Project Report Format

1. INTRODUCTION

1.1 **Project Overview :** The project focused on disease prediction using machine learning, employing K-Nearest Neighbors (KNN) and Random Forest Classifier (RFC) algorithms. It consisted of dataset acquisition, preprocessing, and feature selection. The effectiveness of KNN and RFC was proved through model training and evaluation, as well as insights into their strengths and drawbacks.The model predicts the the disease based on the given symptoms.we have achieved the accuracy of more than 90 percent for this model

1.2 **Purpose :** Disease prediction models are crucial for early intervention, enabling timely treatment and improved outcomes. They help preventative healthcare by identifying high-risk individuals and enabling for focused interventions and lifestyle changes. By focusing on high-risk groups, these models optimise healthcare resources, resulting in more efficient resource allocation. Early detection and intervention reduce the economic burden of advanced-stage illnesses, resulting in cost savings. Disease prediction helps with personalised medicine by adapting therapies to specific risk variables. Furthermore, aggregated data assists public health planning by directing targeted actions and policy development.

2. LITERATURE SURVEY

Existing problem : In today's face paced world, there is little to no time spared for healthcare. Even after developing severe symptoms to various diseases patients do not see a doctor. Using Google to type in our symptoms does not lead to good results, it always boils down to one stereotypical disease. Some people have stopped using Google to look for their symptoms or the probable disease. Late disease detection limits treatment options.Resource constraints in healthcare necessitate efficient allocation through predictive models. Rising healthcare costs .Predictive models address public health challenges by informing targeted campaigns and strategies.

2.1 **References :**

https://www.geeksforgeeks.org/disease-prediction-using-machine-learning/

https://github.com/anuj-glitch/Disease-Prediction-using-Django-and-machine- learning

https://ieeexplore.ieee.org/document/9823744

https://www.nature.com/articles/s41598-021-87171-5

2.2 **Problem Statement Definition :** People often delay seeking medical attention due to various reasons, such as lack of time, high costs, or unreliable online symptom search results.There is a need for a solution that provides accessible and reliable disease prediction to help individuals make informed decisions about their health. Rising healthcare costs due to advanced-stage treatments highlight the need for cost-effective early detection methods. Because healthcare interventions are not personalised, prediction models are needed to customise strategies to specific risk variables. Furthermore, public health planning struggles to address widespread diseases, highlighting the importance of data-driven techniques for focused interventions. To address these difficulties, a strong illness prediction model that enables early, personalised, and proactive healthcare measures is urgently needed.

3. IDEATION & PROPOSED SOLUTION

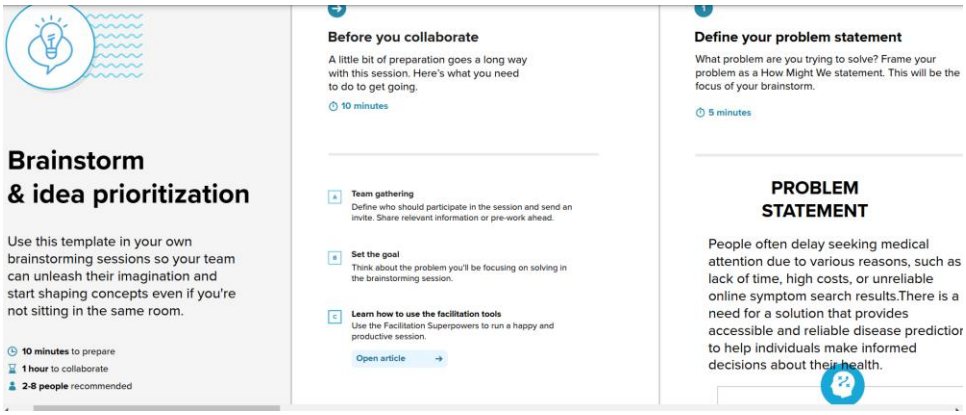
3.1. Empathy Map Canvas:

An empathy map is a simple, easy-to-digest visual that captures knowledge about a user's behavior and attitudes. It is a useful tool to helps teams better understand their users. Creating an effective solution requires understanding the true problem and the person who is experiencing it. The exercise of creating the map helps participants consider things from the user's perspective along with his or her goals and challenges.

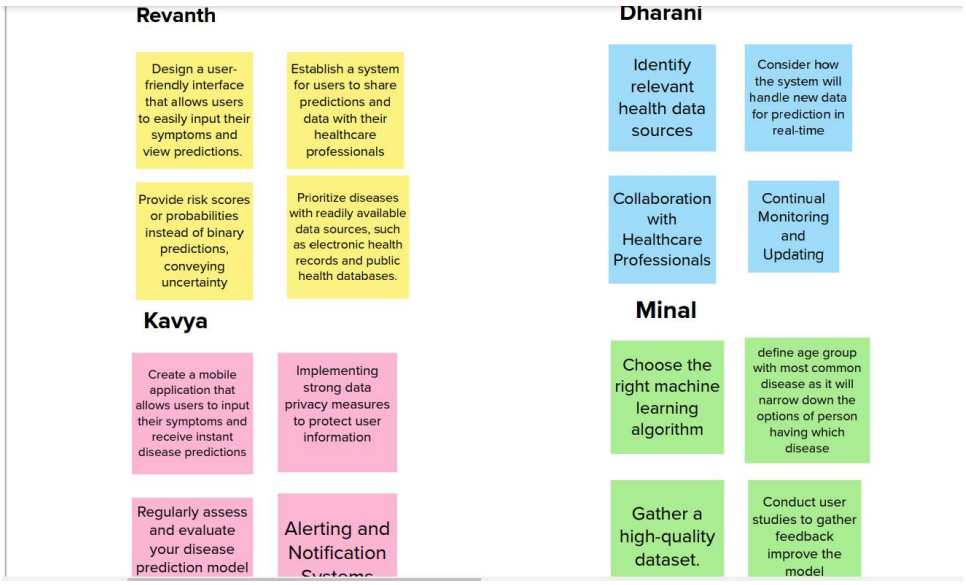
3.2 Ideation & Brainstorming

Brainstorming provides a free and open environment that encourages everyone within a team to participate in the creative thinking process that leads to problem solving. Prioritizing volume over value, out-of-the-box ideas are welcome and built upon, and all participants are encouraged to collaborate, helping each other develop a rich amount of creative solutions. Use this template in your own brainstorming sessions so your team can unleash their imagination and start shaping concepts even if you're not sitting in the same room.

Step-1: Team Gathering, Collaboration and Select the Problem Statement



Step-2: Brainstorm, Idea Listing and Grouping



Step-3: Idea Prioritization

Developing the Application for easy access

Design a user-friendly interface that allows users to easily input their symptoms and view predictions.

Establish a system for users to share predictions and data with their healthcare professionals

Create a mobile application that allows users to input their symptoms and receive instant disease predictions

Outreach, and User Feedback

Launch an awareness campaign to educate the public about the availability and benefits of the disease prediction system.

Regularly assess and evaluate your disease prediction model to ensure fair and unbiased.

Conduct user studies to gather feedback improve the model

Provide risk scores or probabilities instead of binary predictions, conveying uncertainty

Managing the website

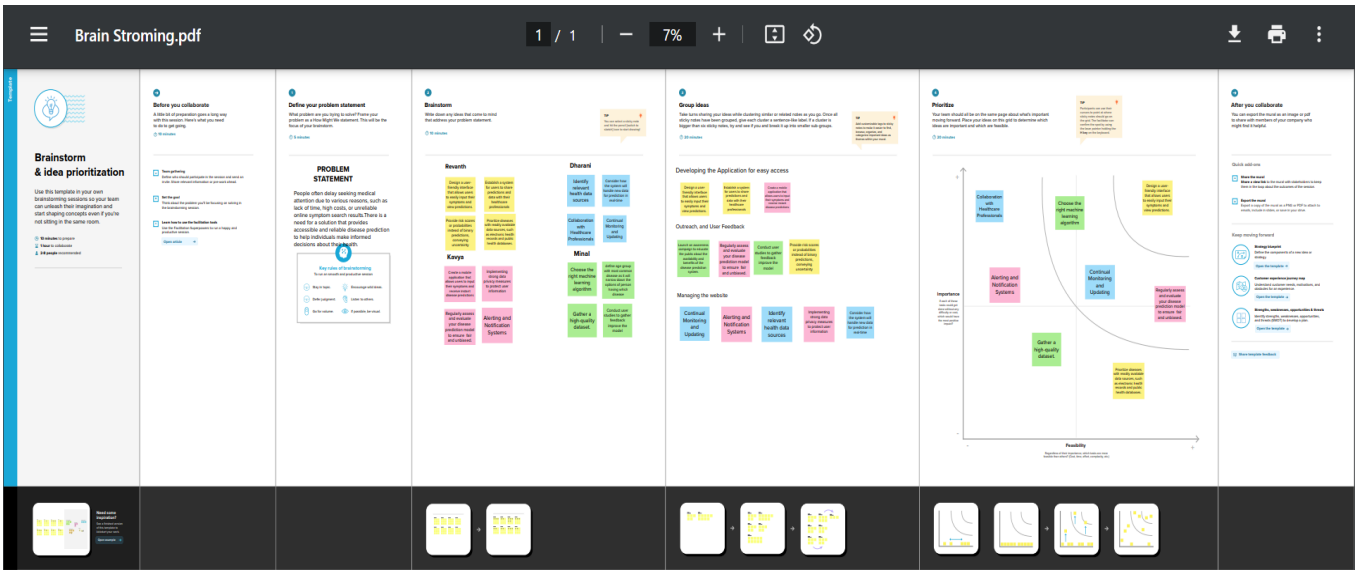
Continual Monitoring and Updating

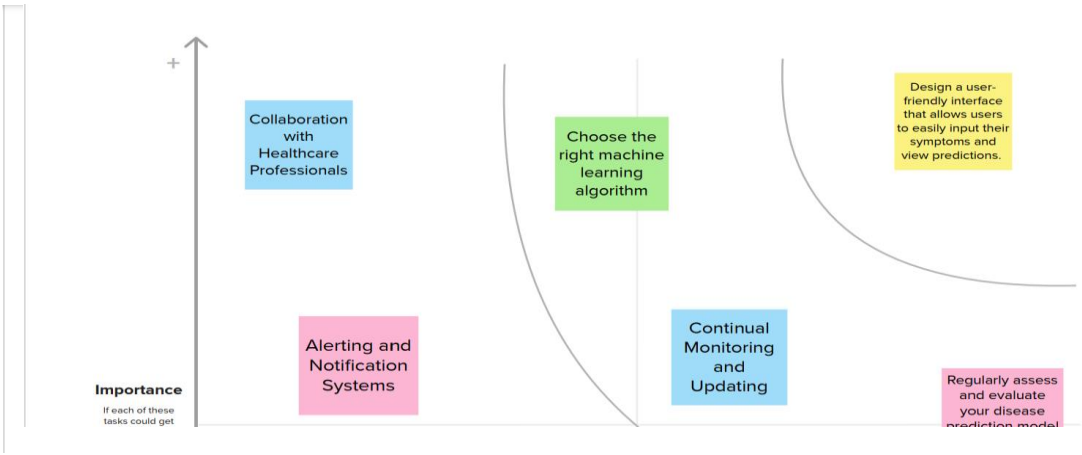
Alerting and Notification Systems

Identify relevant health data sources

Implementing strong data privacy measures to protect user information

Consider how the system will handle new data for prediction in real-time





4. REQUIREMENT ANALYSIS

4.1 Functional requirement

These functional requirements collectively contribute to the development of a reliable, effective, and user-friendly disease prediction model that addresses the unique needs of the healthcare domain.They involve specifying the features and capabilities of the system. User Authentication and Authorization ,Data Input,Data Preprocessing ,Feature Selection,Machine Learning Model,Training the Model,Model Evaluation,Real-time Prediction,Interpretability,Integration with Existing Systems,Notification System,Scalability,User Interface,Documentation and Support,Feedback Mechanism These are the main things that are required for building a good model

4.2 Non-Functional requirements

Non-functional requirements for a disease prediction system using machine learning are characteristics that describe the overall behavior and quality of the system.

Performance**:**The system should be capable of providing predictions within a specified response time, considering both training and prediction phases.

Scalability:The system should scale gracefully to accommodate an increasing number of users and a growing volume of data without a significant degradation in performance.

Reliability:The system should have a high level of reliability, ensuring that predictions are accurate and consistent over time.

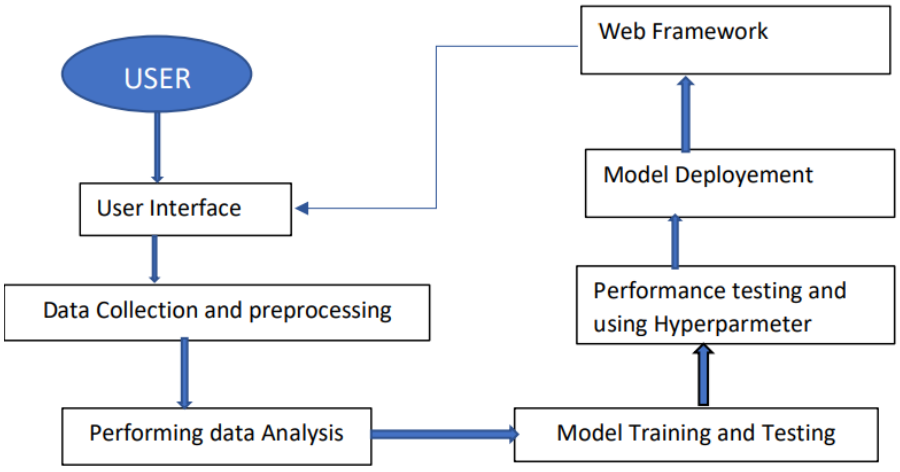
4.3 Privacy and security,cost,compatability,response time,ethucal considerations are some examples of Non-Functional requirements

5. PROJECT DESIGN

5.1 Data Flow Diagrams & User Stories

A Data Flow Diagram (DFD) is a visual representation of how data flows within a system. In the context of disease prediction using machine learning, the DFD can help illustrate

how data is collected, processed, and used to make predictions. Here's a high-level DFD for a disease prediction system



User stories

Below all the user stories for the product.

