**Application Building**

In this section, we will be building a web application that is integrated to the model we build. A UI is provided for the users where he has to enter the values for predictions. The values which are entered will be given to the saved model and prediction is showcased on the UI.

This section has the following tasks

* Building HTML Pages
* Building server-side script

### Building HTML Pages

For this project create four HTML files namely

* about.html
* base.html
* index6.html
* info.html

and save them in templates folder

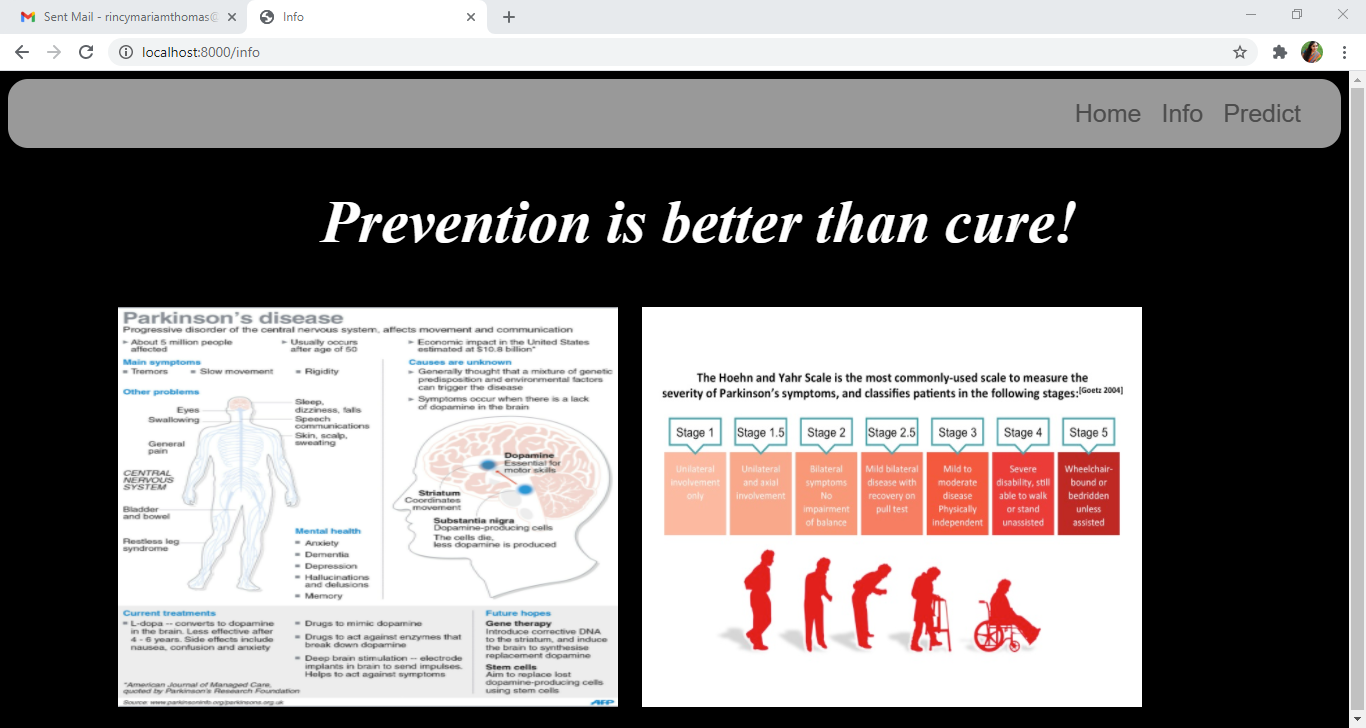
download the HTML files from the given link in the prerequisites section.

