**CORONARY ARTERY DISEASE PREDICTION**

**SYNOPSIS:**

1.Introduction

2. Overview Of The Project

3.Proposed System

4.Dataset And Sampling

5.System Architecture

6.Screenshot

7.Sample Coding

**INTRODUCTION**

Coronary Artery Diseases are the primary cause of death over the past decade. According to the World Health Organization, every year 12 million deaths occur worldwide due to Heart Disease. Over the years, CAD can also weaken your heart and lead to complications, including Arrhythmias (like atrial fibrillation), Cardiac arrest, cardiogenic shock and Heart failure. The main problem of coronary artery disease is a heart attack. The heart muscle starts to die because it’s not receiving enough blood. Heart Disease is even highlighted as a silent killer which leads to the death of the person without obvious symptoms. The early diagnosis of heart disease plays a vital role in making decisions on lifestyle changes in high-risk patients and in turn reduces the complications. ML is an emerging application of AI that uses different analytics and statistical techniques in order to improve the performance of particular machine learning from old data.

The objective of this project is to check whether the patient is likely to be diagnosed with any cardiovascular heart diseases based on their medical attributes such as gender, age, chest pain, fasting sugar level, etc. This helps in early prediction of the disease and is used in many ways, where as it is being provided with the input, in order to find the heart rate based on the health condition.

**OVERVIEW OF THE PROJECT**

# “This Coronary Artery Disease Prediction” website predicts the probabilities of heart condition and finally results whether the person has defective heart or not. This website includes the symptoms and the risk factors of the Coronary Artery Disease.

# This project analyzes the supervised learning modes of Logistic Regression. The project also mention scope for future research and different advancement possibilities. This Coronary Artery disease is a chronic disease that may last for years or be lifelong. Lab tests are often required and it needs diagnosis treatment. This can only help but condition can't be cured. So this system helps to predict the Coronary Artery Disease in the earliest stage.

**PROPOSED SYSTEM**

The proposed system starts with the collection of data and selecting the important attributes. Then the required data is pre-processed into the required format. The data is then divided into two parts training and testing data. The algorithms are applied and the model is trained using the training data. The accuracy of the system is obtained by testing the system using the testing data. This system is implemented using the following modules.

1.) Collection of Dataset

2.) Data Pre-Processing

3.) Splitting the Data into Training data & Test Data

4.) Training the model using Logistic Regression

5.) Building a Predictive System

**1.) Collection of Dataset**

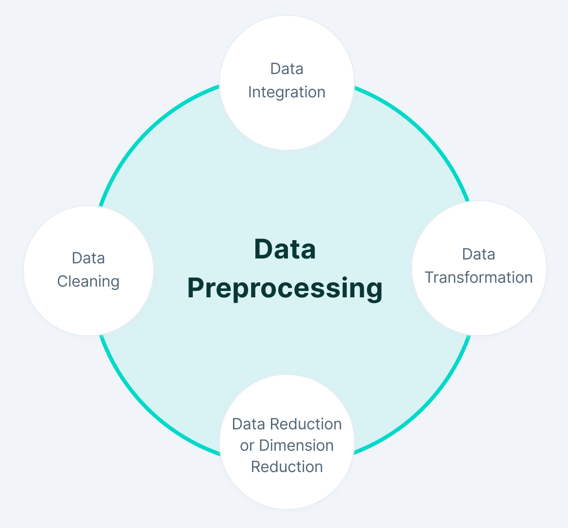
A dataset in machine learning is a collection of data pieces that can be treated by a computer as a single unit for analytic and prediction purposes. This means that the data collected should be made uniform and understandable for a machine that doesn't see data the same way as humans do. For this, after collecting the data, it's important to pre-process it by cleaning and completing it, as well as annotate the data by adding meaningful tags readable by a computer.

Moreover, a good dataset should correspond to certain quality and quantity standards. For smooth and fast training, you should make sure your dataset is relevant and well-balanced.

After the collection of the dataset, we split the dataset into training data and testing data. The training dataset is used for prediction model learning and testing data is used for evaluating the prediction model. For this project, 70% of training data is used and 30% of data is used for testing.