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance Criteria	Priority	Release
Patient	Disease Risk Assessment	1	As a patient, I want to provide my medical history and current symptoms to receive personalized disease risk assessments and recommendations for preventive measures.	- The system accepts patient input for medical history and symptoms. - The system calculates disease risk scores based on input. - The system provides recommendations for preventive measures.	High	1.0
Healthcare Provider	Diagnosis Support	2	As a healthcare provider, I want to input a patient's medical data to obtain disease predictions that can aid in diagnosis and treatment planning.	- The system accepts medical data input. - The system generates disease predictions. - The system offers explanations for prediction results.	High	1.0

Public Health Official	Disease Monitoring	4	As a public health official, I want to monitor disease trends and potential outbreaks using the system to facilitate timely interventions and resource allocation.	- The system provides real-time disease trend data. - The system sends alerts for potential outbreaks. - The system supports resource allocation recommendations.	Medium	1.1
Caregiver	Health Alerts	5	As a caregiver, I want to receive alerts or notifications if a family member's health data suggests an increased risk of a specific disease, enabling early intervention.	The system allows caregivers to set up health alerts. - The system sends notifications for increased disease risk. - The system offers guidance for early intervention.	Medium	1.2
Healthcare Institution	Integration	6	As a healthcare institution, I want to integrate the disease prediction system into our electronic health record (EHR) system to streamline the diagnostic process.	- The system integrates with the EHR system. - Patient data flows seamlessly between systems. - The diagnostic process is improved.	High	1.2
Patient	Health Tracking	7	As a patient, I want to view a user-friendly dashboard that presents my disease risk scores over time, helping me track my health status.	- The system provides a user-friendly dashboard. - Disease risk scores are displayed over time. - The dashboard supports health tracking.	Medium	1.3

5.2 Solution Architecture

Solution architecture is a complex process – with many sub-processes – that bridges the gap between business problems and technology solutions. Its goals are to:

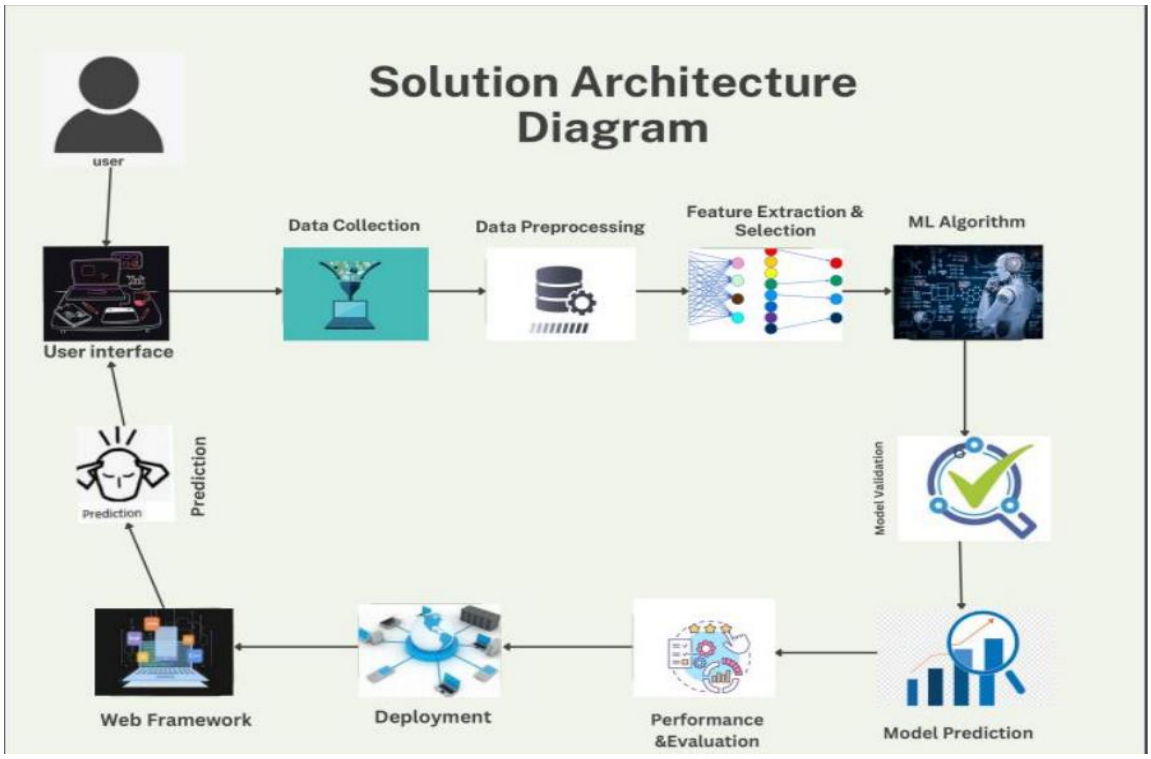
• Find the best tech solution to solve existing business problems.

• Describe the structure, characteristics, behavior, and other aspects of the software to project stakeholders.

• Define features, development phases, and solution requirements.

• Provide specifications according to which the solution is defined, managed, and

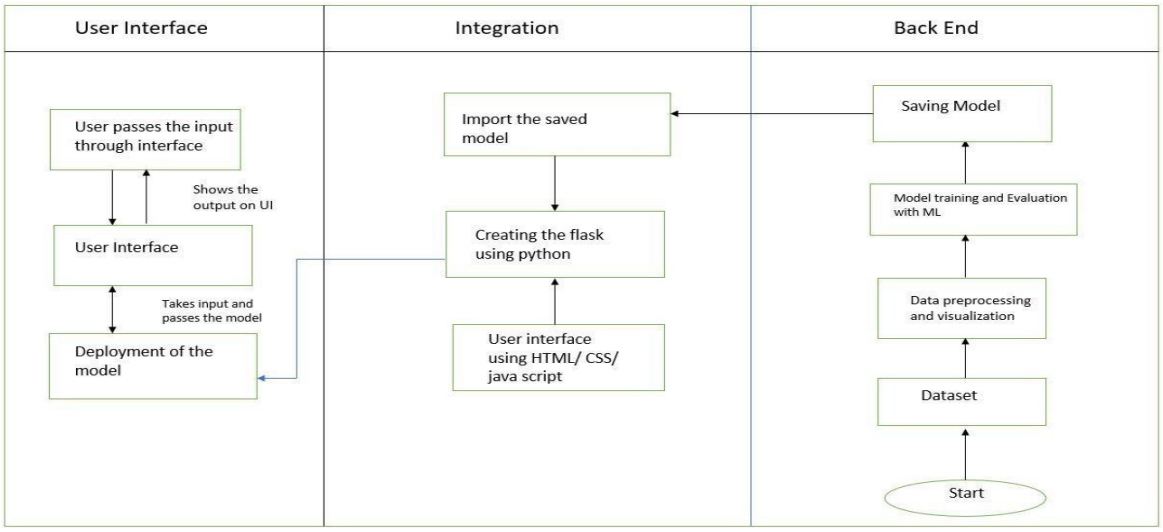
delivered.

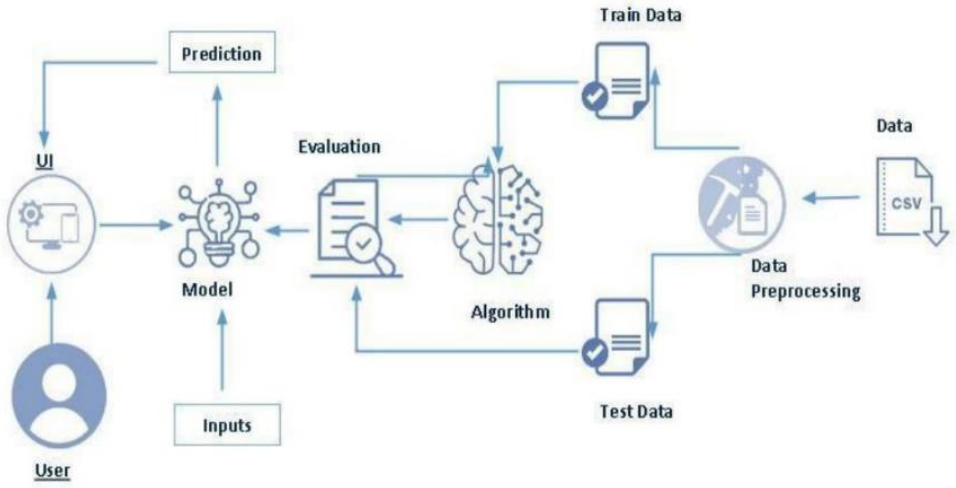


6. PROJECT PLANNING & SCHEDULING

6.1 Technical Architecture

Technical architecture refers to the structure and organization of a system's hardware and software components, as well as the way they interact to fulfill a set of requirements. It provides a blueprint for the design and implementation of a technology solution, outlining the arrangement and relationships among various elements to achieve specific goals.





6.2 Sprint Planning & Estimation

Sprint Planning and Estimation are key activities in Agile methodologies, particularly in Scrum, a popular Agile framework. These activities help teams organize their work, set priorities, and determine how much work can be accomplished in a specific time frame, known as a sprint.

Sprint Planning

Establish the goals and scope of the upcoming sprint.Decide which items from the product backlog will be included in the sprint backlog.

Estimation

Determine the effort required to complete each backlog item.Facilitate better planning and

resource allocation.

Project Planning Phase

Project Planning Template (Product Backlog, Sprint Planning, Stories, Story points)

Product Backlog, Sprint Schedule, and Estimation (4 Marks)

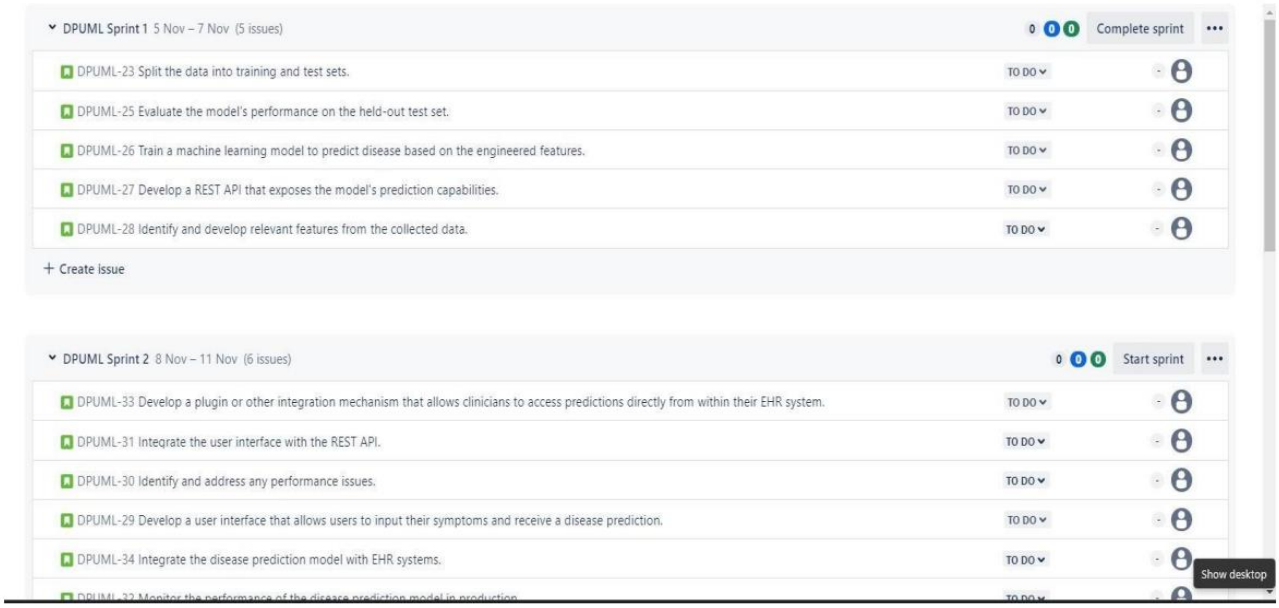
Sprint	Functional Requirement (Epic)	User Story Number	User Story	Acceptance Criteria	Story points	Priority	Team Members
Sprint 1	Data Collection	DSP-1	As a data scientist, I want to collect and prepare a dataset of medical records and symptoms for training the machine learning model, so that the model can be trained to predict disease accurately.	Collect medical records and symptoms data from a variety of sources, such as hospitals, clinics, and research studies.	3	medium	Revanth,Kavya
Sprint 1	Feature Engineering	DSP-2	Split the data into training and test sets.	The data should be split into training and test sets in a ratio of 80% to 20%.	5	High	Minal
Sprint 2	Model Training	DSP-3	Develop and engineer features from the collected data that are relevant to predicting disease.	The features should be engineered in a way that is suitable for training the machine learning model.	5	High	Dharani, Minal, kavya
Sprint 2	Model Evaluation	DSP-4	Train a machine learning model to predict disease based on the engineered features.	The model should be trained to achieve a high level of accuracy on the held-out test set.	4	High	Revanth, Dharani
Sprint 3	Model Deployment	DSP-5	Deploy the trained machine learning model to production.	The model should be deployed to a production environment so that it can be used to predict disease for new patients.	2	Medium	Revanth
Sprint 3	User Interface (UI) Development	DSP-6	Develop a user interface that allows users to input their symptoms and receive a disease prediction.	The user interface should be easy to use and navigate. It should also implement security measures to protect user data.	5	High	Kavya, Dharani
Sprint 4	Integration with Electronic Health Records (EHR) Systems	DSP-7	Integrate the disease prediction model with EHR systems.	The model should be integrated with EHR systems so that clinicians can easily access predictions for their patients.	4	High	Revanth, Minal
Sprint 4	Performance Monitoring and Improvement	DSP-8	Monitor the performance of the disease prediction model in production and make improvements as needed.	The model's performance should be monitored on real-world data. Any performance issues should be identified and addressed.	2	Medium	kavya
Sprint 4	Documentatio n and Training	DSP-9	Develop documentation and training materials on how to use the disease prediction model.	The documentation and training materials should be clear, concise, and easy to understand.	3	Medium	Dharani
Sprint 4	Security and Compliance	DSP-10	Implement security and compliance measures to protect the privacy and security of patient data.	The model should be used in a safe and ethical manner.	5	High	Minal,Kavya,Dharani

