COMPANY NAME

COMPSOFT TECHNOLOGIES

AN INTERSHIP REPORT ON

“VIRTUAL ASSISTANCE FOR VISUALLY IMPAIRED”

Bachelor of Engineering

In
Information Science and Engineering

Submitted by : ROHITH.E[1AP19CS030] (TL)

A.P.S. COLLEGE OF ENGINEERING
Bangalore

COLLEGE NAME : APS COLLEGE OF ENGINEERING
2022

ABOUT THE COMPANY

We are a digital service provider that aims to provide software, designing and marketing solutions to individuals and businesses, we believe that service and quality is the key to success

We provide all kinds of technological and designing solutions from Billing Software to Web Designs or any custom demand that you may have. Experience the service like none other!

Some of our services include:

Development - We develop responsive, functional and super-fast websites.
We keep User Experience in mind while creating websites. A website should load quickly and should be accessible even on a small view-port and slow internet connection.

Mobile Application - We offer a wide range of professional Android, iOS &

Hybrid app development services for our global clients, from a start up to a large enterprise.

Design - We offer professional Graphic design, Brochure design & Logo design. We are experts in crafting visual content to convey the right message to the customers.

Consultancy - We are here to provide you with expert advice on your design and development requirement.

Videos - We create a polished professional video that impresses your audience.

TABLE OF CONTENTS

OVERVIEW OF THE PROJECT

Project Name: Virtual Assistance for Visually Impaired

Team Members: ROHITH.E

Healthcare is very important to lead a good life. However, it is very difficult to obtain a consultation with a doctor for every health problem. The idea is to create a medical chatbot using Artificial Intelligence that can diagnose the disease and provide basic details about the disease before consulting a doctor. This will help to reduce healthcare costs and improve accessibility to medical knowledge through medical chatbots. Chatbots are computer programs that use natural language to interact with users. The chatbot stores the data in the database to identify the sentence keywords and make a query decision and answer the question. Ranking and sentence similarity calculations are performed using n-gram, TFIDF and cosine similarity. The score will be obtained for each sentence from the given input sentence and more similar sentences will be obtained for the query given. The third-party, the expert program, handles the question presented to the bot that is not understood or is not present in the database.

TOOLS USED

Software Requirements

- Visual Studio Code 2019.
- Google Chrome or Microsoft Edge of latest version.
- Microsoft Excel
- Windows 10 OS

Hardware Requirements

- Pentium 200-MHz computer with a minimum of 64 MB of RAM (128 MB of RAM recommended).
- Monitor with a refresh rate of at least 40Hz for a smooth GUI experience (optional).

