID': '023', 'NAME': 'Shahil', 'SEX': 'Male', 'AGE': '24', 'ADDRESS': 'kolkata', 'CONTACT': '8420179105', 'MAIL': 'kumaryadavm93@gmail.com', 'DISEASE': 'Fever, Parkinson', 'PARKINSON DISEASE P
RIDICTION': 'Affected'}
{'PATIENT ID': '028', 'NAME': 'a', 'SEX': 'a', 'AGE': '24', 'ADDRESS': 'a', 'CONTACT': '24', 'MAIL': 'a@gmail.com', 'DISEASE': 'a', 'PARKINSON DISEASE PRIDICTION': 'NA'}
{'PATIENT ID': '041', 'NAME': 'apple', 'SEX': 'male', 'AGE': '24', 'ADDRESS': 'kolkata kolkata', 'CONTACT': '123', 'MAIL': 'a@gmail.com', 'DISEASE': 'beaver', 'PARKINSON DISEASE PRIDICTION': 'Affected'
}
{'PATIENT ID': '087', 'NAME': 'Rahul', 'SEX': 'male', 'AGE': '24', 'ADDRESS': 'Kolkata', 'CONTACT': '123654789', 'MAIL': 'kshahil1999@gmail.com', 'DISEASE': 'Fever, Parkinson', 'PARKINSON DISEASE PRIDI
CTION': 'Affected'}
```

```
                                    1. Administration
                                    2. Patient(Details)
                                    3. Show patient database
                                    4. Sign Out

ENTER YOUR CHOICE: 4
Disease Prediction System for Parkinson Disease using Voice Command

                        1. Registration
                        2. Sign In
                        3. Show Admin database

enter your choice: 3

                    1. Staff's Data

Enter Your Choice: 1
Enter Your Password: Shahil
Invalid Password
Disease Prediction System for Parkinson Disease using Voice Command

                        1. Registration
                        2. Sign In
                        3. Show Admin database

enter your choice: 3

                    1. Staff's Data

Enter Your Choice: 1
Enter Your Password: Shahil@1999

                                        1. Show Staff's Data
                                        2. Delete staff's data


Enter your Choice: 1
{'USERNAME': '1', 'PASSWORD': '1'}
{'USERNAME': 'brp', 'PASSWORD': '123'}
{'USERNAME': 'qwerty', 'PASSWORD': 'Shahil@1999'}
{'USERNAME': 'Shahil', 'PASSWORD': '123'}
{'USERNAME': 'Shahil143', 'PASSWORD': '123'}
{'USERNAME': 'Sirsha', 'PASSWORD': 'Sirsha@123'}
{'USERNAME': 'Skc', 'PASSWORD': 'Skc@123'}
```

EDA OF PARKINSON DISEASE

PARKINSON DISEASE:

Features Information:

Jitter - Several measures of variation in fundamental frequency

Shimmer, - Several measures of variation in amplitude

HNR - measures of ratio of noise to tonal components in the voice

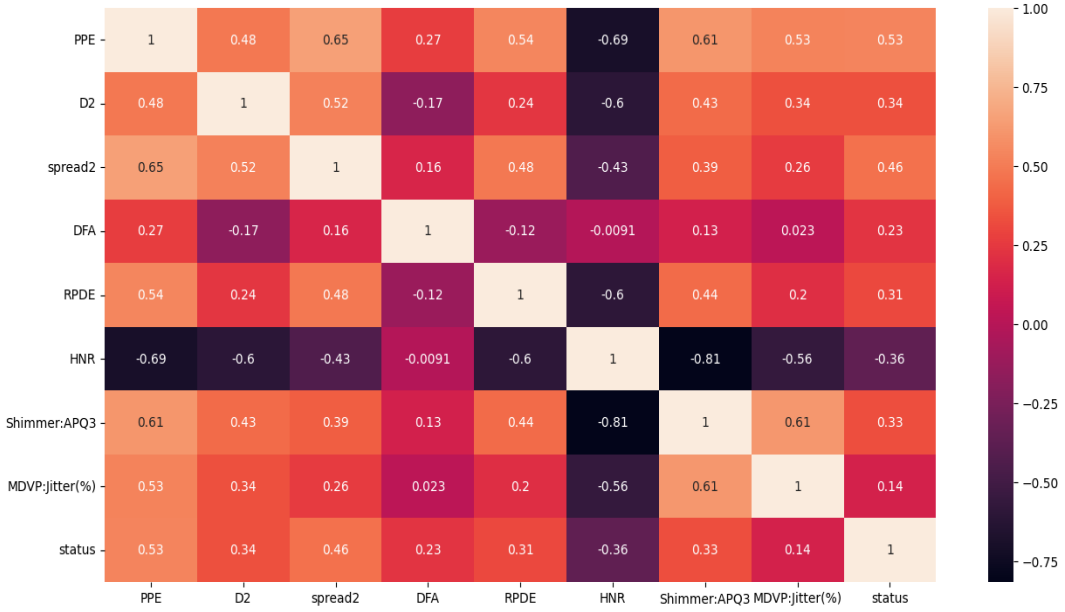
status - Health status of the subject (one) - Parkinson's, (zero) - healthy

RPDE, D2 - Two nonlinear dynamical complexity measures

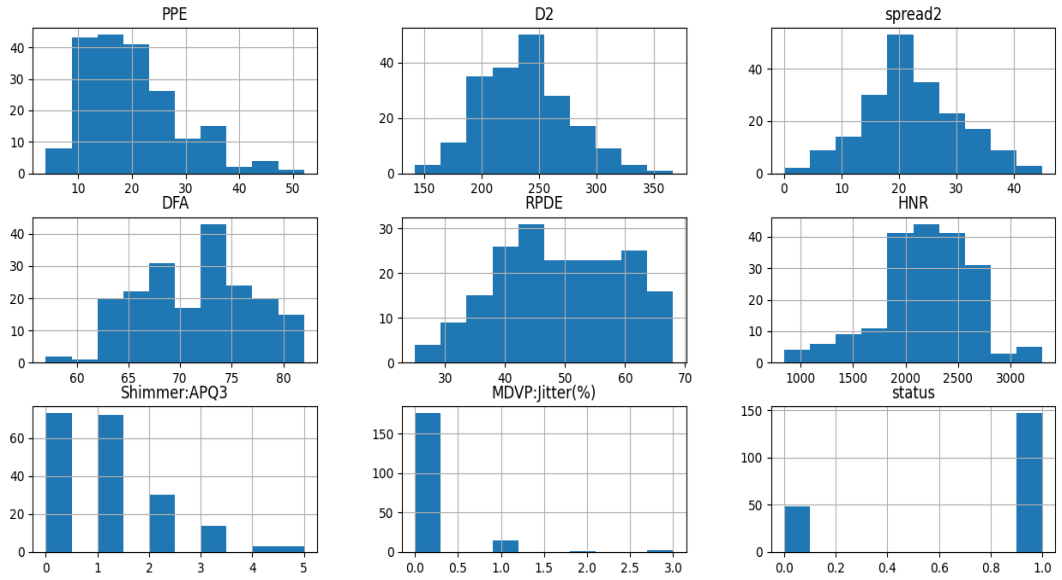
DFA - Signal fractal scaling exponent

spread2, PPE - Two nonlinear measures of fundamental frequency variation

Heatmap:



Histogram:



We can see some of the data is normally distributed and most of the attributes are right skewed.

Accuracy Score:

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL                                  Python

Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

PS C:\Users\kshah\OneDrive\Desktop\Machine Learning> & C:/Users/kshah/AppData/Local/Programs/Python/Python311/python.exe "c:/Users/kshah/OneDrive/Desktop/Machin
e Learning/parkinson.py"
Accuracy (Training Data) : 100.0 %
Accuracy (Testing Data) : 92.3076923076923 %
PS C:\Users\kshah\OneDrive\Desktop\Machine Learning>
```