6.3 Sprint Delivery Schedule

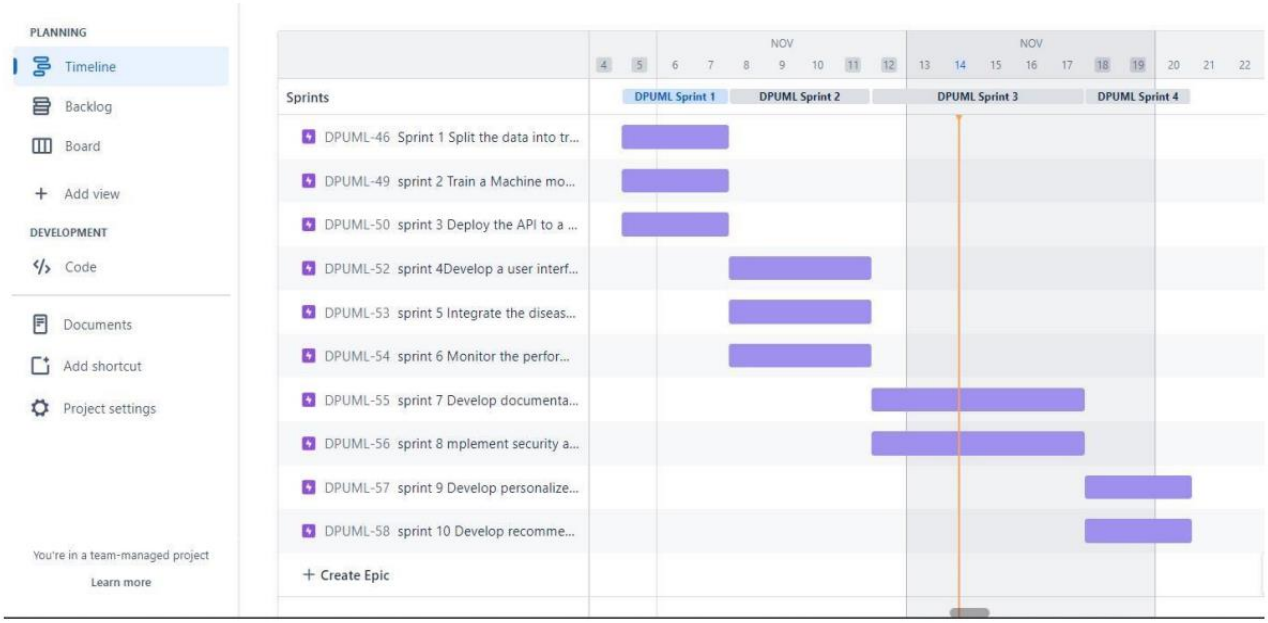
In Scrum, a sprint is a time-boxed iteration during which a potentially shippable product increment is created. Sprints are usually two to four weeks long, and they provide a consistent, short timeframe for development teams to deliver incremental value to the product.

Sprint	Total Story Points	Duration	Sprint Start Date	Sprint End Date (Planned)	Story Points Completed (as on Planned End Date)	Sprint Release Date(Actual)
Sprint-1	8	3 Days	5th Nov 2023	7th Nov 2023	8	5th Nov 2023
Sprint-2	9	4 Days	8th Nov 2023	11th Nov 2023		
Sprint-3	7	6 Days	12th Nov 2023	17th Nov 2023		
Sprint-4	10	3 Days	18h Nov 2023	20th Nov 2023		

Backlog Chart:



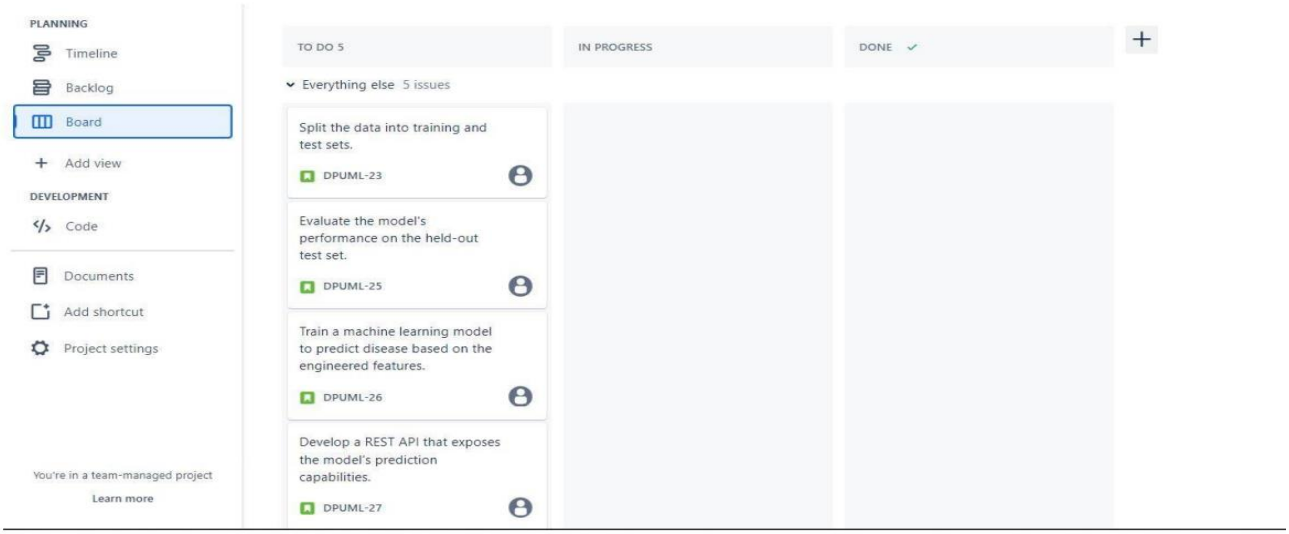
Time line chart:



Burndown Chart:

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

Board chart:



7. CODING & SOLUTIONING (Explain the features added in the project along with code)

- **Collect or Create Dataset:**
  - Acquire relevant data either by collecting from existing sources or creating a dataset that aligns with the objectives of the analysis or modeling.

- **Data Preprocessing:**
  - a. **Import Libraries:**
    - Import necessary programming libraries (e.g., in Python, using **import** statements) for data manipulation, analysis, and modeling.
  - b. **Importing the Dataset:**
    - Load the dataset into the chosen programming environment or tool (e.g., using Pandas in Python).
  - c. **Checking for Null Values:**
    - Examine the dataset for missing values to ensure data quality and decide on appropriate strategies for handling them.
  - d. **Data Visualization:**
    - Utilize visualization techniques (e.g., matplotlib or seaborn in Python) to gain insights into the dataset's distribution, patterns, and relationships between variables.
  - e. **Outlier Detection:**
    - Identify and handle outliers, data points significantly different from others, to prevent them from disproportionately influencing the model.
  - f. **Splitting Dependent and Independent Variables:**
    - Distinguish between the target variable (dependent) and the features (independent variables) to prepare for model training.
  - g. **Encoding:**

        - Convert categorical variables into a numerical format suitable for machine learning algorithms, often using techniques like one-hot encoding.
    - h. **Feature Scaling:**
        - Normalize or scale numerical features to ensure uniformity in their impact on the model.
    - i. **Splitting Data into Train and Test:**
        - Divide the dataset into training and testing sets to assess the model's performance on unseen data.
- **Model Building:**
    - a. **Import Model Building Libraries:**
        - Bring in libraries specific to machine learning algorithms, such as scikit-learn for Python.
    - b. **Initializing the Model:**
        - Create an instance of the chosen machine learning model, defining its structure and initial parameters.
    - c. **Training and Testing the Model:**
        - Use the training dataset to teach the model patterns and relationships, then assess its performance on the testing set to ensure generalization.
    - d. **Evaluation of Model:**
        - Assess the model's performance using appropriate metrics, considering accuracy, precision, recall, and other relevant measures based on the problem at hand.
    - e. **Save the Model:**
        - Preserve the trained model for future use or deployment, allowing for consistency and efficiency in application scenarios.

```
knn_table
```

	Number of features	Training Accuracy	Testing Accuracy
0	3	0.098069	0.095238
1	9	0.244411	0.238095
2	17	0.427846	0.428571
3	24	0.576220	0.571429
4	33	0.716717	0.714286
5	53	0.935976	0.952381
6	63	0.964685	0.952381

```
#Building Model with appropriate features
to_drop = []
for i,j in zip(feat_imp.keys(),feat_imp.values()):
    if j < 0.01:
        to_drop.append(i)
```

```
len(to_drop)
```

```
36
```

```
X_new.head()
```

	itching	joint_pain	vomiting	fatigue	w
0	1	0	0	0	
1	0	0	0	0	
2	1	0	0	0	
3	1	0	0	0	
4	1	0	0	0	

5 rows × 53 columns

8. PERFORMANCE TESTING

8.1 Performace Metrics

#CONFUSION MATRIX

```
[81] confusion_matrix(y1_test, y_pred1)

array([[[1, 0, 0, ..., 0, 0, 0],
        [0, 1, 0, ..., 0, 0, 0],
        [0, 1, 0, ..., 0, 0, 0],
        ...,
        [0, 0, 0, ..., 1, 0, 0],
        [0, 0, 0, ..., 0, 1, 0],
        [0, 0, 0, ..., 0, 0, 1]])
```

```
pd.crosstab(y1_test, y_pred1)
```

col_0	(vertigo) Paroymsal Positional Vertigo	AIDS	Arthritis	Bronchial Asthma	Cervical spondylosis	Chicken pox	Chronic cholestasis	Common Cold	Dengue	Diabetes	...	Osteoarthristis	h
prognosis													
(vertigo) Paroymsal Positional Vertigo	1	0	0	0	0	0	0	0	0	0	...	0	
AIDS	0	1	0	0	0	0	0	0	0	0	...	0	
Acne	0	1	0	0	0	0	0	0	0	0	...	0	
Alcoholic hepatitis	0	1	0	0	0	0	0	0	0	0	...	0	
Allergy	0	1	0	0	0	0	0	0	0	0	...	0	
Arthritis	0	0	1	0	0	0	0	0	0	0	...	0	
Bronchial Asthma	0	0	0	1	0	0	0	0	0	0	...	0	
Cervical													
GERD	0	0	0	0	0	0	0	0	0	0	...	0	
Gastroenteritis	0	0	0	0	0	0	0	0	0	0	...	0	
Heart attack	0	0	0	0	0	0	0	0	0	0	...	0	
Hepatitis B	0	0	0	0	0	0	0	0	0	0	...	0	
Hepatitis C	0	0	0	0	0	0	0	0	0	0	...	0	
Hepatitis D	0	0	0	0	0	0	0	0	0	0	...	0	
Hepatitis E	0	0	0	0	0	0	0	0	0	0	...	0	
Hypertension	0	0	0	0	0	0	0	0	0	0	...	0	
Hyperthyroidism	0	0	0	0	0	0	0	0	0	0	...	0	
Hypoglycemia	0	0	0	0	0	0	0	0	0	0	...	0	
Hypothyroidism	0	0	0	0	0	0	0	0	0	0	...	0	
Impetigo	0	0	0	0	0	0	0	0	0	0	...	0	
Jaundice	0	0	0	0	0	0	0	0	0	0	...	0	
Malaria	0	0	0	0	0	0	0	0	0	0	...	0	
Migraine	0	0	0	0	0	0	0	0	0	0	...	0	
Osteoarthristis	0	0	0	0	0	0	0	0	0	0	...	1	
Paralysis (brain hemorrhage)	0	0	0	0	0	0	0	0	0	0	...	0	1
Peptic ulcer diseae	0	0	0	0	0	0	0	0	0	0	...	0	
Pneumonia	0	0	0	0	0	0	0	0	0	0	...	0	
Psoriasis	0	0	0	0	0	0	0	0	0	0	...	0	

✓ Connected to Python 3 Google Compute Engine backend

Classification Matrix:

```
print(classification_report(y1_test,y_pred1))
```

```
                                   precision    recall  f1-score   support

    (vertigo) Paroymsal  Positional Vertigo       1.00      1.00      1.00         1
                                AIDS       0.20      1.00      0.33         1
                                Acne       0.00      0.00      0.00         1
                Alcoholic hepatitis       0.00      0.00      0.00         1
                             Allergy       0.00      0.00      0.00         1
                           Arthritis       1.00      1.00      1.00         1
                    Bronchial Asthma       1.00      1.00      1.00         1
                Cervical spondylosis       1.00      1.00      1.00         1
                           Chicken pox       1.00      1.00      1.00         1
                Chronic cholestasis       1.00      1.00      1.00         1
                           Common Cold       1.00      1.00      1.00         1
                                Dengue       1.00      1.00      1.00         1
                              Diabetes       1.00      1.00      1.00         1
        Dimorphic hemmorhoids(piles)       0.00      0.00      0.00         1
                          Drug Reaction       1.00      1.00      1.00         1
                       Fungal infection       1.00      1.00      1.00         2
                                   GERD       1.00      1.00      1.00         1
                         Gastroenteritis       1.00      1.00      1.00         1
                            Heart attack       1.00      1.00      1.00         1
                             Hepatitis B       1.00      1.00      1.00         1
                             Hepatitis C       0.50      1.00      0.67         1
                             Hepatitis D       0.00      0.00      0.00         1
                             Hepatitis E       1.00      1.00      1.00         1
                            Hypertension       1.00      1.00      1.00         1
                           Hyperthyroidism       1.00      1.00      1.00         1
                              Hypoglycemia       1.00      1.00      1.00         1
                           Hypothyroidism       1.00      1.00      1.00         1
                                Impetigo       1.00      1.00      1.00         1
                                Jaundice       1.00      1.00      1.00         1
                                 Malaria       1.00      1.00      1.00         1
                               Migraine       1.00      1.00      1.00         1
                    Osteoarthristis       1.00      1.00      1.00         1
          Paralysis (brain hemorrhage)       1.00      1.00      1.00         1
                   Peptic ulcer diseae       1.00      1.00      1.00         1
                             Pneumonia       1.00      1.00      1.00         1
                             Psoriasis       1.00      1.00      1.00         1
                           Tuberculosis       1.00      1.00      1.00         1
                                Typhoid       1.00      1.00      1.00         1
               Urinary tract infection       0.50      1.00      0.67         1
                        Varicose veins       1.00      1.00      1.00         1
                           hepatitis A       1.00      1.00      1.00         1

                              accuracy                           0.95        42
                             macro avg       0.93      0.95      0.93        42
                          weighted avg       0.93      0.95      0.94        42

/usr/local/lib/python3.10/dist-packages/sklearn/metrics/_classification.py:1344: UndefinedMetr
  _warn_prf(average, modifier, msg_start, len(result))
/usr/local/lib/python3.10/dist-packages/sklearn/metrics/_classification.py:1344: UndefinedMetr
  _warn_prf(average, modifier, msg_start, len(result))
/usr/local/lib/python3.10/dist-packages/sklearn/metrics/_classification.py:1344: UndefinedMetr
  _warn_prf(average, modifier, msg_start, len(result))
```