IMPLEMENTATION

Source Code (PYTHON):

```
  import pandas as pd
  import pyttsx3
  from sklearn import preprocessing
  from sklearn.tree import
DecisionTreeClassifier,_tree
  import numpy as np
  from sklearn.model_selection import
train_test_split
  from sklearn.model_selection import
cross_val_score
  from sklearn.svm import SVC
  import csv
  import warnings
  warnings.filterwarnings("ignore",
category=DeprecationWarning)

  training = pd.read_csv('Training.csv')
  testing= pd.read_csv('Testing.csv')
  cols= training.columns
  cols= cols[:-1]
  x = training[cols]
  y = training['prognosis']
  y1= y


  reduced_data =
training.groupby(training['prognosis']).max()

  #mapping strings to numbers
  le = preprocessing.LabelEncoder()
  le.fit(y)
  y = le.transform(y)


  x_train, x_test, y_train, y_test =
train_test_split(x, y, test_size=0.33,
random_state=42)
  testx    = testing[cols]
  testy    = testing['prognosis']
  testy    = le.transform(testy)
  clf1  = DecisionTreeClassifier()
  clf = clf1.fit(x_train,y_train)
  # print(clf.score(x_train,y_train))
  # print ("cross result========")
  scores = cross_val_score(clf, x_test, y_test,
cv=3)
  # print (scores)
  print (scores.mean())



  model=SVC()
  model.fit(x_train,y_train)
  print("for svm: ")
  print(model.score(x_test,y_test))

  importances = clf.feature_importances_
  indices = np.argsort(importances)[::-1]
  features = cols

  def readn(nstr):
      engine = pyttsx3.init()

      engine.setProperty('voice', "english+f5")
      engine.setProperty('rate', 130)

      engine.say(nstr)
```

```
       engine.runAndWait()
       engine.stop()



   severityDictionary=dict()
   description_list = dict()
   precautionDictionary=dict()

   symptoms_dict = {}

   for index, symptom in enumerate(x):
          symptoms_dict[symptom] = index
   def calc_condition(exp,days):
       sum=0
       for item in exp:
            sum=sum+severityDictionary[item]
       if((sum*days)/(len(exp)+1)>13):
           print("You should take the
consultation from doctor. ")
       else:
           print("It might not be that bad but
you should take precautions.")



   def getDescription():
       global description_list
       with open('symptom_Description.csv')
as csv_file:
        csv_reader = csv.reader(csv_file,
delimiter=',')
        line_count = 0
        for row in csv_reader:
           _description={row[0]:row[1]}

description_list.update(_description)




  def getSeverityDict():
     global severityDictionary
     with open('symptom_severity.csv') as
csv_file:

        csv_reader = csv.reader(csv_file,
delimiter=',')
        line_count = 0
        try:
           for row in csv_reader:
              _diction={row[0]:int(row[1])}

severityDictionary.update(_diction)
        except:
           pass
  def getprecautionDict():
      global precautionDictionary
      with open('symptom_precaution.csv') as
csv_file:

          csv_reader = csv.reader(csv_file,
delimiter=',')
          line_count = 0
          for row in csv_reader:

_prec={row[0]:[row[1],row[2],row[3],row[4]]
}
              precautionDictionary.update(_prec)



  def getInfo():
      # name=input("Name:")
      print("Welcome!!")
      print("Your Name \t",end="->")
      name=input("")
      print("Hello ",name)

  def check_pattern(dis_list,inp):
      import re
      pred_list=[]
      ptr=0
      patt = "^" + inp + "$"
      regexp = re.compile(inp)
```

```
        for item in dis_list:

            # print(f"comparing {inp} to {item}")
            if regexp.search(item):
                pred_list.append(item)
                # return 1,item
        if(len(pred_list)>0):
            return 1,pred_list
        else:
            return ptr,item
    def sec_predict(symptoms_exp):
        df = pd.read_csv('Training.csv')
        X = df.iloc[:, :-1]
        y = df['prognosis']
        X_train, X_test, y_train, y_test =
train_test_split(X, y, test_size=0.3,
random_state=20)
        rf_clf = DecisionTreeClassifier()
        rf_clf.fit(X_train, y_train)

        symptoms_dict = {}

        for index, symptom in enumerate(X):
            symptoms_dict[symptom] = index

        input_vector =
np.zeros(len(symptoms_dict))
        for item in symptoms_exp:
          input_vector[[symptoms_dict[item]]] =
1


        return rf_clf.predict([input_vector])


    def print_disease(node):
        #print(node)
        node = node[0]
        #print(len(node))
        val  = node.nonzero()
        # print(val)
        disease = le.inverse_transform(val[0])
        return disease
    def tree_to_code(tree, feature_names):
        tree_ = tree.tree_
        # print(tree_)
        feature_name = [
            feature_names[i] if i !=
_tree.TREE_UNDEFINED else "undefined!"
            for i in tree_.feature
        ]

        chk_dis=",".join(feature_names).split(",")
        symptoms_present = []
```

```
        # conf_inp=int()
        while True:

            print("Enter the symptom you are
experiencing \t",end="->")
            disease_input = input("")

conf,cnf_dis=check_pattern(chk_dis,disease
_input)
            if conf==1:
                print("searches related to input: ")
                for num,it in enumerate(cnf_dis):
                    print(num,")",it)
                if num!=0:
                    print(f"Select the one you meant
(0 - {num}):  ", end="")
                    conf_inp = int(input(""))
                else:
                    conf_inp=0

                disease_input=cnf_dis[conf_inp]
                break
                # print("Did you mean:
",cnf_dis,"?(yes/no) :",end="")
                # conf_inp = input("")
                # if(conf_inp=="yes"):
                #     break
            else:
                print("Enter valid symptom.")

        while True:
            try:
                num_days=int(input("Okay. From
how many days ? : "))
                break
            except:
                print("Enter number of days:")
        def recurse(node, depth):
            indent = "  " * depth
            if tree_.feature[node] !=
_tree.TREE_UNDEFINED:
                name = feature_name[node]
                threshold = tree_.threshold[node]

                if name == disease_input:
                    val = 1
                else:
                    val = 0
                if  val <= threshold:
                    recurse(tree_.children_left[node],
depth + 1)
                else:

symptoms_present.append(name)
```