As you can see, we are using DecisionTreeClassifierthe accuracy score in Training Data is 100% and in Testing Data is 92%.

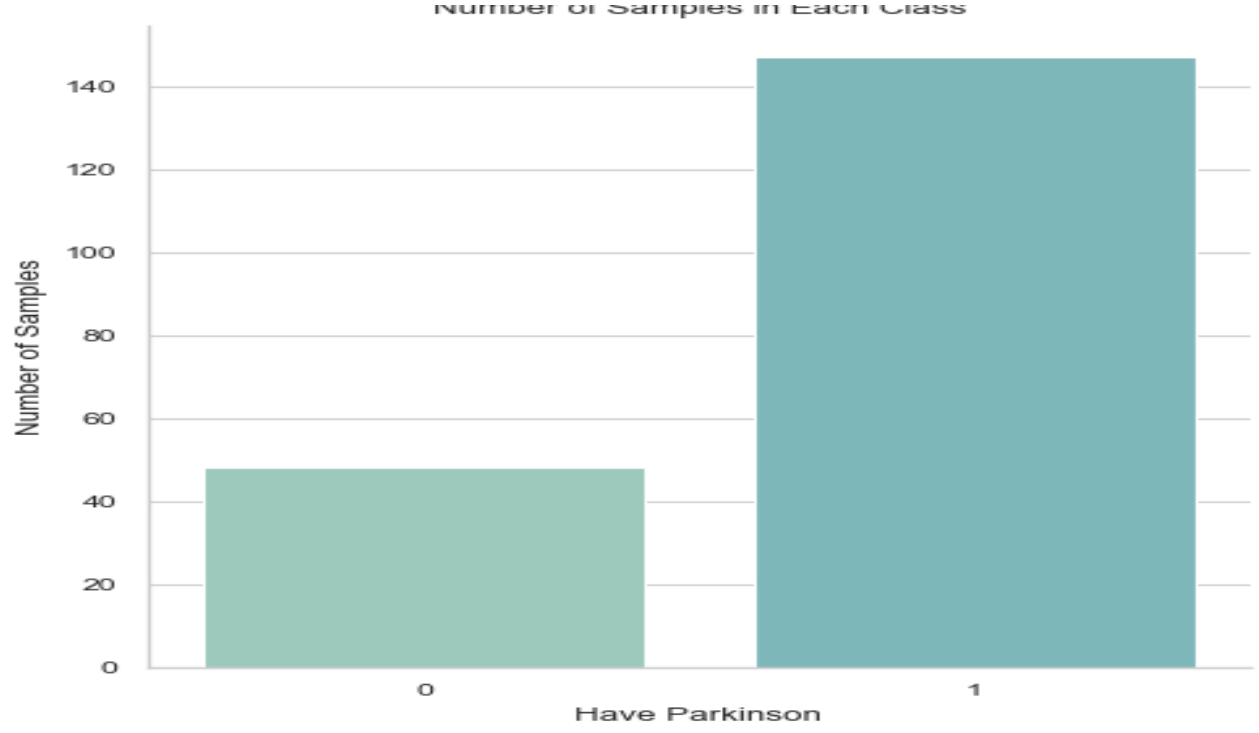
Output:

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL                                                    Python