9. RESULTS

9.1 Output Screenshots

For Random Forest Classifier

```
y_pred = rfc_new.predict(X1_val)
yt_pred = rfc_new.predict(X1_train)
y_pred1 = rfc_new.predict(X1_test)
print('The Training Accuracy of the algorithm is ', accuracy_score(y1_train, yt_pred))
print('The Validation Accuracy of the algorithm is ', accuracy_score(y1_val, y_pred))
print('The Testing Accuracy of the algorithm is', accuracy_score(y1_test, y_pred1))
```

```
The Training Accuracy of the algorithm is  0.9402947154471545
The Validation Accuracy of the algorithm is  0.9278455284552846
The Testing Accuracy of the algorithm is 0.9523809523809523
```

```
randomf = pd.DataFrame(data = rfc_results,columns=['Number of features','Training Accuracy','Testing Accuracy'])
```

```
randomf
```

	Number of features	Training Accuracy	Testing Accuracy
0	3	0.099085	0.095238
1	9	0.248222	0.238095
2	17	0.435213	0.428571
3	24	0.579522	0.571429
4	33	0.718496	0.714286
5	53	0.936230	0.928571
6	63	0.965955	0.952381

For KNN Classifier

```
y_pred = knn_new.predict(X1_val)
yt_pred = knn_new.predict(X1_train)
y_pred1 = knn_new.predict(X1_test)
print('The Training Accuracy of the algorithm is ', accuracy_score(y1_train, yt_pred))
print('The Validation Accuracy of the algorithm is ', accuracy_score(y1_val, y_pred))
print('The Testing Accuracy of the algorithm is', accuracy_score(y1_test, y_pred1))
```

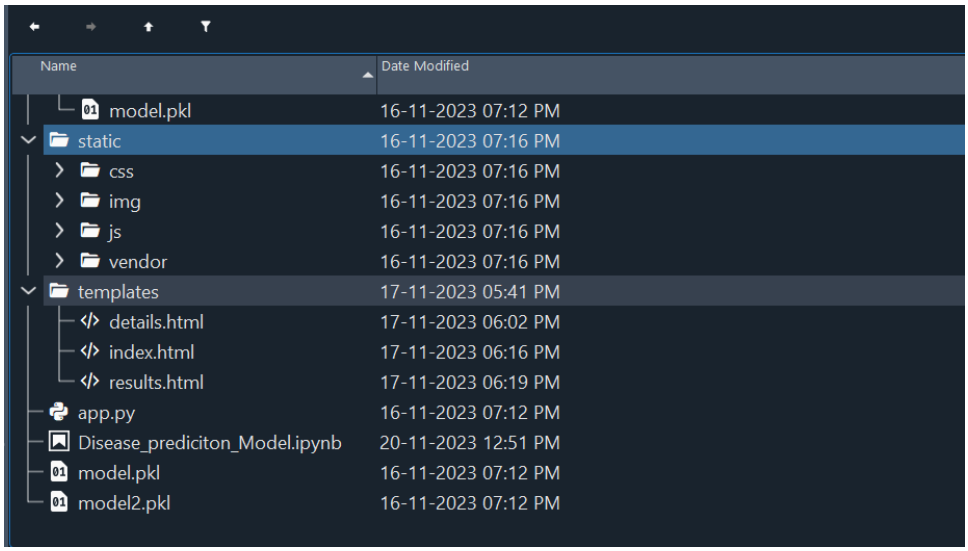
```
The Training Accuracy of the algorithm is  0.9367378048780488
The Validation Accuracy of the algorithm is  0.9359756097560976
The Testing Accuracy of the algorithm is 0.9523809523809523
```

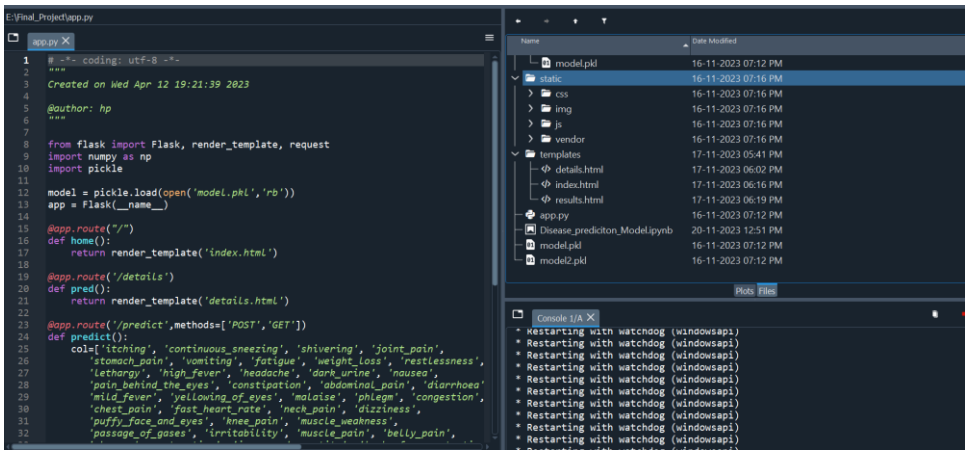
```
knn_table = pd.DataFrame(data = knn_results,columns=['Number of features','Training Accuracy','Testing Accuracy'])
```

```
knn_table
```

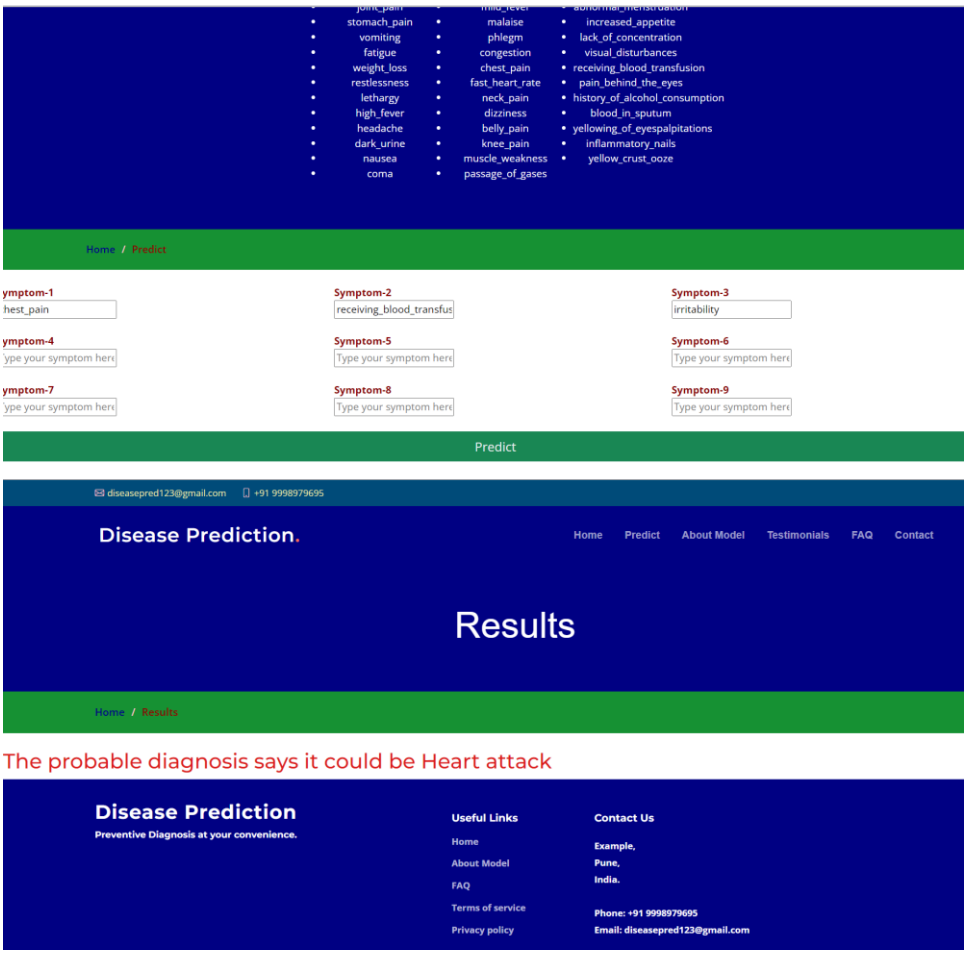
	Number of features	Training Accuracy	Testing Accuracy
0	3	0.098069	0.095238
1	9	0.244411	0.238095
2	17	0.427846	0.428571
3	24	0.576220	0.571429
4	33	0.716717	0.714286
5	53	0.935976	0.952381
6	63	0.964685	0.952381

Integrating with Flask gives


Name
Date Modified
model.pkl 16-11-2023 07:12 PM
static 16-11-2023 07:16 PM
css 16-11-2023 07:16 PM
img 16-11-2023 07:16 PM
js 16-11-2023 07:16 PM
vendor 16-11-2023 07:16 PM
templates 17-11-2023 05:41 PM
details.html 17-11-2023 06:02 PM
index.html 17-11-2023 06:16 PM
results.html 17-11-2023 06:19 PM
app.py 16-11-2023 07:12 PM
Disease_prediciton_Model.ipynb 20-11-2023 12:51 PM
model.pkl 16-11-2023 07:12 PM
model2.pkl 16-11-2023 07:12 PM


E:\Final_Project\app.py
app.py
-*- coding: utf-8 -*-
"""
Created on Wed Apr 12 19:21:39 2023
@author: hp
"""
from flask import Flask, render_template, request
import numpy as np
import pickle
model = pickle.load(open('model.pkl','rb'))
app = Flask(__name__)
@app.route("/")
def home():
return render_template('index.html')
@app.route('/details')
def pred():
return render_template('details.html')
@app.route('/predict',methods=['POST','GET'])
def predict():
col=['itching', 'continuous_sneezing', 'shivering', 'joint_pain',
'stomach_pain', 'vomiting', 'fatigue', 'weight_loss', 'restlessness',
'lethargy', 'high_fever', 'headache', 'dark_urine', 'nausea',
'pain_behind_the_eyes', 'constipation', 'abdominal_pain', 'diarrhoea'
'mild_fever', 'yellowing_of_eyes', 'malaise', 'phlegm', 'congestion',
'chest_pain', 'fast_heart_rate', 'neck_pain', 'dizziness',
'puffy_face_and_eyes', 'knee_pain', 'muscle_weakness',
'passage_of_gases', 'irritability', 'muscle_pain', 'belly_pain',
Name
Date Modified
model.pkl 16-11-2023 07:12 PM
static 16-11-2023 07:16 PM
css 16-11-2023 07:16 PM
img 16-11-2023 07:16 PM
js 16-11-2023 07:16 PM
vendor 16-11-2023 07:16 PM
templates 17-11-2023 05:41 PM
details.html 17-11-2023 06:02 PM
index.html 17-11-2023 06:16 PM
results.html 17-11-2023 06:19 PM
app.py 16-11-2023 07:12 PM
Disease_prediciton_Model.ipynb 20-11-2023 12:51 PM
model.pkl 16-11-2023 07:12 PM
model2.pkl 16-11-2023 07:12 PM
Plots Files
Console 1/A
* Restarting with watchdog (windowsapi)
* Restarting with watchdog (windowsapi)
* Restarting with watchdog (windowsapi)
* Restarting with watchdog (windowsapi)
* Restarting with watchdog (windowsapi)
* Restarting with watchdog (windowsapi)
* Restarting with watchdog (windowsapi)
* Restarting with watchdog (windowsapi)
* Restarting with watchdog (windowsapi)


127.0.0.1:8000
diseasepred123@gmail.com
+91 9998979695
Disease Prediction.
Home
Predict
About Model
Testimonials
FAQ
Contact
Welcome to Disease Prediction Using Machine Learning
This webpage will help you predict the disease you might be having using the symptoms given as input.
Predict
Watch Video



10. ADVANTAGES & DISADVANTAGES

While disease prediction using machine learning has significant advantages, it is critical to consider the underlying problems in order to design responsible and successful healthcare solutions. The successful adoption of ML models in the healthcare domain requires addressing concerns such as privacy, bias, and interpretability.

Advantages

Detection at an Early Stage: ML models provide for early disease detection, allowing for prompt interventions and improved treatment outcomes.

Personalised Healthcare:Based on individual health profiles, ML algorithms can deliver personalised risk assessments, personalising treatments and preventive measures.

Resource Allocation that is Efficient: Predictive models optimise healthcare resources by focusing on high-risk groups, resulting in more effective medical intervention allocation.

Healthcare at a Low Cost: Early identification with machine learning models can result in cost savings by lowering the economic burden associated with treating advanced-stage diseases.

Insights Based on Data:Data is used by machine learning models to discover patterns, trends, and risk factors, providing useful insights to healthcare practitioners and public health planners.