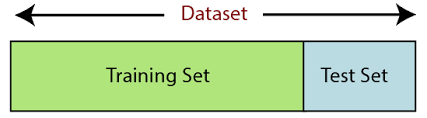
**2.) Data Pre-Processing**

Data pre-processing is an important step for the creation of a machine learning model. Initially, data may not be clean or in the required format for the model which can cause misleading outcomes. In pre-processing of data, we transform data into our required format. It is used to deal with noises, duplicates, and missing values of the dataset. Data pre-processing has the activities like importing datasets, splitting datasets, attribute scaling, etc. Pre-processing of data is required for improving the accuracy of the model.



**3.) Splitting the Data into Training data & Test Data**

In the context of Machine Learning, the split of our modelling dataset into training and testing samples is probably one of the earliest pre-processing steps that we need to undertake. The creation of different samples for training and testing helps us evaluate model performance. Scikit-learn alias **sklearn** is the most useful and robust library for machine learning in Python. The **scikit-learn library** provides us with the model\_selection module in which we have the splitter function train\_test\_split ().



**4.) Training the model using Logistic Regression**

**LOGISTIC REGRESSION ALGORITHM**

Logistic regression is one of the most popular Machine Learning algorithms, which comes under the Supervised Learning technique. It is used for predicting the categorical dependent variable using a given set of independent variables. Logistic regression predicts the output of a categorical dependent variable.

Therefore the outcome must be a categorical or discrete value. It can be either Yes or

No, 0 or 1, true or false, etc. but instead of giving the exact value as 0 and 1, it gives the probabilistic values which lie between 0 and 1.

In Logistic regression, instead of fitting a regression line, we fit an "S" Shaped Logistic function, which predicts two maximum values (0 or 1).The curve from the logistic function indicates the likelihood of something such as whether the cells are cancerous or not, a mouse is obese or not based on its weight, etc.

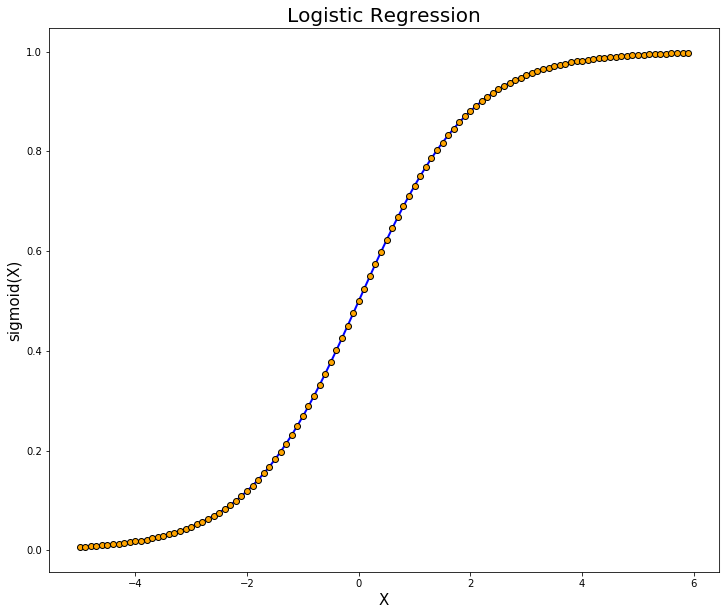
Logistic Regression is a significant machine learning algorithm because it has the ability to provide probabilities and classify new data using continuous and discrete datasets.

**Advantages:**

Logistic Regression is one of the simplest machine learning algorithms and is easy to implement yet provides great training efficiency in some cases. Also due to these reasons, training a model with this algorithm doesn't require high computation power.The predicted parameters (trained weights) give inference about the importance of each feature. The direction of association i.e. positive or negative is also given. So we can use Logistic Regression to find out the relationship between the features.

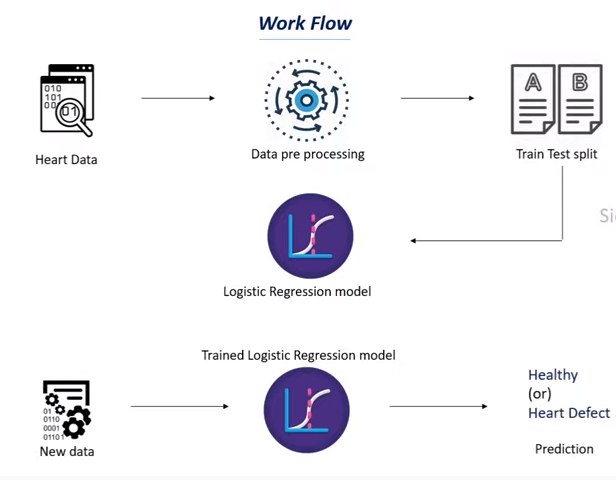
This algorithm allows models to be updated easily to reflect new data, unlike

Decision Tree or Support Vector Machine. The update can be done using stochastic gradient descent.