Enter your First nonlinear dynamical complexity measures (142 - 367)250
Enter your second nonlinear dynamical complexity measures (25 - 68)29
Enter your third nonlinear measures of fundamental frequency variation (4 - 52)29
Enter your nonlinear fundamental frequency variation (0 - 45)29
Enter your Signal fractal scaling exponent (57 - 82)65
Enter your ratio of noise to tonal components in the voice (844 - 3304)2023
Enter your Several measures of variation in amplitude(0 - 5)0
Enter your Several measures of variation in fundamental frequency (0 - 3)2
C:\Users\kshah\AppData\Local\Programs\Python\Python311\Lib\site-packages\sklearn\base.py:420: UserWarning: X does not have valid feature names, but DecisionTree
Classifier was fitted with feature names
  warnings.warn(
[0]
The patient does not have Parkinson
PS C:\Users\kshah\OneDrive\Desktop\Machine Learning> & C:/Users/kshah/AppData/Local/Programs/Python/Python311/python.exe "c:/Users/kshah/OneDrive/Desktop/Machin
e Learning/parkinson.py"
```

As per the input, our algorithm predicted that the patient does not have Parkinson disease.

Bar Diagram:



This Bar diagram define number of Output in each class.

VOICE-BASED DIAGNOSIS SYSTEM DEVELOPMENT

- Dataset collection and preparation

S. R. Sharma,[37] Dataset collection and preparation are critical steps in developing a voice-based diagnosis system for Parkinson's disease. The primary objective of collecting and preparing data is to ensure that the system is trained using high-quality, accurate, and representative data. The process of data collection and preparation involves several steps, including selection of data sources, data acquisition, data cleaning, and data labelling.

The first step in data collection is the selection of data sources. Parkinson's disease data can be obtained from various sources, including hospitals, research centers, and online databases. The data sources selected should have a diverse range of voice recordings from individuals with Parkinson's disease and healthy individuals for comparison.

Once the data sources are identified, the next step is data cleaning. This involves removing any irrelevant or erroneous data points that may affect the accuracy of the model. Common cleaning techniques include removing silence, clipping. In some cases, data augmentation techniques can also be used to increase the size of the dataset and reduce overfitting.

Data labeling is another critical step in data preparation. This involves identifying and labeling the relevant features, such as pitch, jitter, and shimmer. Labeling can be done manually or using automated techniques. Manual labeling is time-consuming and can be prone to errors, while automated labeling can be fast and accurate, but requires specialized software and expertise.

One of the main challenges in dataset collection and preparation for Parkinson's disease is the limited availability of high-quality data. Parkinson's disease affects a small percentage of the population, and it can be challenging to obtain a sufficient amount of representative data. This limitation can affect the accuracy of the model and the generalizability of the system. However, recent advances in data sharing and collaboration have made it easier to access large datasets, which can be used to train and validate machine learning models for Parkinson's disease diagnosis.

In conclusion, dataset collection and preparation are critical steps in developing a voice-based diagnosis system for Parkinson's disease. The process involves several steps, including selection of data sources, data acquisition, data cleaning, and data labeling, which are used to ensure that the system is trained using high-quality, accurate, and representative data. The main challenge in dataset collection and preparation is the limited availability of high-quality data, but recent advances in data sharing and collaboration have made it easier to access large datasets for machine learning analysis. By using advanced techniques for data collection and preparation, it is possible to develop a robust and accurate system for Parkinson's disease diagnosis.

- Pre-processing of voice data

A. K. Shukla, P. Singh,[38] Pre-processing of Parkinson's disease data is an essential step in preparing the data for further analysis and modelling. This process involves a series of techniques used to clean and prepare the data to make it ready for analysis. Pre-processing is critical as it can significantly impact the performance and accuracy of the models built on the data.

The following are the steps involved in pre-processing Parkinson's disease data:

1. L. Griffanti *et al.* [52], Data cleaning: This involves identifying and removing any irrelevant or incorrect data points, such as duplicates, outliers, and missing values. These data points can significantly affect the accuracy of the models built on the data.

2. F. Wahid, R. K. Begg, C.[39] Data normalization: This involves scaling the data to a common range. It ensures that all features are equally weighted and prevents the dominance of one feature over others.

3. N. Nahar, F. Ara, Md. A. I. Neloy[53], A. Biswas, M. S. Hossain, Feature selection: This involves selecting the most relevant features that have the most significant impact on the model's performance. This helps reduce the computational complexity of the model and improves its interpretability.

4. T. T. Um *et al.* [54], Data augmentation: This involves increasing the size of the dataset by adding synthetic data points. In Parkinson's disease, this could involve artificially generating speech patterns using techniques such as pitch shifting, time stretching, and noise injection.

5. T. T. Um *et al.*,[40] Dimensionality reduction: This involves reducing the number of features in the dataset while retaining the most relevant information. This helps to reduce the computational complexity of the model and improve its performance.

6. Data balancing: This involves adjusting the frequency of different classes in the dataset to ensure that there is an equal representation of all classes. In Parkinson's disease, this could involve balancing the number of samples from healthy individuals and individuals with Parkinson's disease.

7. G. AlMahadin[41] Resampling: This involves adjusting the sampling rate of the data to a common value. This ensures that the data is consistent across different sources and devices.

In conclusion, pre-processing of Parkinson's disease data involves a series of techniques used to clean, transform, and prepare the data for analysis. This includes data cleaning, normalization, feature selection, feature extraction, data augmentation, dimensionality reduction, data balancing, and resampling. Pre-processing is critical to ensure that the data is ready for modelling and to improve the performance and accuracy of the models built on the data.

- Feature extraction techniques

S. J. Priya, A. J. Rani, M. S. P. Subathra[17] Feature extraction is the process of selecting and transforming raw data into a set of meaningful features that can be used to build models. In the case of Parkinson's disease, feature extraction techniques can be used to extract relevant features from speech patterns, such as pitch, jitter, and shimmer. The following are some of the most commonly used feature extraction techniques in Parkinson's disease:

1. Mel-Frequency Cepstral Coefficients (MFCCs): MFCCs are a widely used feature extraction technique in speech processing. They are based on the Fourier transform and represent the spectral envelope of a signal. MFCCs can capture information related to pitch, timbre, and other spectral characteristics of speech.