Continuous Inspection: ML-based disease prediction allows for continuous monitoring of health conditions, enabling proactive and ongoing management of individual and population health.

Disadvantages

Data Privacy Concerns: ML models often require access to sensitive health data, raising concerns about privacy and the secure handling of patient information.

Overfitting and Generalization Issues: ML models may face challenges of overfitting to training data or difficulty in generalizing predictions to diverse populations.

Ethical Considerations: The use of predictive models raises ethical concerns, including issues related to consent, transparency, and the potential for unintended consequences in healthcare decision-making.

11. CONCLUSION :

disease prediction using machine learning represents a transformative approach with substantial benefits for healthcare. The capacity to diagnose diseases early, personalise interventions based on individual risk factors, and optimise resource allocation highlights the technology's potential to transform patient care. To ensure ethical and equitable healthcare solutions, concerns such as data privacy, algorithm bias, and model interpretability must be addressed. Despite these obstacles, the constant growth of machine learning models, together with advances in data quality and regulatory frameworks, holds out hope for a future in which predictive analytics is an intrinsic component of proactive and personalised healthcare. As we traverse these issues, the ethical application of illness prediction models remains critical in realising technology's full promise for improving global health outcomes.

12. FUTURE SCOPE

Machine learning has a promising and dynamic future in disease prediction. Integration with wearable devices, multi-modal data fusion, and improvements in precision medicine will improve the accuracy and personalization of disease forecasts. Explainable artificial intelligence (XAI) will be critical in providing transparency and developing confidence in healthcare decision-making. Machine learning models have the potential to make a significant contribution to population health management by recognising trends and enabling preemptive public health actions. For comprehensive and flexible illness prediction research, continuous model learning, wider disease spectrum coverage, and global cooperation are expected. To eliminate bias, protect privacy standards, and increase public trust, ethical considerations and responsible AI deployment will remain focal points. Experiential learning, telemedicine integration, and regulatory framework advancements will further impact the future landscape of disease prediction using machine learning.

13. APPENDIX

Source Code

Introduction

Disease prediction utilizes predictive analytics and machine learning techniques to identify individuals

at risk of developing a specific illness. This involves analyzing extensive datasets containing

medical history and demographic factors to recognize patterns and risk factors associated with different diseases.

Machine learning algorithms are employed to process this data, aiding in the identification of individuals predisposed

to a particular ailment. This proactive approach enables healthcare professionals to offer personalized preventive care

and early intervention, ultimately enhancing the effectiveness of disease management strategies.

Steps to be followed

Data Collection.

o Collect the dataset or Create the dataset

• Data Preprocessing. o Import the Libraries.

o Importing the dataset.

o Checking for Null Values.

o Data Visualization.

o Outlier Detection

o Splitting Dependent and Independent variables

o- Encoding

o Feature Scaling.

o Splitting Data into Train and Test.

• Model Building

o Import the model building Libraries

o Initializing the model

o Training and testing the model

o Evaluation of Model

o Save the Model

#Importing Libraries

```
import numpy as np
import pandas as pd

import matplotlib.pyplot as plt
import seaborn as sns

from sklearn.model_selection import train_test_split
from sklearn.metrics import accuracy_score

from sklearn.neighbors import KNeighborsClassifier
from sklearn.svm import SVC
from sklearn.tree import DecisionTreeClassifier
from sklearn.ensemble import RandomForestClassifier

import pickle
```

#Importing the dataset

```
train = pd.read_csv("E:/final p/Training.csv")
test = pd.read_csv("E:/final p/Testing.csv")
train.head()
train.shape
```

#Removing Redundant Columns

```
train['Unnamed: 133'].value_counts()
train.drop("Unnamed: 133",axis = 1, inplace = True)
```

#checking for null values if any

```
train.isnull().sum()
train.isnull().sum().sum()
### here we dont have any null values so no need to handle anything we can skip this part
train.columns
#Descriptive analysis--- used to study the basic features of data with the statistical process.
train.describe()
test.head()
test.shape
```

```
len(train.prognosis.unique())
train.prognosis.value_counts()
for i in train.columns:
   print(train[i].value_counts())
train['fluid_overload'].unique()
train.drop('fluid_overload',axis = 1, inplace = True)
#multivariate analysis-comparing more features at a time
corr = train.corr()
corr.style.background_gradient('coolwarm')
#we drop the some columns due to high correlation
train.drop(['weight_gain','cold_hands_and_feets','anxiety','irregular_sugar_level',
         'yellow_urine','acute_liver_failure','swelling_of_stomach',
         'drying_and_tingling_lips','continuous_feel_of_urine',
        'internal_itching','polyuria','mood_swings','receiving_unsterile_injections',
        'stomach_bleeding','prominent_veins_on_calf','loss_of_smell','throat_irritation',
        'redness_of_eyes','sinus_pressure','runny_nose','pain_during_bowel_movements',
        'pain_in_anal_region','cramps','bruising','enlarged_thyroid','brittle_nails',
        'swollen_extremeties','slurred_speech','distention_of_abdomen','fluid_overload.1',
        'skin_peeling','silver_like_dusting','small_dents_in_nails','blister',
        'red_sore_around_nose','bloody_stool','swollen_blood_vessels','hip_joint_pain',
        'painful_walking','spinning_movements','altered_sensorium','toxic_look_(typhos)'],axis =1,
inplace = True)
```

#new correlation matrix

```
corr = train.corr()
corr.style.background_gradient('coolwarm')
```

#data Visualization

```
#Univariate Analysis -- understanding the data with a single feature.
#creating a side-by-side comparison of pie charts for distribution of 'itching' and
'continuous_sneezing' symptoms
plt.figure(figsize = (8,8))

a = train['itching'].value_counts()
plt.subplot(121)
```

```
plt.pie(x = a, data = train, labels= ['No','Yes'], autopct='%.0f%%',colors = 'gr')
plt.title("Pie chart showing the distribution of Itching symptom into number of Yes/No ")

b = train['continuous_sneezing'].value_counts()
plt.subplot(122)
plt.pie(x = b, data = train, labels= ['No','Yes'], autopct='%.0f%%',colors = 'gr')
plt.title('Pie Chart showing the distribution of Continuous Sneezing symptom into number of Yes/No')

plt.subplots_adjust(left = 0.5, right = 2.4)
#creating a side-by-side comparison of pie charts for distribution of 'shivering' and 'joint pain ' symptoms
plt.figure(figsize = (8,8))

a = train['shivering'].value_counts()
plt.subplot(121)
plt.pie(x = a, data = train, labels= ['No','Yes'], autopct='%.0f%%',colors = 'gr')
plt.title("Pie chart showing the distribution of Shivering symptom into number of Yes/No ")

b = train['joint_pain'].value_counts()
plt.subplot(122)
plt.pie(x = b, data = train, labels= ['No','Yes'], autopct='%.0f%%',colors = 'gr')
plt.title('Pie Chart showing the distribution of Joint Pain symptom into number of Yes/No')

plt.subplots_adjust(left = 0.5, right = 2.4)
#creating a side-by-side comparison of bar graphs for below distribution
plt.subplot(1,2,1)
train['stomach_pain'].value_counts().plot(kind = 'bar', color = ['g','r'])
plt.title("Bar chart showing the distribution of Stomach Pain symptom")

plt.subplot(1,2,2)
train['vomiting'].value_counts().plot(kind = 'bar', color = ['g','r'])
plt.title("Bar chart showing the distribution of Vomiting symptom")
```

```
plt.subplots_adjust(left = 0.5, right = 2.5)
#creating a side-by-side comparison of bar graphs for below distribution
plt.subplot(1,2,1)
train['fatigue'].value_counts().plot(kind = 'bar', color = ['g','r'])
plt.title("Bar chart showing the distribution of Fatigue symptom")

plt.subplot(1,2,2)
train['weight_loss'].value_counts().plot(kind = 'bar', color = ['g','r'])
plt.title("Bar chart showing the distribution of Weight Loss symptom")

plt.subplots_adjust(left = 0.5, right = 2.5)
#creating a side-by-side comparison of bar graphs horizontally for below distribution
plt.subplot(1,2,1)
train['restlessness'].value_counts().plot(kind = 'barh', color = ['g','r'])
plt.title("Bar chart showing the distribution of Restlessness symptom")

plt.subplot(1,2,2)
train['lethargy'].value_counts().plot(kind = 'barh', color = ['g','r'])
plt.title("Bar chart showing the distribution of Lethargy symptom")

plt.subplots_adjust(left = 0.5, right = 2.5)
#creating a side-by-side comparison of bar graphs horizontally for below distribution
plt.subplot(1,2,1)
train['high_fever'].value_counts().plot(kind = 'barh', color = ['g','r'])
plt.title("Bar chart showing the distribution of High Fever symptom")

plt.subplot(1,2,2)
train['headache'].value_counts().plot(kind = 'barh', color = ['g','r'])
plt.title("Bar chart showing the distribution of Headache symptom")

plt.subplots_adjust(left = 0.5, right = 2.5)
```

```
train[(train['itching'] == 1) & (train['stomach_pain'] == 1)]
```

Bivariate Analysis:

To find the relation between two features

```
a = len(train[train['prognosis'] == 'Fungal infection'])
b = len(train[(train['itching'] == 1) & (train['prognosis'] == 'Fungal infection')])
fi = pd.DataFrame(data = [a,b], columns=['Values'],index = ['Fungal Infection','Itching while Fungal Infection'])

sns.barplot(data = fi, x = fi.index, y = fi['Values'])
plt.title('Importance of Itching symptom to determine Fungal Infection')
a = len(train[train['prognosis'] == 'Jaundice'])
b = len(train[(train['high_fever'] == 1) & (train['prognosis'] == 'Jaundice')])
fi = pd.DataFrame(data = [a,b], columns=['Values'],index = ['Jaundice','High Fever while Jaundice'])

sns.barplot(data = fi, x = fi.index, y = fi['Values'])
plt.title('Importance of High Fever symptom to determine Jaundice')

a = len(train[train['prognosis'] == 'Tuberculosis'])
b = len(train[(train['yellowing_of_eyes'] == 1) & (train['prognosis'] == 'Tuberculosis')])
fi = pd.DataFrame(data = [a,b], columns=['Values'],index = ['Tuberculosis','Yellowing of Eyes while Tuberculosis'])

sns.barplot(data = fi, x = fi.index, y = fi['Values'])
plt.title('Importance of Yellowing of Eyes symptom to determine Tuberculosis')

a = len(train[train['prognosis'] == 'Alcoholic hepatitis'])
b = len(train[(train['history_of_alcohol_consumption'] == 1) & (train['prognosis'] == 'Alcoholic hepatitis')])
fi = pd.DataFrame(data = [a,b], columns=['Values'],index = ['Alcoholic hepatitis','History while Alcoholic hepatitis'])

sns.barplot(data = fi, x = fi.index, y = fi['Values'])
plt.title('Importance of History of Consumption symptom to determine Alcoholic hepatitis')
```

```
a = train[train['prognosis'] == 'Tuberculosis']
a.head()
sns.swarmplot(x = a['weight_loss'], y = a['fatigue'])
train.head()
#We can create a function for test data preprocessing which will only leave us with the required features.
def data_preprocessing(data):

data.drop(['fluid_overload','weight_gain','cold_hands_and_feets','anxiety','irregular_sugar_level',
        'yellow_urine','acute_liver_failure','swelling_of_stomach',
        'drying_and_tingling_lips','continuous_feel_of_urine',
       'internal_itching','polyuria','mood_swings','receiving_unsterile_injections',
       'stomach_bleeding','prominent_veins_on_calf','loss_of_smell','throat_irritation',
       'redness_of_eyes','sinus_pressure','runny_nose','pain_during_bowel_movements',
       'pain_in_anal_region','cramps','bruising','enlarged_thyroid','brittle_nails',
       'swollen_extremeties','slurred_speech','distention_of_abdomen','fluid_overload.1',
       'skin_peeling','silver_like_dusting','small_dents_in_nails','blister',
       'red_sore_around_nose','bloody_stool','swollen_blood_vessels','hip_joint_pain',
       'painful_walking','spinning_movements','altered_sensorium','toxic_look_(typhos)'],axis =1, inplace = True)
   return data
#We split the training data into features(X) and target variable(y).
X = train.drop('prognosis',axis = 1)
y = train.prognosis
#Split training data into training(80%), validation data(20%)
X_train, X_val, y_train, y_val = train_test_split(X,y,test_size = 0.2)
test = data_preprocessing(test)
#Here we split the test data into features(X_test) and the corresponding target variables(y_test)
X_test = test.drop('prognosis',axis = 1)
y_test = test.prognosis
def model_evaluation(classifier):
   y_pred = classifier.predict(X_val)
   yt_pred = classifier.predict(X_train)
```

```
    y_pred1 = classifier.predict(X_test)
    print('The Training Accuracy of the algorithm is ', accuracy_score(y_train, yt_pred))
    print('The Validation Accuracy of the algorithm is ', accuracy_score(y_val, y_pred))
    print('The Testing Accuracy of the algorithm is', accuracy_score(y_test, y_pred1))
    return [(accuracy_score(y_train, yt_pred)), (accuracy_score(y_val, y_pred)),
(accuracy_score(y_test, y_pred1))]
```

Training and testing the models using multiple algorithms steps for model building