**5.) Building a Predictive System**

Machine learning algorithm Logistic Regression is used for classification. Comparative analysis is performed among algorithm and is used for heart disease prediction. “Prediction” refers to the output of an algorithm after it has been trained on a historical dataset and applied to new data when forecasting the likelihood of a particular outcome.

****

**DATASET AND SAMPLING**

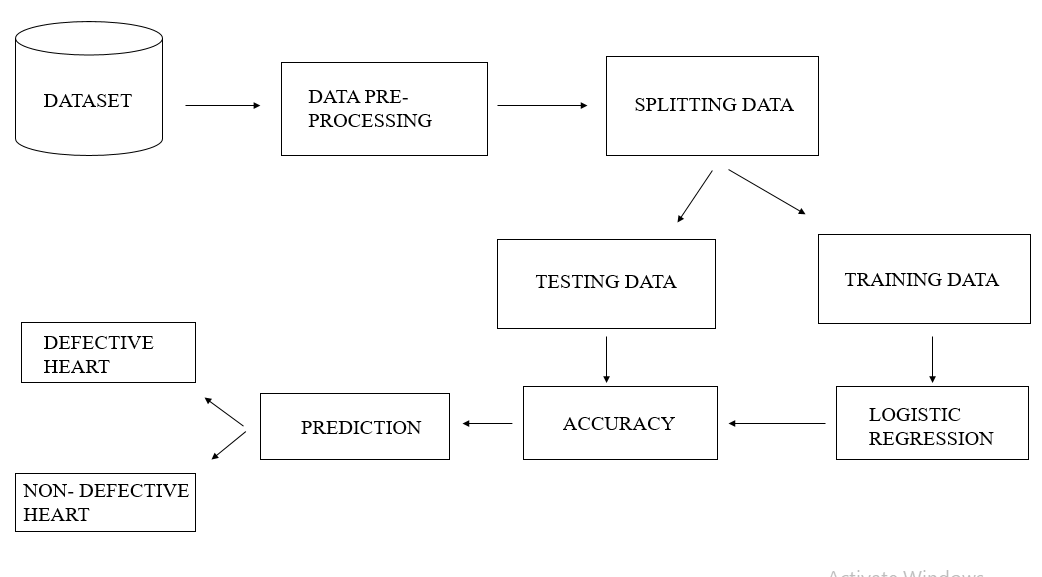
**Data Resource**

The Cleveland heart dataset from the UCI machine learning repository has been used for the experiments. The dataset consists of 14 attributes and 303 instances. There are 8 categorical attributes and 6 numeric attributes. Patients from age 29 to 79 have been selected in this dataset. Male patients are denoted by a gender value 1 and female patients are denoted by gender value 0. Four types of chest pain can be considered as indicative of heart disease. Type 1 angina is caused by reduced blood flow to the heart muscles because of narrowed coronary arteries. Type 1 Angina is a chest pain that occurs during mental or emotional stress. Non-angina chest pain may be caused due to various reasons and may not often be due to actual heart disease. The fourth type, Asymptomatic, may not be a symptom of heart disease. The next attribute trestbps is the reading of the resting blood pressure. Chol is the cholesterol level. Fbs is the fasting blood sugar level; the value is assigned as 1 if the fasting blood sugar is below 120 mg/dl and 0 if it is above. Restecg is the resting electrocardiographic result, thalach is the maximum heart rate, exang is the exercise induced angina which is recorded as 1 if there is pain and 0 if there is no pain, oldpeak is the ST depression induced by exercise, slope is the slope of the peak exercise ST segment, ca is the number of major vessels colored by fluoroscopy, thal is the duration of the exercise test in minutes, and num is the class attribute. The class attribute has a value of 0 for normal and 1 for patients diagnosed with heart disease. The description of the dataset is shown in the table.