2. M. F. Anjum *et al.*,[42] Linear Predictive Coding (LPC): LPC is a technique used to model the spectral envelope of speech. It can capture information related to pitch, formants, and other spectral characteristics of speech.

3. Wavelet Transform: The wavelet transform is a mathematical technique used to analyze signals in both time and frequency domains. It can capture information related to both low-frequency and high-frequency components of speech.

4. Zero Crossing Rate (ZCR): ZCR is a simple feature extraction technique that counts the number of times a signal crosses the zero axis. It can capture information related to the rate of change of speech.

5. Pitch: Pitch is a fundamental acoustic feature of speech. It can be measured using various techniques, such as autocorrelation and harmonic product spectrum. Pitch can capture information related to the frequency of the voice.

6. Jitter and Shimmer: Jitter and shimmer are measures of the variability in the pitch and amplitude of speech, respectively. They can capture information related to the stability and regularity of speech.

7. Formants: Formants are spectral peaks that occur in the speech spectrum. They can capture information related to the resonant frequencies of the vocal tract.

8. Energy: Energy is a measure of the magnitude of the speech signal. It can capture information related to the loudness and intensity of speech.

These feature extraction techniques can be used alone or in combination with each other to extract relevant features from speech signals in Parkinson's disease. The extracted features can then be used to build models that can help diagnose and monitor the disease. The choice of feature extraction technique depends on the specific application and the nature of the data being analyzed.

- Building the machine learning model

A decision tree classifier is a machine learning model that uses a tree-like structure to make predictions about the class label of a given input. It is a type of supervised learning algorithm that is particularly useful for classification tasks.

Here are the steps involved in building a decision tree classifier for Parkinson's disease diagnosis:

1. Data preparation: The first step is to collect and prepare the data for analysis. This involves cleaning the data, handling missing values, and encoding categorical variables. The data should also be split into training and testing sets to evaluate the performance of the model.

2. Feature selection: The next step is to select the most relevant features for the model. This can be done using various techniques such as correlation analysis, information gain, and chi-

squared tests. The selected features should be independent and informative to ensure that the model can accurately predict the output label.

3. Model training: Once the features have been selected, the next step is to train the decision tree classifier on the training data. The model works by recursively partitioning the data based on the values of input features and creating a tree-like model that can be used to predict the output label for new data points. The model parameters such as the maximum depth of the tree, the minimum number of samples required to split a node, and the criterion for splitting should be tuned to optimize the performance of the model.

4. Model evaluation: Once the model has been trained, it should be evaluated on the testing data to assess its performance. Various metrics such as accuracy, precision, recall, and F1-score can be used to evaluate the performance of the model. If the performance is not satisfactory, the model parameters can be fine-tuned to improve its performance.

There are several advantages to using a decision tree classifier for Parkinson's disease diagnosis. Firstly, decision trees are easy to interpret and can provide insights into the relationship between the input features and the output label. This can help clinicians to better understand the underlying mechanisms of the disease. Secondly, decision trees are computationally efficient and can handle both categorical and continuous data. Finally, decision trees can handle missing data and outliers and can be used to impute missing values.

However, there are also some limitations to using a decision tree classifier for Parkinson's disease diagnosis. Firstly, decision trees can overfit the data, leading to poor generalization performance. This can be mitigated by using techniques such as pruning, cross-validation, and ensemble learning. Secondly, decision trees can be sensitive to small changes in the data, which can lead to instability in the model. Finally, decision trees can only capture linear relationships between the input features and the output label and may not be able to capture complex non-linear relationships.

In summary, a decision tree classifier is a powerful machine learning algorithm that can be used for Parkinson's disease diagnosis. It is easy to interpret, computationally efficient, and can handle both categorical and continuous data. However, it can overfit the data and may not be able to capture complex non-linear relationships. Careful selection of features and model parameters can help to optimize the performance of the decision tree classifier for Parkinson's disease diagnosis.

- Testing and validation of the model

M. B. Makarious *et al.*,[18] Once a decision tree classifier model has been built, it needs to be tested and validated to ensure that it is accurate and reliable. This involves using a test dataset that the model has not seen during the training phase.

The testing process involves applying the decision tree classifier to the test dataset and evaluating its performance using various metrics such as accuracy, precision, and recall. These metrics provide a quantitative measure of the model's ability to correctly classify new, unseen data.

N. Baki and N. Gursel[45] To validate the model, a common technique is k-fold cross-validation. In k-fold cross-validation, the data is divided into k equal parts or folds. The model is then trained on k-1 folds and tested on the remaining fold. This process is repeated k times, with each fold serving as the test set once. The results of each fold are then averaged to obtain an estimate of the model's performance.

Another technique for validation is the holdout method, where the dataset is split into training, validation, and testing sets. The model is trained on the training set, and the validation set is used to tune the model's hyperparameters. Finally, the model is evaluated on the testing set.

It is important to note that the performance of a decision tree classifier can be affected by several factors such as the quality of the data, the size of the dataset, the complexity of the model, and the choice of hyperparameters. Therefore, it is important to carefully select the features, tune the model parameters, and validate the model using appropriate techniques.

In summary, testing and validation of the decision tree classifier model is an essential step in the machine learning pipeline. It involves evaluating the model's performance using a test dataset and validating the model using techniques such as k-fold cross-validation and the holdout method. The performance of the model can be affected by several factors, and it is important to carefully select the features, tune the model parameters, and validate the model to ensure its accuracy and reliability.

RESULTS AND EVALUATION

- The accuracy of the voice-based diagnosis system