Let’s see how our home.html page looks like:



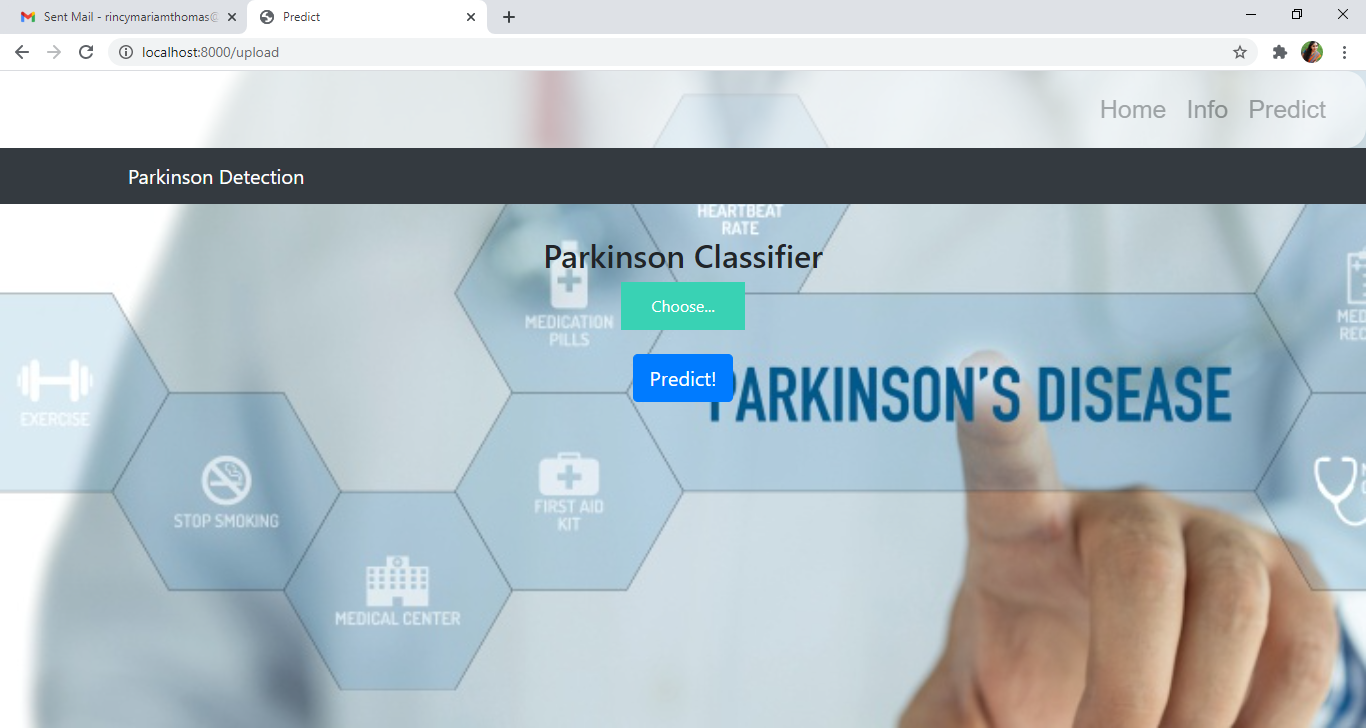
Now when you click on info button from the top right corner you will be redirected to info.html

Let's look at how our info.html file looks like:



Now when you click on predict button from the top right corner you will be redirected to index6.html

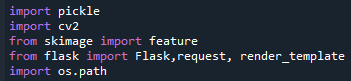
Let's look at how our index6.html file looks like:



Now from this page give the input image and see the predicted output on the UI

### Build Python Code

**Import the libraries**



Importing the flask module in the project is mandatory. Flask constructor takes the name of the current module (\_\_name\_\_) as argument.

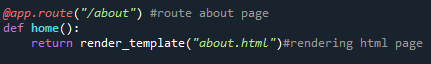


**Render HTML Page:**

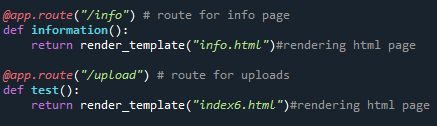


Here, declared constructor is used to route to the HTML page created earlier.

In the above example, ‘/’ URL is bound with about.html function. Hence, when the home page of the web server is opened in browser, the html page is rendered.