KNN

SVM

Decision

Tree Random Forest

#Steps for model building

o Import the model building Libraries

o Initializing the model

o Training and testing the model

o Evaluation of Model

o Save the Model

```
knn = KNeighborsClassifier(n_neighbors=7)
knn.fit(X_train,y_train)
knn_results = model_evaluation(knn)
svm = SVC(C=1)
svm.fit(X_train, y_train)
svm_results = model_evaluation(svm)
dtc = DecisionTreeClassifier(max_features= 10)
dtc.fit(X_train,y_train)
dtc_results = model_evaluation(dtc)
rfc = RandomForestClassifier(max_depth = 13)
rfc.fit(X_train, y_train)
rfc_results = model_evaluation(rfc)
#Testing model with Multiple Evaluatiton metrics
results = pd.DataFrame(data = [knn_results, svm_results, dtc_results, rfc_results],
```

```
                columns= ['Training Accuracy','Validation Accuracy', 'Testing Accuracy'],
                index = ['K Nearest Neighbors Classifier','Support Vector Machines',
                        'Decision Trees Classifier', 'Random Forest Classifier'])
results
#We can check the feature importance using the Random Forest Classifier model.
a = rfc.feature_importances_
col = X.columns
feat_imp = {}
for i, j in zip(a,col):
    feat_imp[j] = i
feat_imp
def model_evaluation1(n_feat,classifier):
    y_pred = classifier.predict(X1_val)
    yt_pred = classifier.predict(X1_train)
    y_pred1 = classifier.predict(X1_test)
    return [(n_feat),(accuracy_score(y1_train, yt_pred)), (accuracy_score(y1_test, y_pred1))]
rfc_results = []
knn_results = []
#We will drop columns which have very less feature importance.
#we have created a for loop which will train the model and give out the accuracy.
for main in [0.020,0.018,0.016,0.014,0.012,0.01,0.008]:
    to_drop = []
    for i,j in zip(feat_imp.keys(),feat_imp.values()):
        if j < main:
            to_drop.append(i)

    X_new = X.drop(to_drop,axis = 1)
    y_new = y
    X1_train, X1_val, y1_train, y1_val = train_test_split(X_new, y_new, test_size=0.2)
    X1_test = X_test.drop(to_drop,axis = 1)
    y1_test = y_test
    rfc_new = RandomForestClassifier()
```

```
    rfc_new.fit(X1_train, y1_train)
    temp1 = model_evaluation1(X1_train.shape[1], rfc_new)
    rfc_results.append(temp1)
    knn_new = KNeighborsClassifier()
    knn_new.fit(X1_train, y1_train)
    temp2 = model_evaluation1(X1_train.shape[1],knn_new)
    knn_results.append(temp2)
randomf = pd.DataFrame(data = rfc_results,columns=['Number of features','Training Accuracy','Testing Accuracy'])
randomf
knn_table = pd.DataFrame(data = knn_results,columns=['Number of features','Training Accuracy','Testing Accuracy'])
knn_table
#Building Model with appropriate features
to_drop = []
for i,j in zip(feat_imp.keys(),feat_imp.values()):
    if j < 0.01:
        to_drop.append(i)
len(to_drop)
X_new = X.drop(to_drop,axis = 1)
y_new = y
X_new.head()
X1_train, X1_val, y1_train, y1_val = train_test_split(X_new, y_new, test_size=0.2)
X1_test = X_test.drop(to_drop,axis = 1)
y1_test = y_test
rfc_new = RandomForestClassifier()
rfc_new.fit(X1_train, y1_train)
y_pred = rfc_new.predict(X1_val)
yt_pred = rfc_new.predict(X1_train)
y_pred1 = rfc_new.predict(X1_test)
print('The Training Accuracy of the algorithm is ', accuracy_score(y1_train, yt_pred))
print('The Validation Accuracy of the algorithm is ', accuracy_score(y1_val, y_pred))
print('The Testing Accuracy of the algorithm is', accuracy_score(y1_test, y_pred1))
```

```
knn_new = KNeighborsClassifier()
knn_new.fit(X1_train, y1_train)
y_pred = knn_new.predict(X1_val)
yt_pred = knn_new.predict(X1_train)
y_pred1 = knn_new.predict(X1_test)
print('The Training Accuracy of the algorithm is ', accuracy_score(y1_train, yt_pred))
print('The Validation Accuracy of the algorithm is ', accuracy_score(y1_val, y_pred))
print('The Testing Accuracy of the algorithm is', accuracy_score(y1_test, y_pred1))
#comparing our predicted results with the actual values.
test.join(pd.DataFrame(y_pred1,columns=["predicted"]))[["prognosis","predicted"]]
X1_train.sum(axis = 1).max()
X1_train.columns
#Model Deployment
pickle.dump(knn_new, open('model.pkl','wb'))
```

Index.html

```
<!DOCTYPE html>
<html lang="en">

<head>
  <meta charset="utf-8">
  <meta content="width=device-width, initial-scale=1.0" name="viewport">

  <title>Disease Prediction</title>
  <meta content="" name="description">
  <meta content="" name="keywords">

  <!-- Favicons -->
  <link href="https://static.vecteezy.com/system/resources/previews/000/355/283/non_2x/vector-health-icon.jpg" rel="icon">
  <link href="https://static.vecteezy.com/system/resources/previews/000/355/283/non_2x/vector-health-icon.jpg" rel="apple-touch-icon">
```

```
  <!-- Google Fonts -->
  <link rel="preconnect" href="https://fonts.googleapis.com">
  <link rel="preconnect" href="https://fonts.gstatic.com" crossorigin>
  <link
href="https://fonts.googleapis.com/css2?family=Open+Sans:ital,wght@0,300;0,400;0,500;0,600;
0,700;1,300;1,400;1,600;1,700&family=Montserrat:ital,wght@0,300;0,400;0,500;0,600;0,700;1,
300;1,400;1,500;1,600;1,700&family=Raleway:ital,wght@0,300;0,400;0,500;0,600;0,700;1,300;
1,400;1,500;1,600;1,700&display=swap" rel="stylesheet">

  <!-- Vendor CSS Files -->
  <link href="static/vendor/bootstrap/css/bootstrap.min.css" rel="stylesheet">
  <link href="static/vendor/bootstrap-icons/bootstrap-icons.css" rel="stylesheet">
  <link href="static/vendor/aos/aos.css" rel="stylesheet">
  <link href="static/vendor/glightbox/css/glightbox.min.css" rel="stylesheet">
  <link href="static/vendor/swiper/swiper-bundle.min.css" rel="stylesheet">

  <!-- Template Main CSS File -->
  <link href="static/css/main.css" rel="stylesheet">

  <!-- =======================================================
  * Template Name: Impact
  * Updated: Mar 10 2023 with Bootstrap v5.2.3
  * Template URL: https://bootstrapmade.com/impact-bootstrap-business-website-template/
  * Author: BootstrapMade.com
  * License: https://bootstrapmade.com/license/
  ======================================================= -->
</head>

<body>

  <!-- ======= Header ======= -->
  <section id="topbar" class="topbar d-flex align-items-center">
```

```
    <div class="container d-flex justify-content-center justify-content-md-between">
      <div class="contact-info d-flex align-items-center">
        <i class="bi bi-envelope d-flex align-items-center"><a
href="mailto:contact@example.com">diseasepred123@gmail.com</a></i>
        <i class="bi bi-phone d-flex align-items-center ms-4"><span>+91 9998979695</span></i>
      </div>
    </div>
  </section><!-- End Top Bar -->

  <header id="header" class="header d-flex align-items-center">

    <div class="container-fluid container-xl d-flex align-items-center justify-content-between">
      <a href="index.html" class="logo d-flex align-items-center">
        <!-- Uncomment the line below if you also wish to use an image logo -->
        <img src="static/img/logo.png" alt="">
        <h1>Disease Prediction<span>.</span></h1>
      </a>
      <nav id="navbar" class="navbar">
        <ul>
          <li><a href="#hero">Home</a></li>
          <li><a href = "/details">Predict</a></li>
          <li><a href="#about">About Model</a></li>
          <li><a href="#testimonials">Testimonials</a></li>
          <li><a href="#faq">FAQ</a></li>
          <li><a href="#contact">Contact</a></li>
        </ul>
      </nav><!-- .navbar -->

      <i class="mobile-nav-toggle mobile-nav-show bi bi-list"></i>
      <i class="mobile-nav-toggle mobile-nav-hide d-none bi bi-x"></i>

    </div>
```

```
</header><!-- End Header -->
<!-- End Header -->

<!-- ======= Hero Section ======= -->
<section id="hero" class="hero">
  <div class="container position-relative">
    <div class="row gy-5" data-aos="fade-in">
      <div class="col-lg-6 order-2 order-lg-1 d-flex flex-column justify-content-center text-center text-lg-start">
        <h2>Welcome to <span>Disease Prediction</span> <span>Using Machine Learning</span></h2>
        <p>This webpage will help you predict the disease you might be having using the symptoms given as input.</p>
        <div class="d-flex justify-content-center justify-content-lg-start">
          <a href="/details" class="btn-get-started">Predict</a>
          <a href="https://www.youtube.com/watch?v=JOArz7wggkQ&ab_channel=EyeonTech" class="glightbox btn-watch-video d-flex align-items-center"><i class="bi bi-play-circle"></i><span>Watch Video</span></a>
        </div>
      </div>
      <div class="col-lg-6 order-1 order-lg-2">
        <img src="https://etimg.etb2bimg.com/photo/77373028.cms" class="img-fluid" alt="" data-aos="zoom-out" data-aos-delay="100">
      </div>
    </div>
  </div>

  </div>
</section>
<!-- End Hero Section -->

<main id="main">

  <!-- ======= About Us Section ======= -->
```

```
<section id="about" class="about">
  <div class="container" data-aos="fade-up">

    <div class="section-header">
      <h2>About Model</h2>
      <p>This model is developed using Machine Learning which will predict the disease you likely have.</p>
      <p>The output of this model can be useful to carry preventive health checkups for the disease predicted by model.</p>
    </div>

    <div class="row gy-4">
      <div class="col-lg-6">
        <img src="https://med.stanford.edu/news/all-news/2015/06/precision-health-predicting-and-preventing-disease/_jcr_content/main/image.img.620.high.jpg/precision-health-stock.jpg" class="img-fluid rounded-4 mb-4" alt="">
      </div>
      <div class="col-lg-6">
        <div class="content ps-0 ps-lg-5">
          <p class="fst-italic">
            The model is accurate in predicting the right disease 97 out of 100 times.
            The model gives a brief about disease for which tests need to be carried out for the symptoms given as input by user.
            The model is made with right amounts of the following:
          </p>
          <ul>
            <li><i class="bi bi-check-circle-fill"></i> Appropriate data in large amounts.</li>
            <li><i class="bi bi-check-circle-fill"></i> Statistical and Mathematical Techniques.</li>
            <li><i class="bi bi-check-circle-fill"></i> Machine Learning Techniques.</li>
          </ul>
          <p>
            This model is perfect for Self Diagnosis or Primary Diagnosis before visiting a doctor.
          </p>
```

```
          </div>
        </div>
      </div>

    </div>
  </section><!-- End About Us Section -->

  <!-- ======= Testimonials Section ======= -->
  <section id="testimonials" class="testimonials">
    <div class="container" data-aos="fade-up">

      <div class="section-header">
        <h2>Testimonials</h2>
        <p>Some testimonials about the experience of using this model.</p>
      </div>

      <div class="slides-3 swiper" data-aos="fade-up" data-aos-delay="100">
        <div class="swiper-wrapper">

          <div class="swiper-slide">
            <div class="testimonial-wrap">
              <div class="testimonial-item">
                <div class="d-flex align-items-center">
                  <img src="static/img/testimonials/testimonials-1.jpg" class="testimonial-img flex-shrink-0" alt="">
                  <div>
                    <h3>Minal</h3>
                    <h4>Journalist</h4>
                    <div class="stars">
                      <i class="bi bi-star-fill"></i><i class="bi bi-star-fill"></i><i class="bi bi-star-fill"></i><i class="bi bi-star-fill"></i><i class="bi bi-star-fill"></i>
                    </div>
                  </div>
```

```
</div>
<p>
<i class="bi bi-quote quote-icon-left"></i>
```
This is a great invention which is helpful for doctors as well as general patients round the world to check if there is a need to visit the doctor.
```
<i class="bi bi-quote quote-icon-right"></i>
</p>
</div>
<div class="testimonial-wrap">
<div class="testimonial-item">
<div class="d-flex align-items-center">
<img src="static/img/testimonials/testimonials-2.jpg" class="testimonial-img flex-shrink-0" alt="">
<div>
<h3>Bhavya</h3>
<h4>Designer</h4>
<div class="stars">
<i class="bi bi-star-fill"></i><i class="bi bi-star-fill"></i><i class="bi bi-star-fill"></i><i class="bi bi-star-fill"></i><i class="bi bi-star-fill"></i>
</div>
</div>
</div>
<p>
<i class="bi bi-quote quote-icon-left"></i>
```
My doctor recommended me to use this model once when he was not sure of his availability. Since then I've been using this model whenever I feel a bit low.
```
<i class="bi bi-quote quote-icon-right"></i>
</p>
</div>
</div>
</div>
<div class="testimonial-wrap">
<div class="testimonial-item">
```

<img src="static/img/testimonials/testimonials-3.jpg" class="testimonial-img flex-shrink-0" alt="">

<h3>Sruthi</h3>

<h4>Doctor</h4>

This model is really useful for doctors. I recommend all my patients to use this model before deciding they should visit me or no. I can shift my focus on more critical patients now.

<img src="static/img/testimonials/testimonials-4.jpg" class="testimonial-img flex-shrink-0" alt="">

<h3>Revanth</h3>

<h4>Architect</h4>

```
        </div>
        <p>
          <i class="bi bi-quote quote-icon-left"></i>
          Patients come to me to check if they have certain disease which was predicted by this model. Majority of the times the are diagnosed with that particular disease but at an early stage.
          <i class="bi bi-quote quote-icon-right"></i>
        </p>
      </div>
    </div>
    <div class="testimonial-wrap">
      <div class="testimonial-item">
        <div class="d-flex align-items-center">
          <img src="static/img/testimonials/testimonials-5.jpg" class="testimonial-img flex-shrink-0" alt="">
          <div>
            <h3>Dharani</h3>
            <h4>Receptionist</h4>
            <div class="stars">
              <i class="bi bi-star-fill"></i><i class="bi bi-star-fill"></i><i class="bi bi-star-fill"></i><i class="bi bi-star-fill"></i><i class="bi bi-star-fill"></i>
            </div>
          </div>
        </div>
        <p>
          <i class="bi bi-quote quote-icon-left"></i>
          Ever since people are using this model, the rush in hospitals has decreased significantly. The patients come in after doing their primary diagnosis, which makes work easier for the docctors.
          <i class="bi bi-quote quote-icon-right"></i>
        </p>
      </div>
    </div><!-- End testimonial item -->
```

```
        </div>
        <div class="swiper-pagination"></div>
       </div>

     </div>
    </section><!-- End Testimonials Section -->


    <!-- ======= Frequently Asked Questions Section ======= -->
    <section id="faq" class="faq">
     <div class="container" data-aos="fade-up">

      <div class="row gy-4">

       <div class="col-lg-4">
        <div class="content px-xl-5">
         <h3><strong>Frequently Asked Questions</strong></h3>
         <p>
          There are some questions that are very common while using the Disease Prediction Model which can come to anybody's mind. Lets us try and answer in detail regarding your questions.
         </p>
        </div>
       </div>

       <div class="col-lg-8">

        <div class="accordion accordion-flush" id="faqlist" data-aos="fade-up" data-aos-delay="100">

         <div class="accordion-item">
          <h3 class="accordion-header">
```

<button class="accordion-button collapsed" type="button" data-bs-toggle="collapse" data-bs-target="#faq-content-1">

1.