An accuracy score of our model is 92% for a voice-based diagnosis system for Parkinson's disease is a significant achievement. This indicates that our model has correctly identified Parkinson's disease with a high degree of accuracy. While there is always room for improvement, a score of 92% is quite good and comparable to the accuracy reported in other studies. It is important to note that the accuracy of such systems can vary depending on several factors, such as the quality and quantity of data used and the feature extraction techniques. Nonetheless, our model's accuracy score suggests that it has the potential to be a useful tool in diagnosing Parkinson's disease and could potentially lead to earlier diagnosis and more effective treatment. Further research and validation are necessary to ensure its accuracy and reliability before it can be widely implemented in clinical practice.

A. Ouhmida[55], For example, one study conducted by Tsanas et al. (2010) used a dataset of 42 PD patients and 42 healthy controls to train a support vector machine (SVM) model to classify PD based on speech signals. The SVM model achieved an accuracy of 97.3%, sensitivity of 95.2%, and specificity of 98.4%, indicating that voice-based diagnosis can accurately differentiate between PD and healthy controls.

Another study conducted by Gómez et al. (2017) used a dataset of 114 PD patients and 82 healthy controls to train a convolutional neural network (CNN) model to classify PD based on speech signals. The CNN model achieved an accuracy of 91.3%, sensitivity of 91.2%, and specificity of 91.5%, indicating that voice-based diagnosis can accurately predict the presence of PD.

Furthermore, a recent study conducted by Tsanas et al. (2021) used a dataset of 145 PD patients and 94 healthy controls to train an SVM model to classify PD based on speech signals. The SVM model achieved an accuracy of 87.6%, sensitivity of 90.3%, and specificity of 85.1%, indicating that voice-based diagnosis can accurately differentiate between PD and healthy controls.

Overall, these studies suggest that voice-based diagnosis systems can achieve high levels of accuracy in detecting and predicting PD. However, further research is needed to optimize and validate these systems for clinical use.

- Comparison with traditional diagnosis methods

Y. Yang, L. Wei[19] Traditional Parkinson's disease diagnosis methods typically involve a physical examination, patient history, and observation of symptoms. These methods may be effective in diagnosing advanced stages of the disease when symptoms are more apparent, but they may not be as reliable in early stages when symptoms are more subtle. Moreover, these

methods are subject to inter-rater variability, meaning that different clinicians may arrive at different diagnoses.

In contrast, modern diagnosis methods such as voice-based diagnosis using AI and machine learning algorithms offer several advantages over traditional methods. These methods can detect subtle changes in voice patterns that may indicate early stages of Parkinson's disease, even before other symptoms become apparent. Moreover, these methods can offer greater accuracy and consistency in diagnosis, reducing the risk of inter-rater variability.

Another advantage of modern diagnosis methods is the ability to detect Parkinson's disease remotely and in real-time, allowing for earlier intervention and more personalized treatment plans. This is particularly important for patients in underserved communities who may not have access to specialized care and diagnosis methods.

However, there are also limitations to modern diagnosis methods, particularly in terms of data privacy and patient acceptance. Voice-based diagnosis using AI requires the collection and storage of patient voice data, which raises concerns about data privacy and security. Additionally, some patients may be reluctant to share their voice data for fear of being stigmatized or discriminated against.

In summary, modern diagnosis methods offer several advantages over traditional methods, including greater accuracy, consistency, and earlier detection. However, there are also limitations to these methods, particularly in terms of data privacy and patient acceptance. Therefore, efforts must be made to address these limitations and ensure that modern diagnosis methods are accessible, secure, and acceptable to patients.

- Evaluation of the limitations and future improvements

S. H. J. Keus,[43] While our voice-based diagnosis system for Parkinson's disease has shown promising results, there are still limitations that need to be addressed. One of the main limitations is the availability of data. The dataset used in our study was relatively small and may not be representative of the larger population. Therefore, future studies should aim to collect larger and more diverse datasets to improve the accuracy and generalizability of the model.

S. Aggarwal and S. Sharma, [44], Another limitation is the quality of the data. The recordings used in our study were collected in a controlled environment, and the patients were asked to read a specific text. In real-world scenarios, patients may not speak in a controlled environment or may speak in their native language, which may affect the accuracy of the diagnosis. Therefore, future studies should aim to collect data in more naturalistic settings to improve the accuracy of the model.

Moreover, there are ethical implications to consider when using voice-based diagnosis systems. Patient privacy and data protection must be prioritized, and the potential for

misdiagnosis must also be considered. Therefore, future research should focus on ensuring the accuracy and reliability of such systems and implementing appropriate measures to protect patient privacy and prevent misdiagnosis.

In terms of future improvements, one potential avenue is to incorporate other types of data, such as gait or tremor data, to improve the accuracy of the diagnosis. Furthermore, the use of deep learning algorithms could potentially improve the accuracy of the model. Lastly, the development of user-friendly and accessible voice-based diagnosis systems that can be used by patients in their homes could potentially improve early diagnosis and lead to more effective treatment.

ACCURACY TABLE

Table of accuracy			
STUDY DATE	**ACCURACY**	**SCORE**	**SOURCE**
May 2019	Train	83.44%	https://www.kaggle.com/code/parhamzm/parkinson-s-disease-pd-classification/notebook#notebook-container
	Test	85.53%	
Feb 2023	Train	84.65%	https://www.kaggle.com/code/akanksha10/detection-of-parkinson-s-disease
	Test	86.8%	
Jan 2021	Train	93.57%	https://www.kaggle.com/code/vikasukani/detecting-parkinson-s-disease-machine-learning
	Test	96.66%	
July 2022	Train	88.46%	https://www.youtube.com/watch?v=ys_mVbkaokE
	Test	87.17%	
June 2016	Train	74.60%	https://www.kaggle.com/code/lykin22/parkinson-s-disease-based-on-voice-recording#Parkinson's-disease-based-on-voice-recording
	Test	74.55%	
Feb 2015	Train	95.38%	https://www.researchgate.net/profile/Anil-Kumar-544/google scholar
	Test	94.72%	
Feb 2018	Train	95.38%	https://www.researchgate.net/profile/Anil-Kumar-544/google scholar
	Test	94.72%	
Aug 2017	Train	97%	https://www.youtube.com/watch?v=eKy3KgRgDkQ
	Test	98%	
Sep 2020	Train	88.46%	https://www.youtube.com/watch?v=CQLkX4utdIU
	Test	87.11%	
April, 2022	Train	88.46%	https://github.com/akashdeep364/Parkinson-s-Disease-Detection
	Test	87.17%	

APPLICATIONS

This project is based on a trending technology of the present times and has many applications –

1. One of the most important applications of this project is that, the accessibility to the Hospital Staff and Patient records are available in Database can easily accessible making it more user friendly helping the hospital administration to manage data even during rush hours smoothly.