**DATASET INFORMATION**

|  |  |  |
| --- | --- | --- |
| Attribute | Description | Range |
| Age | Age of person in years | 29-79 |
| Sex | Gender of person (1-M 0-F) | 0,1 |
| Cp | Chest pain type | 1,2,3,4 |
| Trestbps | Resting blood pressure in mm Hg | 94-200 |
| Chol | Serum cholesterol in mg/dl | 126-564 |
| Fbs | Fasting blood sugar in mg/dl | 0,1 |
| Restecg | Resting Electrocardiographic results | 0,1,2 |
| Thalach | Maximum heart rate achieved | 71-202 |
| Exang | Exercise Induced Angina | 0,1 |
| OldPeak | ST depression induced by exercise relative to rest | 1-3 |
| Slope | Slope of the Peak Exercise ST segment | 1,2,3 |
| Ca | Number of major vessels colored by  Fluoroscopy | 0-3 |
| Thal | 3-Normal, 6-Fixed Defect, 7 – Reversible Defect | 3,6,7 |
| Target | Class Attribute | 0,1 |

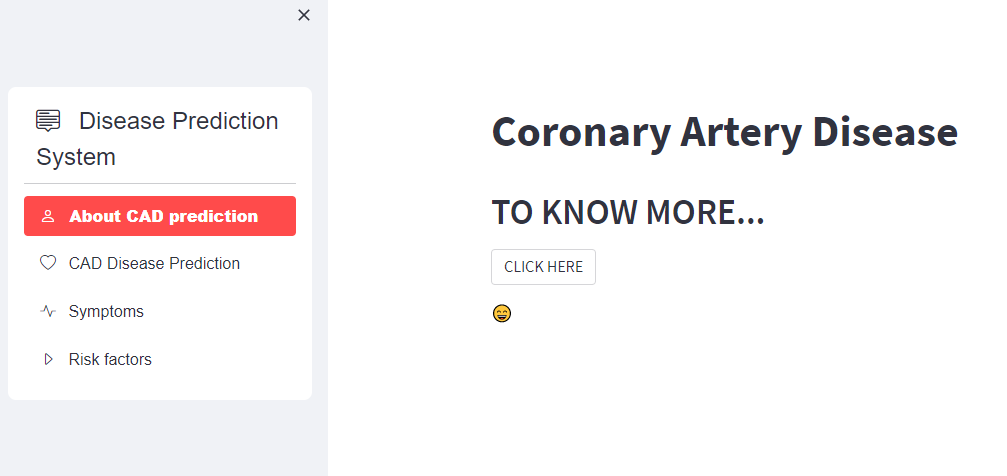
**SYSTEM ARCHITECTURE**

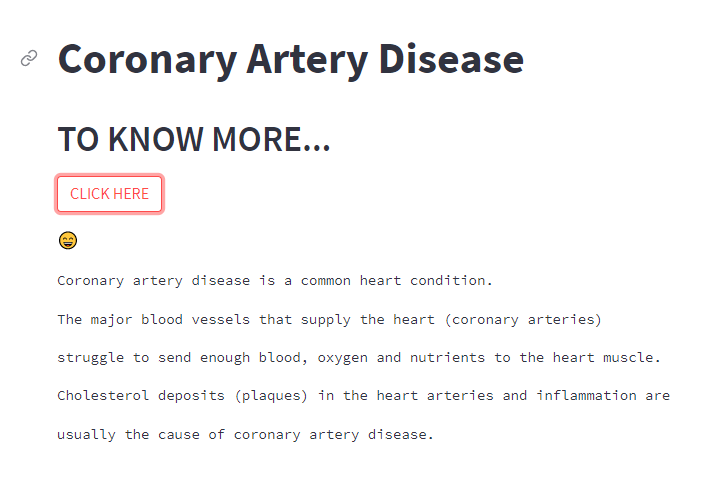
****

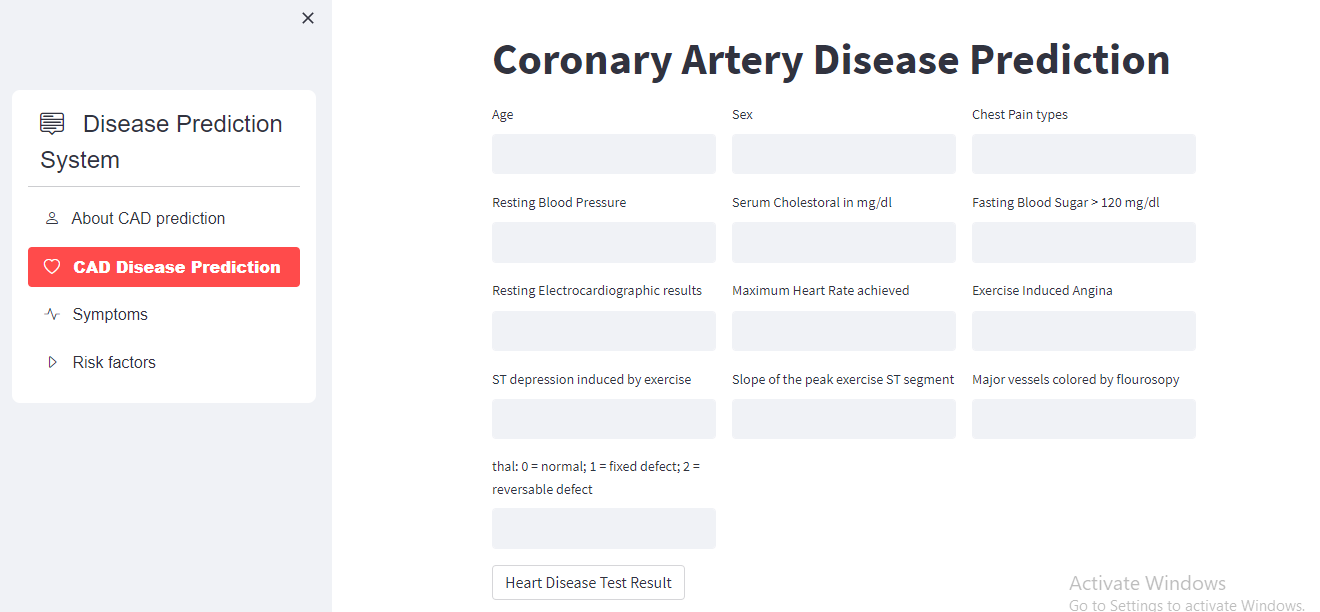
“Coronary Artery Disease Prediction” allows user to predict whether the person has defective heart or non – defective heart. It is a WEB application. The main problem of coronary artery disease is a heart attack. The early diagnosis of heart disease plays a vital role in making decisions on lifestyle changes in high-risk patients and in turn reduces the complications. The main objective of this project is to check whether the patient is likely to be diagnosed with any cardiovascular heart diseases based on their medical attributes such as gender, age, chest pain, fasting sugar level, etc. This helps in early prediction of the disease . This project makes use of Machine Learning classification algorithm which works efficiently to find the person has defective heart or not. The users can get their results as soon as they fill the attributes. This website is developed using the languages Python, Streamlit framework integrates the model and user interface. It is a framework which is used to implement the webpage with python. It also makes use of the Machine Learning algorithm Logistic Regression which is a classification model used to classify the model and predicts the output.