What level of trust is shown by doctors for this Disease Prediction model?

</button>

</h3>

This model does not try to replace doctors. It assists doctors and patients to get primary diagnosis or self diagnosis. Doctors trust this model at the primary diagnosis and suggest further test for proper diagnosis.

<!-- # Faq item-->

<h3 class="accordion-header">

<button class="accordion-button collapsed" type="button" data-bs-toggle="collapse" data-bs-target="#faq-content-2">

2.

Are there any timings as to when we can use this model?

</button>

</h3>

No. This model can be used anytime for primary diagnosis. This model works 24/7 throughout the year.

<!-- # Faq item-->

<h3 class="accordion-header">

<button class="accordion-button collapsed" type="button" data-bs-toggle="collapse" data-bs-target="#faq-content-3">

3.

Will this model suggest tests for full diagnosis?

</button>

</h3>

This model does not suggest any test. This model only predicts the disease which is likely for the symptoms you have given as input. You can use the information for preventive diagnosis or can consult a doctor regarding the same.

<!-- # Faq item-->

<h3 class="accordion-header">

<button class="accordion-button collapsed" type="button" data-bs-toggle="collapse" data-bs-target="#faq-content-4">

4.

How many dieases can this model predict based on symptoms?

</button>

</h3>

This model is capable of predicting 41 diseases. The model will output the disease which is most likely based on the symptoms.

<!-- # Faq item-->

```
          </div>

        </div>
      </div>

    </div>
  </section><!-- End Frequently Asked Questions Section -->

  <!-- ======= Contact Section ======= -->
  <section id="contact" class="contact">
    <div class="container" data-aos="fade-up">

      <div class="section-header">
        <h2>Contact</h2>
        <p>In case of any queries, please feel free to type in a message below or email us on diseasepred123@gmail.com </p>
        <p>Also check the FAQ section for your question.</p>
      </div>

      <div class="row gx-lg-0 gy-4">

        <div class="col-lg-4">

          <div class="info-container d-flex flex-column align-items-center justify-content-center">
            <div class="info-item d-flex">
              <i class="bi bi-geo-alt flex-shrink-0"></i>
              <div>
                <h4>Location:</h4>
                <p> Vijayawada,Andhra Pradesh</p>
              </div>
            </div><!-- End Info Item -->
```

```
    <div class="info-item d-flex">
      <i class="bi bi-envelope flex-shrink-0"></i>
      <div>
        <h4>Email:</h4>
        <p>diseasepred123@gmail.com</p>
      </div>
    </div><!-- End Info Item -->

    <div class="info-item d-flex">
      <i class="bi bi-phone flex-shrink-0"></i>
      <div>
        <h4>Call:</h4>
        <p>+91 9998979695</p>
      </div>
    </div><!-- End Info Item -->

    <div class="info-item d-flex">
      <i class="bi bi-clock flex-shrink-0"></i>
      <div>
        <h4>Open Hours:</h4>
        <p>Mon-Sat: 9AM - 5PM</p>
      </div>
    </div><!-- End Info Item -->
  </div>

</div>

<div class="col-lg-8">
  <form action="forms/contact.php" method="post" role="form" class="php-email-form">
    <div class="row">
      <div class="col-md-6 form-group">
        <input type="text" name="name" class="form-control" id="name"
```

```
placeholder="Your Name" required>
              </div>
              <div class="col-md-6 form-group mt-3 mt-md-0">
                <input type="email" class="form-control" name="email" id="email"
placeholder="Your Email" required>
              </div>
            </div>
            <div class="form-group mt-3">
              <input type="text" class="form-control" name="subject" id="subject"
placeholder="Subject" required>
            </div>
            <div class="form-group mt-3">
              <textarea class="form-control" name="message" rows="7" placeholder="Message"
required></textarea>
            </div>
            <div class="my-3">
              <div class="loading">Loading</div>
              <div class="error-message"></div>
              <div class="sent-message">Your message has been sent. Thank you!</div>
            </div>
            <div class="text-center"><button type="submit">Send Message</button></div>
          </form>
        </div><!-- End Contact Form -->

      </div>

    </div>
  </section><!-- End Contact Section -->

</main><!-- End #main -->

<!-- ======= Footer ======= -->
<footer id="footer" class="footer">
```

```
<div class="container">
  <div class="row gy-4">
    <div class="col-lg-5 col-md-12 footer-info">
      <a href="index.html" class="logo d-flex align-items-center">
        <span>Disease Prediction</span>
      </a>
      <p>Preventive Diagnosis at your convenience.</p>
    </div>

    <div class="col-lg-2 col-6 footer-links">
      <h4>Useful Links</h4>
      <ul>
        <li><a href="#">Home</a></li>
        <li><a href="#">About Model</a></li>
        <li><a href="#">FAQ</a></li>
        <li><a href="#">Terms of service</a></li>
        <li><a href="#">Privacy policy</a></li>
      </ul>
    </div>

    <div class="col-lg-3 col-md-12 footer-contact text-center text-md-start">
      <h4>Contact Us</h4>
      <p>
        Example, <br>
        Vijayawada,<br>
        Andhra Pradesh,India. <br><br>
        <strong>Phone:</strong> +91 9998979695<br>
        <strong>Email:</strong> diseasepred123@gmail.com<br>
      </p>

    </div>
```

```
    </div>
   </div>

   <div class="container mt-4">
    <div class="copyright">
     &copy; Copyright <strong><span>Disease Prediction</span></strong>. All Rights Reserved
    </div>
    <div class="credits">
     <!-- All the links in the footer should remain intact. -->
     <!-- You can delete the links only if you purchased the pro version. -->
     <!-- Licensing information: https://bootstrapmade.com/license/ -->
     <!-- Purchase the pro version with working PHP/AJAX contact form: https://bootstrapmade.com/impact-bootstrap-business-website-template/ -->
    </div>
   </div>

  </footer><!-- End Footer -->
  <!-- End Footer -->

  <a href="#" class="scroll-top d-flex align-items-center justify-content-center"><i class="bi bi-arrow-up-short"></i></a>

  <div id="preloader"></div>

  <!-- Vendor JS Files -->
  <script src="static/vendor/bootstrap/js/bootstrap.bundle.min.js"></script>
  <script src="static/vendor/aos/aos.js"></script>
  <script src="static/vendor/glightbox/js/glightbox.min.js"></script>
  <script src="static/vendor/purecounter/purecounter_vanilla.js"></script>
  <script src="static/vendor/swiper/swiper-bundle.min.js"></script>
  <script src="static/vendor/isotope-layout/isotope.pkgd.min.js"></script>
```

```
  <script src="static/vendor/php-email-form/validate.js"></script>

  <!-- Template Main JS File -->
  <script src="static/js/main.js"></script>

</body>

</html>
```

Results.html

```
<!DOCTYPE html>
<html lang="en">

<head>
  <meta charset="utf-8">
  <meta content="width=device-width, initial-scale=1.0" name="viewport">

  <title>Results</title>
  <meta content="" name="description">
  <meta content="" name="keywords">

  <!-- Favicons -->
  <link
href="https://static.vecteezy.com/system/resources/previews/000/355/283/non_2x/vector-
health-icon.jpg" rel="icon">
  <link
href="https://static.vecteezy.com/system/resources/previews/000/355/283/non_2x/vector-
health-icon.jpg" rel="apple-touch-icon">

  <!-- Google Fonts -->
  <link rel="preconnect" href="https://fonts.googleapis.com">
  <link rel="preconnect" href="https://fonts.gstatic.com" crossorigin>
  <link
href="https://fonts.googleapis.com/css2?family=Open+Sans:ital,wght@0,300;0,400;0,500;0
```

```
,600;0,700;1,300;1,400;1,600;1,700&family=Montserrat:ital,wght@0,300;0,400;0,500;0,60
0;0,700;1,300;1,400;1,500;1,600;1,700&family=Raleway:ital,wght@0,300;0,400;0,500;0,6
00;0,700;1,300;1,400;1,500;1,600;1,700&display=swap" rel="stylesheet">

  <!-- Vendor CSS Files -->
  <link href="static/vendor/bootstrap/css/bootstrap.min.css" rel="stylesheet">
  <link href="static/vendor/bootstrap-icons/bootstrap-icons.css" rel="stylesheet">
  <link href="static/vendor/aos/aos.css" rel="stylesheet">
  <link href="static/vendor/glightbox/css/glightbox.min.css" rel="stylesheet">
  <link href="static/vendor/swiper/swiper-bundle.min.css" rel="stylesheet">

  <!-- Template Main CSS File -->
  <link href="static/css/main.css" rel="stylesheet">

  <!-- =======================================================
  * Template Name: Impact
  * Updated: Mar 10 2023 with Bootstrap v5.2.3
  * Template URL: https://bootstrapmade.com/impact-bootstrap-business-website-template/
  * Author: BootstrapMade.com
  * License: https://bootstrapmade.com/license/
  ======================================================== -->
</head>

<body>

  <!-- ======= Header ======= -->
  <section id="topbar" class="topbar d-flex align-items-center">
    <div class="container d-flex justify-content-center justify-content-md-between">
      <div class="contact-info d-flex align-items-center">
        <i class="bi bi-envelope d-flex align-items-center"><a
href="mailto:contact@example.com">diseasepred123@gmail.com</a></i>
        <i class="bi bi-phone d-flex align-items-center ms-4"><span>+91
9998979695</span></i>
```

```
    </div>
   </div>
  </section><!-- End Top Bar -->

  <header id="header" class="header d-flex align-items-center">

    <div class="container-fluid container-xl d-flex align-items-center justify-content-between">
      <a href="index.html" class="logo d-flex align-items-center">
        <!-- Uncomment the line below if you also wish to use an image logo -->
        <img src="static/img/logo.png" alt="">
        <h1>Disease Prediction<span>.</span></h1>
      </a>
      <nav id="navbar" class="navbar">
        <ul>
          <li><a href="/">Home</a></li>
          <li><a href = "/details">Predict</a></li>
          <li><a href="/">About Model</a></li>
          <li><a href="/">Testimonials</a></li>
          <li><a href="/">FAQ</a></li>
          <li><a href="/">Contact</a></li>
        </ul>
      </nav><!-- .navbar -->

      <i class="mobile-nav-toggle mobile-nav-show bi bi-list"></i>
      <i class="mobile-nav-toggle mobile-nav-hide d-none bi bi-x"></i>

    </div>
  </header><!-- End Header -->
  <!-- End Header -->

  <main id="main">
```

```
  <!-- ======= Breadcrumbs ======= -->
  <div class="breadcrumbs">
    <div class="page-header d-flex align-items-center" style="background-image: url('');">
      <div class="container position-relative">
        <div class="row d-flex justify-content-center">
          <div class="col-lg-6 text-center">
            <h2>Results</h2>
          </div>
        </div>
      </div>
    </div>
    <nav>
      <div class="container">
        <ol>
          <li><a href="index.html">Home</a></li>
          <li>Results</li>
        </ol>
      </div>
    </nav>
  </div><!-- End Breadcrumbs -->

  <div class="content">
    <br>
                <b>  <h2 style="color:rgb(221, 28, 28)"> {{ prediction_text }}<b></h2>

    </div>

  </main><!-- End #main -->

<!-- ======= Footer ======= -->
<footer id="footer" class="footer">
```

```
<div class="container">
  <div class="row gy-4">
    <div class="col-lg-5 col-md-12 footer-info">
      <a href="index.html" class="logo d-flex align-items-center">
        <span>Disease Prediction</span>
      </a>
      <p>Preventive Diagnosis at your convenience.</p>
    </div>

    <div class="col-lg-2 col-6 footer-links">
      <h4>Useful Links</h4>
      <ul>
        <li><a href="#">Home</a></li>
        <li><a href="#">About Model</a></li>
        <li><a href="#">FAQ</a></li>
        <li><a href="#">Terms of service</a></li>
        <li><a href="#">Privacy policy</a></li>
      </ul>
    </div>

    <div class="col-lg-3 col-md-12 footer-contact text-center text-md-start">
      <h4>Contact Us</h4>
      <p>
        Example, <br>
        Pune,<br>
        India. <br><br>
        <strong>Phone:</strong> +91 9998979695 <br>
        <strong>Email:</strong> diseasepred123@gmail.com<br>
      </p>

    </div>
```

```
    </div>
  </div>

  <div class="container mt-4">
    <div class="copyright">
      &copy; Copyright <strong><span>Disease Prediction</span></strong>. All Rights Reserved
    </div>
    <div class="credits">
      <!-- All the links in the footer should remain intact. -->
      <!-- You can delete the links only if you purchased the pro version. -->
      <!-- Licensing information: https://bootstrapmade.com/license/ -->
      <!-- Purchase the pro version with working PHP/AJAX contact form: https://bootstrapmade.com/impact-bootstrap-business-website-template/ -->
    </div>
  </div>

</footer><!-- End Footer -->
<!-- End Footer -->

  <a href="#" class="scroll-top d-flex align-items-center justify-content-center"><i class="bi bi-arrow-up-short"></i></a>

  <div id="preloader"></div>

  <!-- Vendor JS Files -->
  <script src="static/vendor/bootstrap/js/bootstrap.bundle.min.js"></script>
  <script src="static/vendor/aos/aos.js"></script>
  <script src="static/vendor/glightbox/js/glightbox.min.js"></script>
  <script src="static/vendor/purecounter/purecounter_vanilla.js"></script>
  <script src="static/vendor/swiper/swiper-bundle.min.js"></script>
  <script src="static/vendor/isotope-layout/isotope.pkgd.min.js"></script>
```

```
    <script src="static/vendor/php-email-form/validate.js"></script>

    <!-- Template Main JS File -->
    <script src="static/js/main.js"></script>

  </body>

  </html>
```