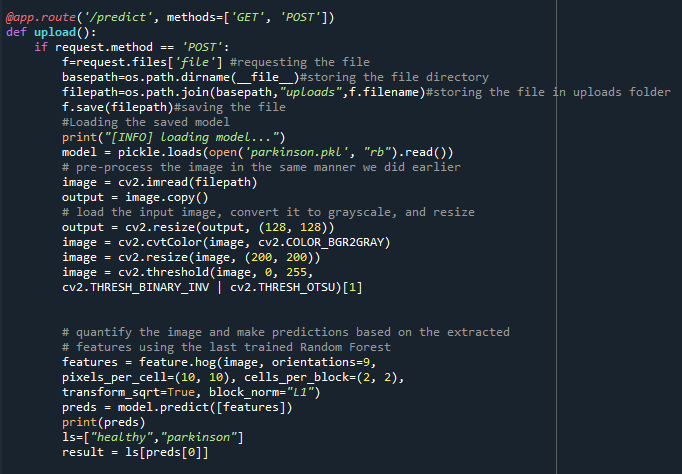
Here, **“about.html”**is rendered when home button is clicked on the UI.

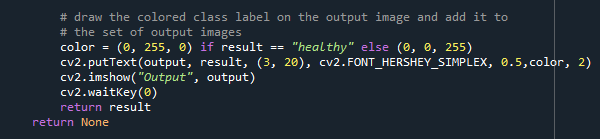


Similarly, **info.html** and **index6.html** are rendered when info and predict buttons are clicked on UI.

Retrieve the Values from UI

Whenever you give the inputs from the html page, the values can be retrieved using POST Method.





Here we are routing our app to upload() function. This function retrieves all the values from the HTML page using Post request. We are requesting to upload image using the request function.

We take the input from the user, and preprocess the image(convert to gray scale, resize, threshold)).It is necessary to preprocess the data so as give it to the model to predict the output. Once the output is predicted, result is shown on opencv window and html page.