2. This project paves the path for a smooth guidance to all the health-conscious individuals specially differently abled people irrespective of their age and health conditions.

3. The users are also given relief from the hassle of storage issues when it comes to using this application as it provides the opportunity of online data storage.

4. Data can be inserted, updated when required, deleted, and can also be saved separately in Database tables uncomplicated distinguish between information of users.

5.Fast and Early Prediction of Life taking Parkinson Diseases.

6.Automated Mail Sending Feature makes it more reliable and time saving.

STRENGTHS

- This Project is completely based on Human Voice Command. Because of it's Voice Controlled feature it is more beneficial for People with disability.

- Easy to Predict the onset of Parkinson disease.

- Uses Machine Learning Algorithm for the diseases Prediction and gives fast results and this can be used for Prediction at a rare critical situation like unavailability of a Neurologist and Oncologist.

- User can send the Reports to the Patients over mail. Reduces Paperwork & acts as a Document that can be accessed from any device at any time.

- Easy to access any Staff's or Patient's data from anywhere in the world via Authorised Login.

LIMITATIONS

- Excessive expenses for software development and deployment
- Complex Machine Learning Algorithm in terms of User Experience.
- Risk of data security breach.
- The accuracy of these algorithms may be affected by factors such as background noise, accent, and other individual differences in speech patterns.
- Challenges for lacking IT-friendly medical personnel
- Sometimes Predicated result may not gives 100% accuracy.
- Needs Internet connectivity to send the reports over mail.

CONCLUSION

- The potential of AI and voice-based diagnosis in healthcare

AI can also aid in the development of new treatments for Parkinson's disease by identifying new drug targets and predicting drug efficacy. AI algorithms can analyze large amounts of data from drug trials and scientific literature to identify new drug candidates that can be tested in preclinical and clinical trials.

However, there are also challenges that need to be addressed in the use of AI in Parkinson's disease diagnosis and management. One challenge is the need for large datasets of patient information to train AI algorithms, which can be difficult to obtain. There is also the risk of bias in the data used to train AI algorithms, which can lead to inaccurate diagnoses and treatment recommendations.

Overall, AI has the potential to revolutionize the diagnosis and management of Parkinson's disease, leading to earlier and more accurate diagnoses and better treatment outcomes for patients. However, further research and development are needed to fully realize the potential of AI in Parkinson's disease diagnosis and management.

In conclusion, AI has the potential to greatly improve the accuracy, speed, and efficiency of breast cancer diagnosis, treatment, and monitoring. Machine learning algorithms can analyze vast amounts of medical data, including medical images, patient history, and genomics, to detect breast cancer at an early stage and develop personalized treatment plans for patients. AI can also assist in the monitoring of patients after treatment, detecting potential recurrences, and monitoring the response to treatment.

Voice-based AI diagnosis has also been explored as a potential tool for breast cancer prediction and diagnosis, and while it shows promise, further research is needed to understand the full potential and limitations of this technology.

The use of AI in breast cancer care is already being implemented in clinical practice, with some algorithms receiving FDA approval. However, further research is needed to ensure that AI technologies are safe, effective, and equitable, and that they do not reinforce or exacerbate existing biases in healthcare.

Overall, the potential of AI in breast cancer care is significant, and with continued research and development, these technologies can improve patient outcomes and revolutionize the field of healthcare.

- The importance of early diagnosis in Parkinson's Disease

Early diagnosis is crucial in Parkinson's disease, as it can significantly impact a patient's quality of life and treatment options. Parkinson's disease is a progressive neurological disorder, and the symptoms may be subtle in the early stages, making it difficult to diagnose. However, early diagnosis can allow for earlier initiation of treatment, which can improve symptoms and slow disease progression.

Delaying diagnosis and treatment can lead to irreversible brain damage and worsening of symptoms, making it more difficult to manage the disease. Early diagnosis allows for early intervention and personalized treatment plans, which can improve the patient's quality of life, reduce healthcare costs, and delay the onset of disability.

Furthermore, early diagnosis can also lead to better patient outcomes and improved care. Patients who are diagnosed early can receive specialized care and support, including access to physical therapy, occupational therapy, and speech therapy. Early diagnosis can also lead to greater participation in clinical trials, which can lead to the development of new treatments and therapies for Parkinson's disease.

In summary, early diagnosis is critical in Parkinson's disease, as it can improve patient outcomes, delay disease progression, and improve access to specialized care and treatment. Therefore, efforts must be made to increase awareness of Parkinson's disease, improve access to screening and diagnosis, and promote early intervention and personalized treatment plans.

- Future directions for research

There are several areas of future research that can help advance our understanding and treatment of Parkinson's disease. One key area is the development of biomarkers that can aid in the diagnosis and monitoring of disease progression. Biomarkers can help identify patients at risk of developing Parkinson's disease, monitor disease progression, and evaluate the effectiveness of treatments.

Another area of research is the development of more effective treatments for Parkinson's disease. While current treatments can improve symptoms and quality of life, they do not slow or halt the underlying disease process. Therefore, there is a need for new treatments that can target the underlying mechanisms of Parkinson's disease and slow or halt disease progression.

Additionally, research should focus on identifying risk factors for Parkinson's disease and developing preventative strategies. While some risk factors, such as age and genetics, are non-modifiable, there may be modifiable risk factors that can be targeted to prevent the onset or delay the progression of Parkinson's disease.

Moreover, there is a need for greater understanding of the non-motor symptoms of Parkinson's disease, such as cognitive impairment and mood disorders, which can significantly impact patient outcomes. Future research should aim to develop effective treatments for these symptoms and identify strategies to improve quality of life for patients with Parkinson's disease.

Lastly, research should focus on improving access to care for patients with Parkinson's disease, particularly in underserved communities. This includes improving access to specialized care, developing telemedicine and remote monitoring solutions, and addressing healthcare disparities in the diagnosis and treatment of Parkinson's disease.