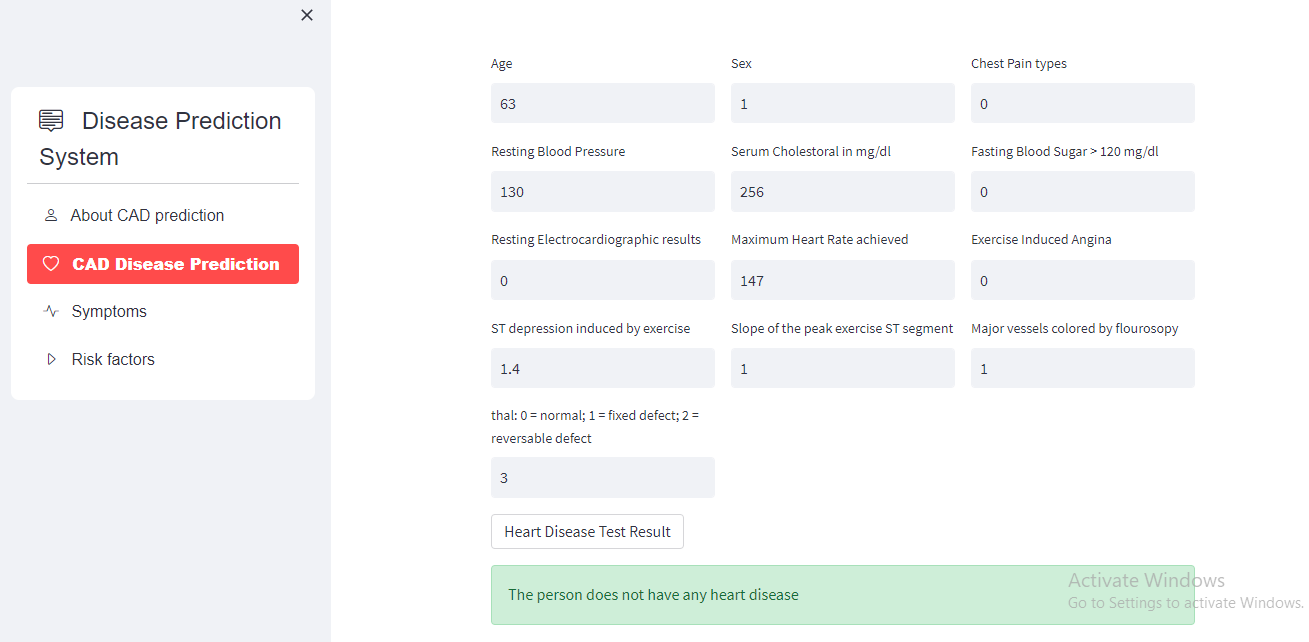
**APPENDIX**

**SCREENSHOT**









**SAMPLE CODING**

**#Importing the Dependencies**

import numpy as np

import pandas as pd

from sklearn.model\_selection import train\_test\_split

from sklearn.linear\_model import LogisticRegression

from sklearn.metrics import accuracy\_score

**#Data Collection and Processing**

# loading the csv data to a Pandas DataFrame

heart\_data = pd.read\_csv('/content/heart.csv')

# print first 5 rows of the dataset

heart\_data.head()

# print last 5 rows of the dataset

heart\_data.tail()

# number of rows and columns in the dataset

heart\_data.shape

# getting some info about the data

heart\_data.info()

# checking for missing values

heart\_data.isnull().sum()

# statistical measures about the data

heart\_data.describe()

# checking the distribution of Target Variable

heart\_data['target'].value\_counts()

**#Splitting the Features and Target**

X = heart\_data.drop(columns='target', axis=1)

Y = heart\_data['target']

print(X)

print(Y)

**#Splitting the Data into Training data & Test Data**

X\_train, X\_test, Y\_train, Y\_test = train\_test\_split(X, Y, test\_size=0.2, stratify=Y, random\_state=2)

print(X.shape, X\_train.shape, X\_test.shape)

**#Model Training**

model = LogisticRegression()

model.fit(X\_train, Y\_train)

**#Model Evaluation**

# accuracy on training data

X\_train\_prediction = model.predict(X\_train)

training\_data\_accuracy = accuracy\_score(X\_train\_prediction, Y\_train)

print('Accuracy on Training data : ', training\_data\_accuracy)

# accuracy on test data

X\_test\_prediction = model.predict(X\_test)

test\_data\_accuracy = accuracy\_score(X\_test\_prediction, Y\_test)

print('Accuracy on Test data : ', test\_data\_accuracy)

**#Building a Predictive System**

input\_data = (62,0,0,140,268,0,0,160,0,3.6,0,2,2)

# change the input data to a numpy array

input\_data\_as\_numpy\_array= np.asarray(input\_data)

# reshape the numpy array as we are predicting for only on instance

input\_data\_reshaped = input\_data\_as\_numpy\_array.reshape(1,-1)

prediction = model.predict(input\_data\_reshaped)

print(prediction)

if (prediction[0]== 0):

  print('The Person does not have a Heart Disease')

else:

  print('The Person has Heart Disease')

**#Saving the trained model**

import pickle

filename = 'heart\_disease\_model.sav'

pickle.dump(model, open(filename, 'wb'))

# loading the saved model

loaded\_model = pickle.load(open('heart\_disease\_model.sav', 'rb'))

for column in X.columns:

  print(column)

**#FRONT-END**

import pickle

import streamlit as st

import numpy as np

from streamlit\_option\_menu import option\_menu

heart\_disease\_model = pickle.load(open('C:/Users/DELL/Desktop/disease prediction/saved models/heart\_disease\_model.sav', 'rb'))

with st.sidebar:

selected = option\_menu('Disease Prediction System',

['About CAD prediction','CAD Disease Prediction','Symptoms','Risk factors'],

icons=['person','heart','activity'],

default\_index=0)

if (selected == 'About CAD prediction'):

# page title

st.title('Coronary Artery Disease')

st.header('TO KNOW MORE...')

re=st.button('CLICK HERE')

st.write(":smile:")

if re:

st.text(' Coronary artery disease is a common heart condition.')