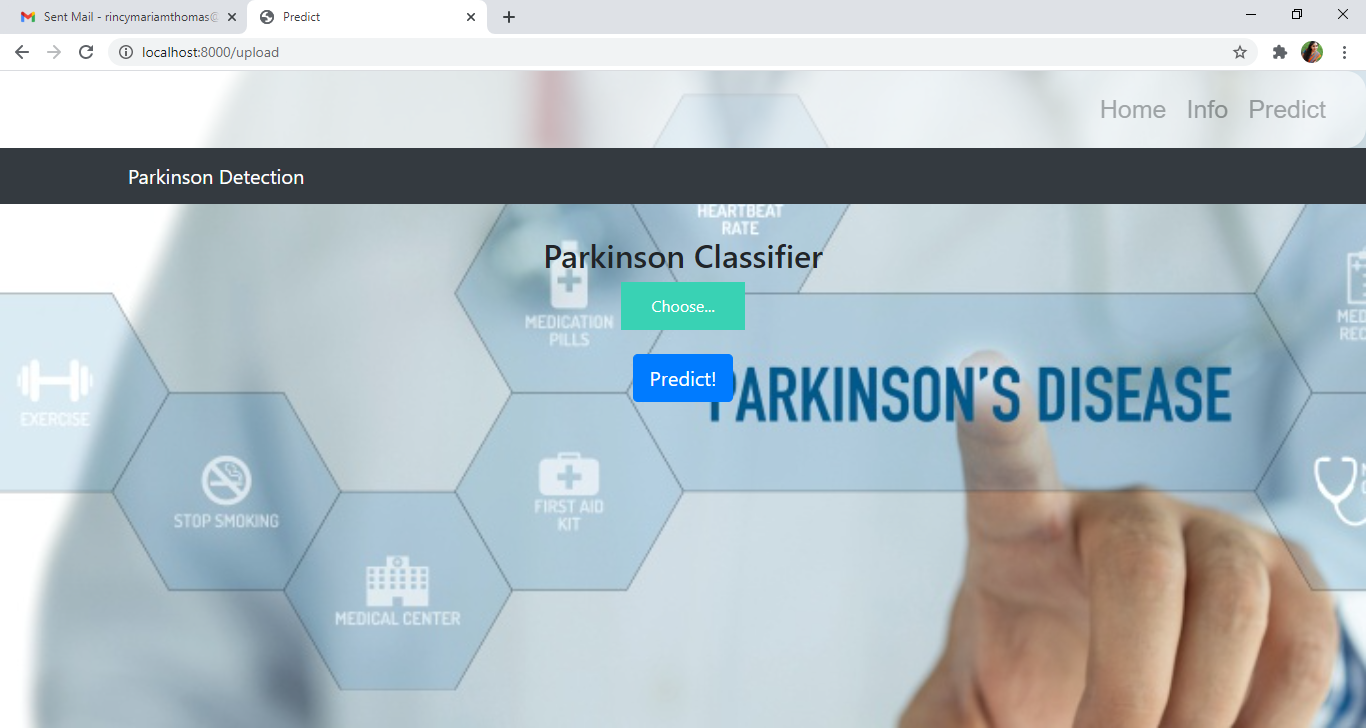
**Main Function:**



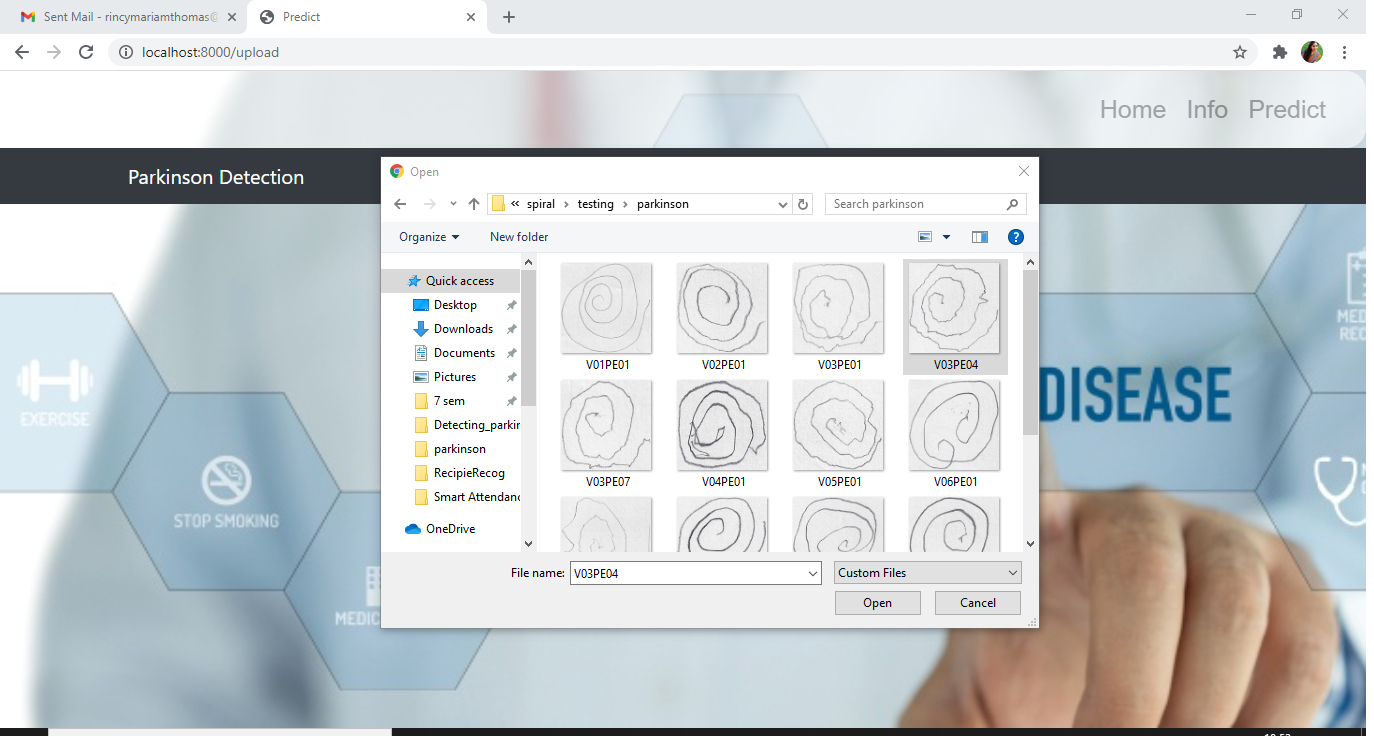
### Run The App

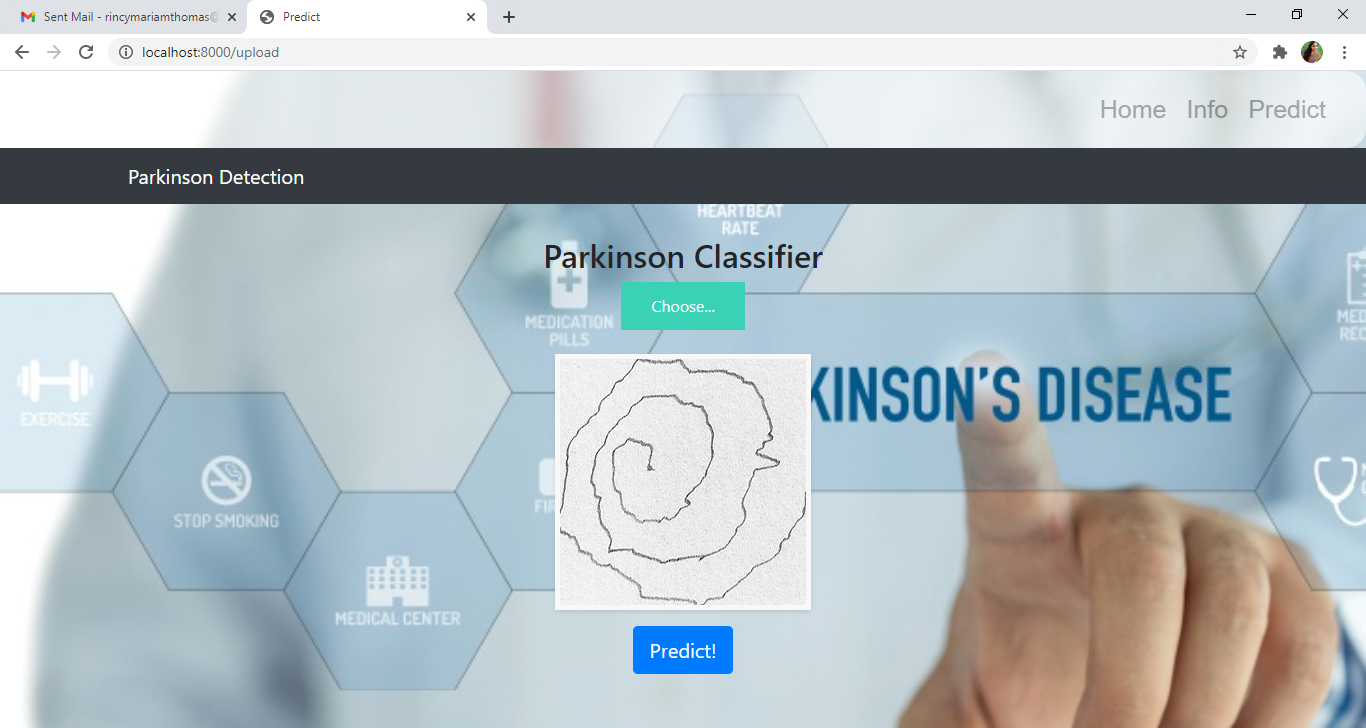
**Open anaconda prompt from the start menu.**

* Navigate to the folder where your app.py resides.
* Now type**“python app.py”**command.
* It will show the local host where your app is running (on http://localhost:8000/)
* Copy that localhost URL and open that URL in the browser. It navigates to view the web page.
* Click on the **predict button** from the top right corner.