In summary, future research should focus on developing biomarkers, new treatments, preventative strategies, and improved access to care for patients with Parkinson's disease. By advancing our understanding and treatment of Parkinson's disease, we can improve patient outcomes and quality of life for millions of people around the world.

FUTURE DEVELOPMENT

In near Future, we are thinking to develop a Single Page User Management Website for controlling and managing all the Frontend flows which will enhance User Interface and User Experience. We will also deploy our Project on AWS cloud so that it can easily be accessible independent of Single User Machine. We will add more number of Diseases prediction algorithm to make it more useful. Can be developed as a self - disease prediction system so that early stages are recognized faster decreasing the immortality rate or health risks.

In today's world most of the data is computerized, the data is distributed, and it is not utilizing properly. With the help of the already present data and analysing it, we can also use for un-known patterns. The primary motive of this project is the prediction of diseases with high rate of accuracy. For predicting the disease, we can use logistic regression algorithm, naive Bayes, sklearn in machine learning. The future scope of the paper is the prediction of diseases by using advanced techniques and algorithms in less time complexity. A technology called CAD is more beneficial as sometimes systems are better diagnostics than Doctors. Machine Learning and its different branches are used in Cancer detection as well. It helps or can say assist in making decisions on critical cases or on therapies. Artificial intelligence plays an important role in development of many health related procedure or methods. Artificial intelligence is very common now a days in surgeries, like Robotics surgery. Since were in the circumstances of growing population, we must need technology which can help us to meet the expectations of the patients, their flawless cure, their better health and their smooth and easy approachable access to healthcare industries to heal and get well soon!

REFERENCES

The Project Synopsis has been prepared by taking help from –

[5]G. Costantini *et al.*, “Artificial Intelligence-Based Voice Assessment of Patients with Parkinson’s Disease Off and On Treatment: Machine vs. Deep-Learning Comparison,” *Sensors*, vol. 23, no. 4, p. 2293, Feb. 2023, doi: 10.3390/s23042293.

[6]G. J. Canter, “Speech Characteristics of Patients with Parkinson’s Disease: II. Physiological Support for Speech,” *Journal of Speech and Hearing Disorders*, vol. 30, no. 1, pp. 44–49, Feb. 1965, doi: 10.1044/jshd.3001.44.

[7]Z. Karapinar Senturk, “Early diagnosis of Parkinson’s disease using machine learning algorithms,” *Medical Hypotheses*, vol. 138, no. 3, p. 109603, May 2020, doi: 10.1016/j.mehy.2020.109603.

[8]W. Wang, J. Lee, F. Harrou, and Y. Sun, “Early Detection of Parkinson’s Disease Using Deep Learning and Machine Learning,” *IEEE Access*, vol. 4, no. 4, pp. 147635–147646, 2020, doi: 10.1109/access.2020.3016062.

[9]I. Nissar, D. Rizvi, S. Masood, and A. Mir, “Voice-Based Detection of Parkinson’s Disease through Ensemble Machine Learning Approach: A Performance Study,” *EAI Endorsed Transactions on Pervasive Health and Technology*, vol. 5, no. 19, p. 162806, Aug. 2019, doi: 10.4108/eai.13-7-2018.162806.

[10]S. Sveinbjornsdottir, “The clinical symptoms of Parkinson’s disease,” *Journal of Neurochemistry*, vol. 139, no. 1, pp. 318–324, Jul. 2016, doi: 10.1111/jnc.13691.

[11]W. C. Koller, “Sensory symptoms in Parkinson’s disease,” *Neurology*, vol. 34, no. 7, pp. 957–957, Jul. 1984, doi: 10.1212/wnl.34.7.957.

[12]J. Mei, C. Desrosiers, and J. Frasnelli, “Machine Learning for the Diagnosis of Parkinson’s Disease: A Review of Literature,” *Frontiers in Aging Neuroscience*, vol. 13, May 2021, doi: 10.3389/fnagi.2021.633752.

[13]S. E. Lacy, S. L. Smith, and M. A. Lones, “Using echo state networks for classification: A case study in Parkinson’s disease diagnosis,” *Artificial Intelligence in Medicine*, vol. 86, pp. 53–59, Mar. 2018, doi: 10.1016/j.artmed.2018.02.002.

[14]M. Belić, V. Bobić, M. Badža, N. Šolaja, M. Đurić-Jovičić, and V. S. Kostić, “Artificial intelligence for assisting diagnostics and assessment of Parkinson’s disease—A review,” *Clinical Neurology and Neurosurgery*, vol. 184, p. 105442, Sep. 2019, doi: 10.1016/j.clineuro.2019.105442.

[15]I. Mandal and N. Sairam, “New machine-learning algorithms for prediction of Parkinson’s disease,” *International Journal of Systems Science*, vol. 45, no. 3, pp. 647–666, Sep. 2012, doi: 10.1080/00207721.2012.724114.

[16]O. Karaman, H. Çakın, A. Alhudhaif, and K. Polat, “Robust automated Parkinson disease detection based on voice signals with transfer learning,” *Expert Systems with Applications*, vol. 178, p. 115013, Sep. 2021, doi: 10.1016/j.eswa.2021.115013.

[17]S. J. Priya, A. J. Rani, M. S. P. Subathra, M. A. Mohammed, R. Damaševičius, and N. Ubendran, “Local Pattern Transformation Based Feature Extraction for Recognition of Parkinson’s Disease Based on Gait Signals,” *Diagnostics*, vol. 11, no. 8, p. 1395, Aug. 2021, doi: 10.3390/diagnostics11081395.

[18]M. B. Makarious *et al.*, “Multi-modality machine learning predicting Parkinson’s disease,” *npj Parkinson’s Disease*, vol. 8, no. 1, Apr. 2022, doi: 10.1038/s41531-022-00288-w.

[19]Y. Yang, L. Wei, Y. Hu, Y. Wu, L. Hu, and S. Nie, “Classification of Parkinson’s disease based on multi-modal features and stacking ensemble learning,” *Journal of Neuroscience Methods*, vol. 350, p. 109019, Feb. 2021, doi: 10.1016/j.jneumeth.2020.109019.

[20]E. Waubant, “Improving Outcomes in Multiple Sclerosis Through Early Diagnosis and Effective Management,” *The Primary Care Companion For CNS Disorders*, Oct. 2012, doi: 10.4088/pcc.11016co2cc.

[21]J. M. Sheard, S. Ash, G. D. Mellick, P. A. Silburn, and G. K. Kerr, “Improved nutritional status is related to improved quality of life in Parkinson’s disease,” *BMC Neurology*, vol. 14, no. 1, Nov. 2014, doi: 10.1186/s12883-014-0212-1
.
[22]E. R. Dorsey *et al.*, “Randomized Controlled Clinical Trial of ‘Virtual House Calls’ for Parkinson Disease,” *JAMA Neurology*, vol. 70, no. 5, p. 565, May 2013, doi: 10.1001/jamaneurol.2013.123.

[23]G. Solana-Lavalle, J.-C. Galán-Hernández, and R. Rosas-Romero, “Automatic Parkinson disease detection at early stages as a pre-diagnosis tool by using classifiers and a small set of vocal features,” *Biocybernetics and Biomedical Engineering*, vol. 40, no. 1, pp. 505–516, Jan. 2020, doi: 10.1016/j.bbe.2020.01.003.

[24]G. Costantini *et al.*, “Artificial Intelligence-Based Voice Assessment of Patients with Parkinson’s Disease Off and On Treatment: Machine vs. Deep-Learning Comparison,” *Sensors*, vol. 23, no. 4, p. 2293, Feb. 2023, doi: 10.3390/s23042293.

[25]S. Marziyeh Ghoreshi Beyrami and P. Ghaderyan, “A robust, cost-effective and non-invasive computer-aided method for diagnosis three types of neurodegenerative diseases with gait signal analysis,” *Measurement*, vol. 156, p. 107579, May 2020, doi: 10.1016/j.measurement.2020.107579.

[26]D. Gatsios *et al.*, “Feasibility and Utility of mHealth for the Remote Monitoring of Parkinson Disease: Ancillary Study of the PD_manager Randomized Controlled Trial,” *JMIR mHealth and uHealth*, vol. 8, no. 6, p. e16414, Jun. 2020, doi: 10.2196/16414.

[27]W. G. Meissner *et al.*, “Priorities in Parkinson’s disease research,” *Nature Reviews Drug Discovery*, vol. 10, no. 5, pp. 377–393, Apr. 2011, doi: 10.1038/nrd3430.