st.text(' The major blood vessels that supply the heart (coronary arteries)')

st.text('struggle to send enough blood, oxygen and nutrients to the heart muscle.')

st.text(' Cholesterol deposits (plaques) in the heart arteries and inflammation are ')

st.text('usually the cause of coronary artery disease.')

if (selected == 'Symptoms'):

st.title('Coronary Artery Disease Symptoms')

st.text(' For many people, the first clue that they have Coronary Artery Disease is a heart attack.')

st.text('Symptoms of heart attack include:')

st.text(' 1.Chest pain or discomfort (angina)')

st.text(' 2.Weakness, light-headedness, nausea (feeling sick to your stomach), or a cold sweat')

st.text(' 3.Pain or discomfort in the arms or shoulder')

st.text(' 4.Shortness of breath')

if (selected == 'Risk factors'):

st.title('Coronary Artery Disease Risk factors')

st.write('Coronary artery disease risk factors include:')

st.subheader('Age:')

st.text('Getting older increases the risk of damaged and narrowed arteries.')

st.subheader('Sex:')

st.text('Men are generally at greater risk of coronary artery disease. However, the risk for women increases after menopause.')

st.subheader('Smoking:')

st.text('If you smoke, quit. Smoking is bad for heart health. People who smoke have a increased risk of heart disease.')

st.subheader('High blood pressure:')

st.text('Uncontrolled high blood pressure can make arteries hard and stiff (arterial stiffness).')

st.text('The coronary arteries may become narrow, slowing blood flow.')

st.subheader('Diabetes:')

st.text('Diabetes increases the risk of coronary artery disease. ')

st.subheader('Overweight or obesity:')

st.text('Excess body weight is bad for overall health. Obesity can lead to type 2 diabetes and high blood pressure. ')

st.subheader('Chronic kidney disease:')

st.text('Having long-term kidney disease increases the risk of coronary artery disease.')

st.subheader('Not getting enough exercise:')

st.text('Physical activity is important for good health. A lack of exercise is linked to coronary artery disease .')

st.subheader('A lot of stress:')

st.text('Emotional stress may damage the arteries and worsen other risk factors for coronary artery disease.')

st.subheader('Unhealthy diet:')

st.text('Eating foods with a lot of saturated fat, trans fat, salt and sugar can increase the risk of coronary artery disease.')

st.subheader('Alcohol use:')

st.text('Heavy alcohol use can lead to heart muscle damage. It can also worsen other risk factors of coronary artery disease.')

st.subheader('Amount of sleep:')

st.text('Too little and too much sleep have both been linked to an increased risk of heart disease.')

if (selected == 'CAD Disease Prediction'):

# page title

st.title('Coronary Artery Disease Prediction ')

col1, col2, col3 = st.columns(3)

with col1:

age = st.text\_input('Age')

with col2:

sex = st.text\_input('Sex')

with col3:

cp = st.text\_input('Chest Pain types')

with col1:

trestbps = st.text\_input('Resting Blood Pressure')

with col2:

chol = st.text\_input('Serum Cholestoral in mg/dl')

with col3:

fbs = st.text\_input('Fasting Blood Sugar > 120 mg/dl')

with col1:

restecg = st.text\_input('Resting Electrocardiographic results')

with col2:

thalach = st.text\_input('Maximum Heart Rate achieved')

with col3:

exang = st.text\_input('Exercise Induced Angina')

with col1:

oldpeak = st.text\_input('ST depression induced by exercise')

with col2:

slope = st.text\_input('Slope of the peak exercise ST segment')

with col3:

ca = st.text\_input('Major vessels colored by flourosopy')

with col1:

thal = st.text\_input('thal: 0 = normal; 1 = fixed defect; 2 = reversable defect')

lst = [[float(i) for i in [age, sex, cp, trestbps, chol, fbs, restecg,thalach,exang,oldpeak,slope,ca,thal]if i !=""]]

# code for Prediction

heart\_diagnosis = ''

# creating a button for Prediction

#b = np.array(a, dtype=float) # convert using numpy

if st.button('Heart Disease Test Result'):

heart\_prediction = heart\_disease\_model.predict(lst)

print(heart\_prediction)

if (heart\_prediction[0] == 1):

heart\_diagnosis = 'The person is having heart disease'

else:

heart\_diagnosis = 'The person does not have any heart disease'

st.success(heart\_diagnosis)