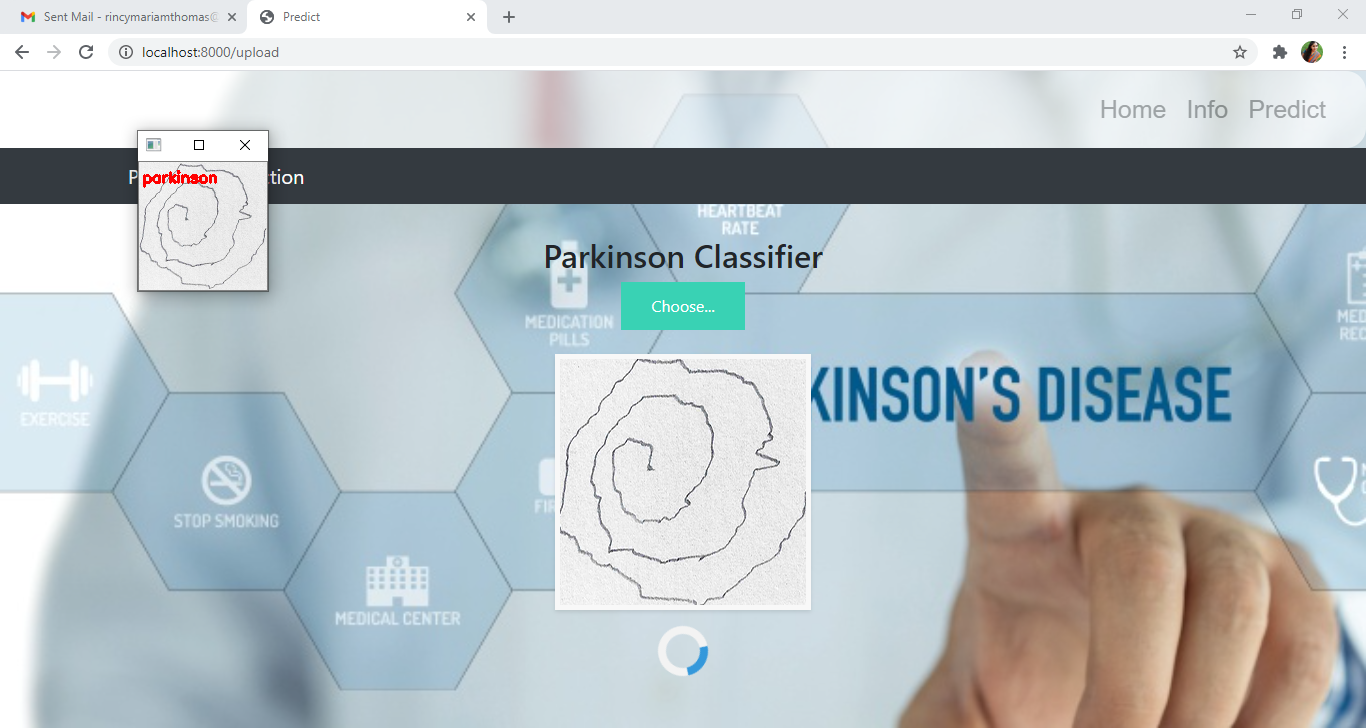


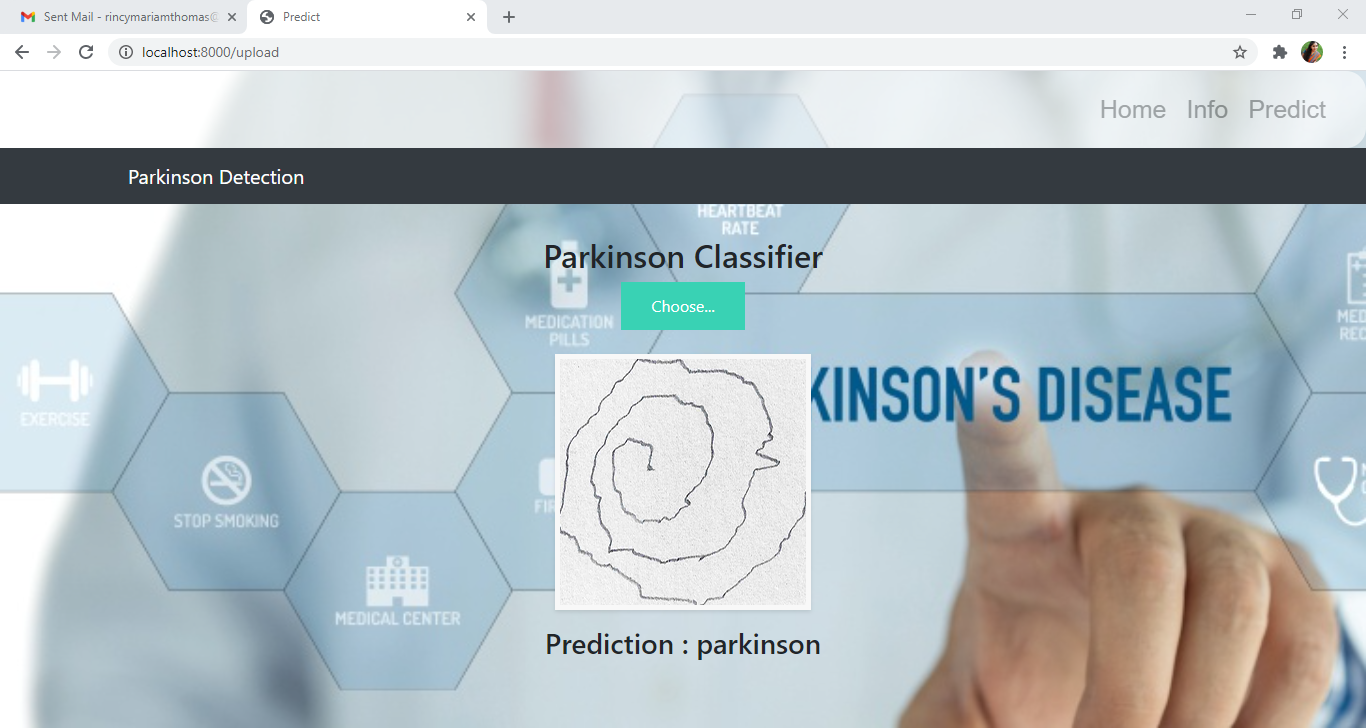
Click on choose and select the image and click on the **“Predict!”** button





Finally, the output is displayed on **predict.html**





**Train The Model On IBM**

In this milestone, you will learn how to build a Machine Learning  Model and deploy it on the IBM Cloud.

**Register For IBM Cloud**

**IBM Account:**

* Please click [here](https://www.ibm.com/academic/home) to register for IBM
* Please click [here](https://cloud.ibm.com/login)to log in to IBM Account

**Watch the below video to register and login into your IBM account**

<https://youtu.be/x6i43M7BAqE>

**Train Model On IBM Watson**

**Please watch the below video to train the model on IBM  and integrate it with the flask Application**

<https://youtu.be/BzouqMGJ41k>

**Ideation Phase**

In this milestone you are expected to get started with the Ideation process.

**Literature Survey On The Selected Project & Information Gathering**

In this activity you are expected to gather/collect the relevant information on project usecase, refer the existing solutions, technical papers, research publications etc.

**Prepare Empathy Map**

In this activity you are expected to prepare the empathy map canvas to capture the user Pains & Gains, Prepare list of problem statements.

**Ideation**

In this activity you are expected to list the ideas (at least 4 per each team member) by organizing the brainstorming session and prioritize the top 3 ideas based on the feasibility & importance.