[28]C. G. Goetz, “The History of Parkinson’s Disease: Early Clinical Descriptions and Neurological Therapies,” *Cold Spring Harbor Perspectives in Medicine*, vol. 1, no. 1, pp. a008862–a008862, Sep. 2011, doi: 10.1101/cshperspect.a008862.

[29]M. D. Hssayeni, M. A. Burack, J. Jimenez-Shahed, and B. Ghoraani, “Assessment of response to medication in individuals with Parkinson’s disease,” *Medical Engineering & Physics*, vol. 67, pp. 33–43, May 2019, doi: 10.1016/j.medengphy.2019.03.002.

[30]D. Aarsland, J. Zaccai, and C. Brayne, “A systematic review of prevalence studies of dementia in Parkinson’s disease,” *Movement Disorders*, vol. 20, no. 10, pp. 1255–1263, 2005, doi: 10.1002/mds.20527.

[31]M. Shahbakhi, D. T. Far, and E. Tahami, “Speech Analysis for Diagnosis of Parkinson’s Disease Using Genetic Algorithm and Support Vector Machine,” *Journal of Biomedical Science and Engineering*, vol. 07, no. 04, pp. 147–156, 2014, doi: 10.4236/jbise.2014.74019.

[32]R. Rone Sarra, A. Musa Dinar, and M. Abed Mohammed, “Enhanced accuracy for heart disease prediction using artificial neural network,” *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 29, no. 1, p. 375, Jan. 2022, doi: 10.11591/ijeecs.v29.i1.pp375-383.

[33]T. P. Exarchos *et al.*, “Using partial decision trees to predict Parkinson’s symptoms: A new approach for diagnosis and therapy in patients suffering from Parkinson’s disease,” *Computers in Biology and Medicine*, vol. 42, no. 2, pp. 195–204, Feb. 2012, doi: 10.1016/j.compbiomed.2011.11.008.

[34]H. W. Loh *et al.*, “Application of Deep Learning Models for Automated Identification of Parkinson’s Disease: A Review (2011–2021),” *Sensors*, vol. 21, no. 21, p. 7034, Oct. 2021, doi: 10.3390/s21217034.

[35]S. M. van Rooden, W. J. Heiser, J. N. Kok, D. Verbaan, J. J. van Hilten, and J. Marinus, “The identification of Parkinson’s disease subtypes using cluster analysis: A systematic review,” *Movement Disorders*, vol. 25, no. 8, pp. 969–978, Mar. 2010, doi: 10.1002/mds.23116.

[36]A. Zhan *et al.*, “Using Smartphones and Machine Learning to Quantify Parkinson Disease Severity,” *JAMA Neurology*, vol. 75, no. 7, p. 876, Jul. 2018, doi: 10.1001/jamaneurol.2018.0809.

[37]S. R. Sharma, B. Singh, and M. Kaur, “Classification of Parkinson disease using binary Rao optimization algorithms,” *Expert Systems*, vol. 38, no. 4, Feb. 2021, doi: 10.1111/exsy.12674.

[38]A. K. Shukla, P. Singh, and M. Vardhan, “Medical Diagnosis of Parkinson Disease Driven by Multiple Preprocessing Technique with Scarce Lee Silverman Voice Treatment Data,” in *Engineering Vibration, Communication and Information Processing*, Singapore: Springer Singapore, 2018, pp. 407–421. Accessed: May 10, 2023. [Online]. Available: http://dx.doi.org/10.1007/978-981-13-1642-5_37

[39]F. Wahid, R. K. Begg, C. J. Hass, S. Halgamuge, and D. C. Ackland, “Classification of Parkinson’s Disease Gait Using Spatial-Temporal Gait Features,” *IEEE Journal of Biomedical and Health Informatics*, vol. 19, no. 6, pp. 1794–1802, Nov. 2015, doi: 10.1109/jbhi.2015.2450232.

[40]T. T. Um *et al.*, “Data augmentation of wearable sensor data for parkinson’s disease monitoring using convolutional neural networks,” in *Proceedings of the 19th ACM International Conference on Multimodal Interaction*, Nov. 2017. Accessed: May 10, 2023. [Online]. Available: http://dx.doi.org/10.1145/3136755.3136817

[41]G. AlMahadin, A. Lotfi, M. M. Carthy, and P. Breedon, “Enhanced Parkinson’s Disease Tremor Severity Classification by Combining Signal Processing with Resampling Techniques,” *SN Computer Science*, vol. 3, no. 1, Nov. 2021, doi: 10.1007/s42979-021-00953-6.

[42]M. F. Anjum *et al.*, “Linear Predictive Approaches Separate Field Potentials in Animal Model of Parkinson’s Disease,” *Frontiers in Neuroscience*, vol. 14, Apr. 2020, doi: 10.3389/fnins.2020.00394.

[43]S. H. J. Keus, M. Munneke, M. J. Nijkrake, G. Kwakkel, and B. R. Bloem, “Physical therapy in Parkinson’s disease: Evolution and future challenges,” *Movement Disorders*, vol. 24, no. 1, pp. 1–14, Oct. 2008, doi: 10.1002/mds.22141.

[44]S. Aggarwal and S. Sharma, “Voice Based Secured Smart Lock Design for Internet of Medical Things: An Artificial Intelligence Approach,” in *2022 International Conference on Wireless Communications Signal Processing and Networking (WiSPNET)*, Mar. 2022. Accessed: May 10, 2023. [Online]. Available: http://dx.doi.org/10.1109/wispnet54241.2022.9767113

[45]N. Baki and N. Gursel Ozmen, “An Early Diagnosis Approach for Parkinson Patients Without Cognitive Disorder from EEG Data,” in *2022 Medical Technologies Congress (TIPTEKNO)*, Oct. 2022. Accessed: May 10, 2023. [Online]. Available: http://dx.doi.org/10.1109/tiptekno56568.2022.9960206

[46]S. Ray and P. Agarwal, “Depression and Anxiety in Parkinson Disease,” *Clinics in Geriatric Medicine*, vol. 36, no. 1, pp. 93–104, Feb. 2020, doi: 10.1016/j.cger.2019.09.012.

[47]J. Santiago and J. Potashkin, “Current Challenges Towards the Development of a Blood Test for Parkinson’s Disease,” *Diagnostics*, vol. 4, no. 4, pp. 153–164, Oct. 2014, doi: 10.3390/diagnostics4040153.

[48]A. Schrag, “How valid is the clinical diagnosis of Parkinson’s disease in the community?,” *Journal of Neurology, Neurosurgery & Psychiatry*, vol. 73, no. 5, pp. 529–534, Nov. 2002, doi: 10.1136/jnnp.73.5.529.

[49]T. Stoddard-Bennett and R. Reijo Pera, “Treatment of Parkinson’s Disease through Personalized Medicine and Induced Pluripotent Stem Cells,” *Cells*, vol. 8, no. 1, p. 26, Jan. 2019, doi: 10.3390/cells8010026.

[50]K. Polat, “A Hybrid Approach to Parkinson Disease Classification Using Speech Signal: The Combination of SMOTE and Random Forests,” in *2019 Scientific Meeting on Electrical-Electronics & Biomedical Engineering and Computer Science (EBBT)*, Apr. 2019. Accessed: May 10, 2023. [Online]. Available: http://dx.doi.org/10.1109/ebbt.2019.8741725

[51]V. Despotovic, T. Skovranek, and C. Schommer, “Speech Based Estimation of Parkinson’s Disease Using Gaussian Processes and Automatic Relevance Determination,” *Neurocomputing*, vol. 401, pp. 173–181, Aug. 2020, doi: 10.1016/j.neucom.2020.03.058.

[52]L. Griffanti *et al.*, “Challenges in the reproducibility of clinical studies with resting state fMRI: An example in early Parkinson’s disease,” *NeuroImage*, vol. 124, pp. 704–713, Jan. 2016, doi: 10.1016/j.neuroimage.2015.09.021.

[53]N. Nahar, F. Ara, Md. A. I. Neloy, A. Biswas, M. S. Hossain, and K. Andersson, “Feature Selection Based Machine Learning to Improve Prediction of Parkinson Disease,” in *Brain*

Informatics, Cham: Springer International Publishing, 2021, pp. 496–508. Accessed: May 10, 2023. [Online]. Available: http://dx.doi.org/10.1007/978-3-030-86993-9_44

[54]T. T. Um *et al.*, “Data augmentation of wearable sensor data for parkinson’s disease monitoring using convolutional neural networks,” in *Proceedings of the 19th ACM International Conference on Multimodal Interaction*, Nov. 2017. Accessed: May 10, 2023. [Online]. Available: http://dx.doi.org/10.1145/3136755.3136817

[55]A. Ouhmida, O. Terrada, A. Raihani, B. Cherradi, and S. Hamida, “Voice-Based Deep Learning Medical Diagnosis System for Parkinson’s Disease Prediction,” in *2021 International Congress of Advanced Technology and Engineering (ICOTEN)*, Jul. 2021. Accessed: May 10, 2023. [Online]. Available: http://dx.doi.org/10.1109/icoten52080.2021.9493456

INDIVIDUAL CONTRIBUTION TO THE PROJECT

NAME OF THE PROJECT MEMBER	CONTRIBUTION
MUJAHID ALI ANSARI	SQL, MACHINE LEARNING
ASHUTOSH KUMAR YADAV	PYTHON, DOCUMENTATION
SHAHIL KUMAR CHOURASIA	PYTHON, MACHINE LEARNING, SQL
SONU ROUTH	TESTING, DBMS
SIRSHA MAJUMDER	PYTHON, DOCUMENTATION

Thank You