Details.html

```
<!DOCTYPE html>
<html lang="en">

<head>
  <meta charset="utf-8">
  <meta content="width=device-width, initial-scale=1.0" name="viewport">

  <title>Details</title>
  <style>
    ul {
      columns: 3;
      -webkit-columns: 3;
      -moz-columns: 3;
    }
  </style>
  <meta content="" name="description">
  <meta content="" name="keywords">

  <!-- Favicons -->
  <link href="https://static.vecteezy.com/system/resources/previews/000/355/283/non_2x/vector-health-icon.jpg" rel="icon">
  <link href="https://static.vecteezy.com/system/resources/previews/000/355/283/non_2x/vector-health-icon.jpg" rel="apple-touch-icon">

  <!-- Google Fonts -->
  <link rel="preconnect" href="https://fonts.googleapis.com">
  <link rel="preconnect" href="https://fonts.gstatic.com" crossorigin>
  <link
href="https://fonts.googleapis.com/css2?family=Open+Sans:ital,wght@0,300;0,400;0,500;0,600;0,700;1,300;1,400;1,600;1,700&family=Montserrat:ital,wght@0,300;0,400;0,500;0,600;0,700;1,300;1,400;1,500;1,600;1,700&family=Raleway:ital,wght@0,300;0,400;0,500;0,600;0,700;1,300;1,400;1,500;1,600;1,700&display=swap" rel="stylesheet">
```

```
  <!-- Vendor CSS Files -->
  <link href="static/vendor/bootstrap/css/bootstrap.min.css" rel="stylesheet">
  <link href="static/vendor/bootstrap-icons/bootstrap-icons.css" rel="stylesheet">
  <link href="static/vendor/aos/aos.css" rel="stylesheet">
  <link href="static/vendor/glightbox/css/glightbox.min.css" rel="stylesheet">
  <link href="static/vendor/swiper/swiper-bundle.min.css" rel="stylesheet">

  <!-- Template Main CSS File -->
  <link href="static/css/main.css" rel="stylesheet">

  <!-- =======================================================
  * Template Name: Impact
  * Updated: Mar 10 2023 with Bootstrap v5.2.3
  * Template URL: https://bootstrapmade.com/impact-bootstrap-business-website-template/
  * Author: BootstrapMade.com
  * License: https://bootstrapmade.com/license/
  ======================================================== -->
</head>

<body>

  <!-- ======= Header ======= -->
  <section id="topbar" class="topbar d-flex align-items-center">
    <div class="container d-flex justify-content-center justify-content-md-between">
      <div class="contact-info d-flex align-items-center">
        <i class="bi bi-envelope d-flex align-items-center"><a
href="mailto:contact@example.com">diseasepred123@gmail.com</a></i>
        <i class="bi bi-phone d-flex align-items-center ms-4"><span>+91 9998979695</span></i>
      </div>
    </div>
  </section><!-- End Top Bar -->

  <header id="header" class="header d-flex align-items-center">

    <div class="container-fluid container-xl d-flex align-items-center justify-content-between">
      <a href="index.html" class="logo d-flex align-items-center">
        <h1>Disease Prediction<span>.</span></h1>
      </a>
      <nav id="navbar" class="navbar">
        <ul style="color: white">
          <li><a href="/">Home</a></li>
          <li><a href = "/details">Predict</a></li>
          <li><a href="/">About Model</a></li>
          <li><a href="/">Testimonials</a></li>
          <li><a href="/">FAQ</a></li>
```

```
        <li><a href="/">Contact</a></li>
      </ul>
    </nav><!-- .navbar -->

    <i class="mobile-nav-toggle mobile-nav-show bi bi-list"></i>
    <i class="mobile-nav-toggle mobile-nav-hide d-none bi bi-x"></i>

  </div>
 </header><!-- End Header -->
 <!-- End Header -->

 <main id="main">

  <!-- ======= Breadcrumbs ======= -->
  <div class="breadcrumbs">
   <div class="page-header d-flex align-items-center" style="background-image: url('');">
    <div class="container position-relative">
     <div class="row d-flex justify-content-center">
      <div class="col-lg-6 text-center">
       <h2>Disease Prediction</h2>
       <p>You will have the input box below where you can select your symptoms.</p>
       <p>You can input the number of symptoms you have and leave others blank.</p>
       <p>This is the list of the symptoms . If you have symptoms which are from this list
please enter the symptom in the same form as shown below.</p>
       <div class="container">
        <ul class="list" style="color: white">
          <li class="list-item">itching</li>
          <li class="list-item">muscle_pain</li>
          <li class="list-item">shivering</li>
          <li class="list-item">joint_pain</li>
          <li class="list-item">stomach_pain</li>
          <li class="list-item">vomiting</li>
          <li class="list-item">fatigue</li>
          <li class="list-item">weight_loss</li>
          <li class="list-item">restlessness</li>
          <li class="list-item">lethargy</li>
          <li class="list-item">high_fever</li>
          <li class="list-item">headache</li>
          <li class="list-item">dark_urine</li>
          <li class="list-item">nausea</li>
          <li class="list-item">coma</li>
          <li class="list-item">constipation</li>
          <li class="list-item">abdominal_pain</li>
          <li class="list-item">diarrhoea</li>
          <li class="list-item">mild_fever</li>
          <li class="list-item">malaise</li>
```

```
                <li class="list-item">phlegm</li>
                <li class="list-item">congestion</li>
                <li class="list-item">chest_pain</li>
                <li class="list-item">fast_heart_rate</li>
                <li class="list-item">neck_pain</li>
                <li class="list-item">dizziness</li>
                <li class="list-item">belly_pain</li>
                <li class="list-item">knee_pain</li>
                <li class="list-item">muscle_weakness</li>
                <li class="list-item">passage_of_gases</li>
                <li class="list-item">irritability</li>
                <li class="list-item">continuous_sneezing</li>
                <li class="list-item">puffy_face_and_eyes</li>
                <li class="list-item">abnormal_menstruation</li>
                <li class="list-item">increased_appetite</li>
                <li class="list-item">lack_of_concentration</li>
                <li class="list-item">visual_disturbances</li>
                <li class="list-item">receiving_blood_transfusion</li>
                <li class="list-item">pain_behind_the_eyes</li>
                <li class="list-item">history_of_alcohol_consumption</li>
                <li class="list-item">blood_in_sputum</li>
                <li class="list-item">yellowing_of_eyespalpitations</li>
                <li class="list-item">inflammatory_nails</li>
                <li class="list-item">yellow_crust_ooze</li>
              </ul>
            </div>
            </div>
          </div>
        </div>
      </div>
      <nav>
        <div class="container">
          <ol>
            <li><a href="/">Home</a></li>
            <li>Predict</li>
          </ol>
        </div>
      </nav>
    </div><!-- End Breadcrumbs -->

    <div class="row">
      <form action='/predict' method="post" class="col s12">
        <br>
        <div class="row">
          <div class="input-field col s4">
            <label for="s1"><b>Symptom-1</b></label>
```

```
          <br>
          <input id="s1" name="Symptom1" placeholder="Type your symptom here"  type="text" class="validate">
        </div>

        <div class="input-field col s4">
          <label for="s2"><b> Symptom-2</b></label>
          <br>
          <input id="s2" name="Symptom2" placeholder="Type your symptom here"  type="text" class="validate">
        </div>

        <div class="input-field col s4">
          <label for="s3"><b>Symptom-3</b></label>
          <br>
          <input id="s3" name="Symptom3" placeholder="Type your symptom here" type="text" class="validate">
        </div>
      </div>

      <br>
      <div class="row">
        <div class="input-field col s4">
          <label for="s4"><b>Symptom-4</b></label>
          <br>
          <input id="s4" name="Symptom4" placeholder="Type your symptom here"  type="text" class="validate">
        </div>

        <div class="input-field col s4">
          <label for="s5"><b> Symptom-5</b></label>
          <br>
          <input id="s5" name="Symptom5" placeholder="Type your symptom here"  type="text" class="validate">
        </div>

        <div class="input-field col s4">
          <label for="s6"><b>Symptom-6</b></label>
          <br>
          <input id="s6" name="Symptom6" placeholder="Type your symptom here" type="text" class="validate">
        </div>
      </div>

      <br>
      <div class="row">
```

```
        <div class="input-field col s4">
           <label for="s7"><b>Symptom-7</b></label>
           <br>
          <input id="s7" name="Symptom7" placeholder="Type your symptom here"  type="text"
class="validate">
        </div>

        <div class="input-field col s4">
          <label for="s8"><b> Symptom-8</b></label>
           <br>
          <input id="s8" name="Symptom8" placeholder="Type your symptom here"  type="text"
class="validate">
        </div>

        <div class="input-field col s4">
           <label for="s9"><b>Symptom-9</b></label>
           <br>
          <input id="s9" name="Symptom9" placeholder="Type your symptom here" type="text"
class="validate">
        </div>
       </div>

       <br>
       <div class="row center">
          <button type="submit" class="btn btn-success btn-lg" style="color:rgb(237, 239,
241)">Predict</button>
       </div>
     </form>
       </div>

     </div>
   </div>
   </div>
   </div>>
  </main><!-- End #main -->

  <!-- ======= Footer ======= -->
  <footer id="footer" class="footer">

    <div class="container">
     <div class="row gy-4">
       <div class="col-lg-5 col-md-12 footer-info">
         <a href="index.html" class="logo d-flex align-items-center">
           <span>Disease Prediction</span>
         </a>
         <p>Preventive Diagnosis at your convenience.</p>
```

```
        </div>

        <div class="col-lg-2 col-6 footer-links">


        <div class="col-lg-3 col-md-12 footer-contact text-center text-md-start">
          <h4>Contact Us</h4>
          <p>
            Example, <br>
            Pune,<br>
            India. <br><br>
            <strong>Phone:</strong> +1 5589 55488 55<br>
            <strong>Email:</strong> info@gmail.com<br>
          </p>

        </div>

      </div>
    </div>

    <div class="container mt-4">
      <div class="copyright">
        &copy; Copyright <strong><span>Disease Prediction</span></strong>. All Rights
Reserved
      </div>
      <div class="credits">
        <!-- All the links in the footer should remain intact. -->
        <!-- You can delete the links only if you purchased the pro version. -->
        <!-- Licensing information: https://bootstrapmade.com/license/ -->
        <!-- Purchase the pro version with working PHP/AJAX contact form:
https://bootstrapmade.com/impact-bootstrap-business-website-template/ -->
      </div>
    </div>

  </footer><!-- End Footer -->
  <!-- End Footer -->
  <a href="#" class="scroll-top d-flex align-items-center justify-content-center"><i class="bi bi-
arrow-up-short"></i></a>

  <div id="preloader"></div>

  <!-- Vendor JS Files -->
  <script src="static/vendor/bootstrap/js/bootstrap.bundle.min.js"></script>
  <script src="static/vendor/aos/aos.js"></script>
  <script src="static/vendor/glightbox/js/glightbox.min.js"></script>
  <script src="static/vendor/purecounter/purecounter_vanilla.js"></script>
```

```
  <script src="static/vendor/swiper/swiper-bundle.min.js"></script>
  <script src="static/vendor/isotope-layout/isotope.pkgd.min.js"></script>
  <script src="static/vendor/php-email-form/validate.js"></script>

  <!-- Template Main JS File -->
  <script src="static/js/main.js"></script>

</body>

</html>
```

App.py

```
from flask import Flask, render_template, request
import numpy as np
import pickle


model = pickle.load(open('model.pkl','rb'))
app = Flask(__name__)


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


@app.route('/details')
def pred():
    return render_template('details.html')


@app.route('/predict',methods=['POST','GET'])
def predict():
    col=['itching', 'continuous_sneezing', 'shivering', 'joint_pain',
       'stomach_pain', 'vomiting', 'fatigue', 'weight_loss', 'restlessness',
       'lethargy', 'high_fever', 'headache', 'dark_urine', 'nausea',
       'pain_behind_the_eyes', 'constipation', 'abdominal_pain', 'diarrhoea',
       'mild_fever', 'yellowing_of_eyes', 'malaise', 'phlegm', 'congestion',
       'chest_pain', 'fast_heart_rate', 'neck_pain', 'dizziness',
       'puffy_face_and_eyes', 'knee_pain', 'muscle_weakness',
       'passage_of_gases', 'irritability', 'muscle_pain', 'belly_pain',
       'abnormal_menstruation', 'increased_appetite', 'lack_of_concentration',
       'visual_disturbances', 'receiving_blood_transfusion', 'coma',
       'history_of_alcohol_consumption', 'blood_in_sputum', 'palpitations',
       'inflammatory_nails', 'yellow_crust_ooze']
    if request.method=='POST':
        inputt = [str(x) for x in request.form.values()]


        b=[0]*45
        for x in range(0,45):
            for y in inputt:
                if(col[x]==y):
                    b[x]=1
        b=np.array(b)
        b=b.reshape(1,45)
        prediction = model.predict(b)
        prediction = prediction[0]
    return render_template('results.html', prediction_text="The probable diagnosis says it could
```

```
be {}".format(prediction)
if __name__ == "__main__":
    app.run(debug=True,port=8000)
```

GitHub & Project Demo Link

Demo Link

https://drive.google.com/file/d/1_pUgqefddyp8YiWPiExsIWUQqUGD053V/view?usp=sharing

Github link

https://github.com/smartinternz02/SI-GuidedProject-611945-1698755699