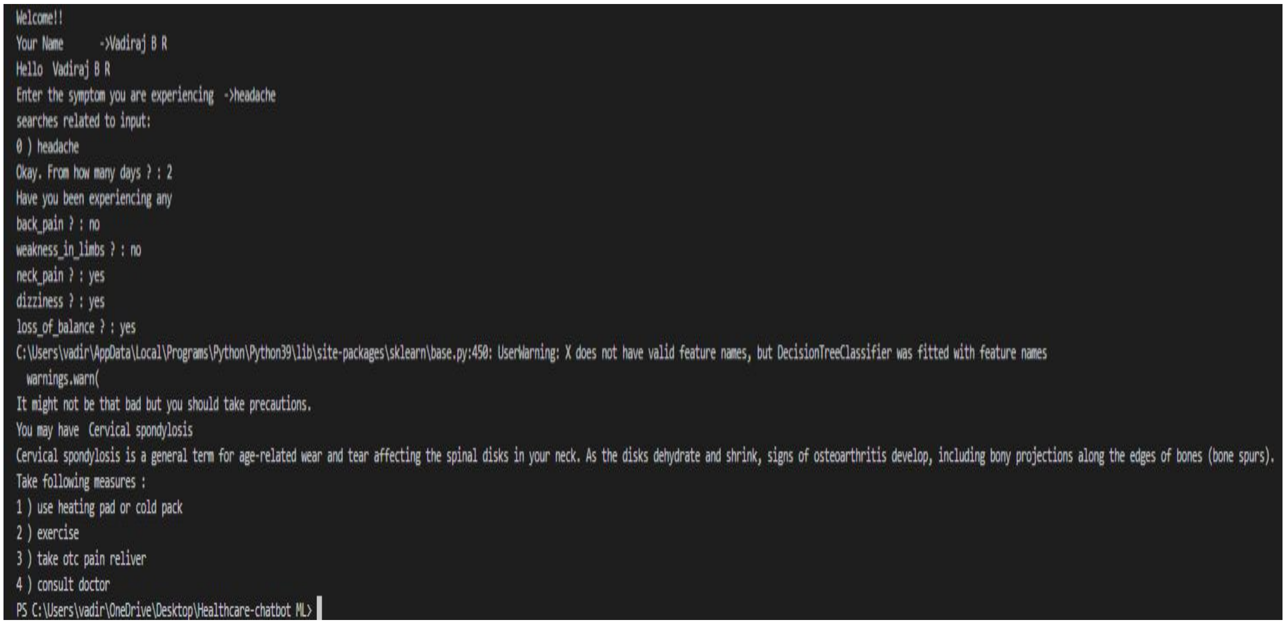
```
recurse(tree_.children_right[node], depth +
1)
        else:
            present_disease =
print_disease(tree_.value[node])
            # print( "You may have " +
present_disease )
            red_cols = reduced_data.columns
            symptoms_given =
red_cols[reduced_data.loc[present_disease].
values[0].nonzero()]
            # dis_list=list(symptoms_present)
            # if len(dis_list)!=0:
            #     print("symptoms present  " +
str(list(symptoms_present)))
            # print("symptoms given "  +
str(list(symptoms_given)) )
            print("Have you been experiencing
any ")
            symptoms_exp=[]
            for syms in list(symptoms_given):
                inp=""
                print(syms,"? : ",end='')
                while True:
                    inp=input("")
                    if(inp=="yes" or inp=="no"):
                        break
                    else:
                        print("provide proper
answers i.e. (yes/no) : ",end="")
                if(inp=="yes"):
                    symptoms_exp.append(syms)


second_prediction=sec_predict(symptoms_e
xp)
            # print(second_prediction)

calc_condition(symptoms_exp,num_days)

if(present_disease[0]==second_prediction[0]
):
                print("You may have ",
present_disease[0])


print(description_list[present_disease[0]])


                # readn(f"You may have
{present_disease[0]}")
                #
readn(f"{description_list[present_disease[0]]
}")


            else:
                print("You may have ",
present_disease[0], "or ",
second_prediction[0])
```

```
print(description_list[present_disease[0]])

print(description_list[second_prediction[0]])

            #
print(description_list[present_disease[0]])

precution_list=precautionDictionary[present
_disease[0]]
            print("Take following measures : ")
            for  i,j in enumerate(precution_list):
                print(i+1,")",j)

            # confidence_level =
(1.0*len(symptoms_present))/len(symptoms
_given)
            # print("confidence level is " +
str(confidence_level))

     recurse(0, 1)
  getSeverityDict()
  getDescription()
  getprecautionDict()
  getInfo()
  tree_to_code(clf,cols)
```

IMPLEMENTATION

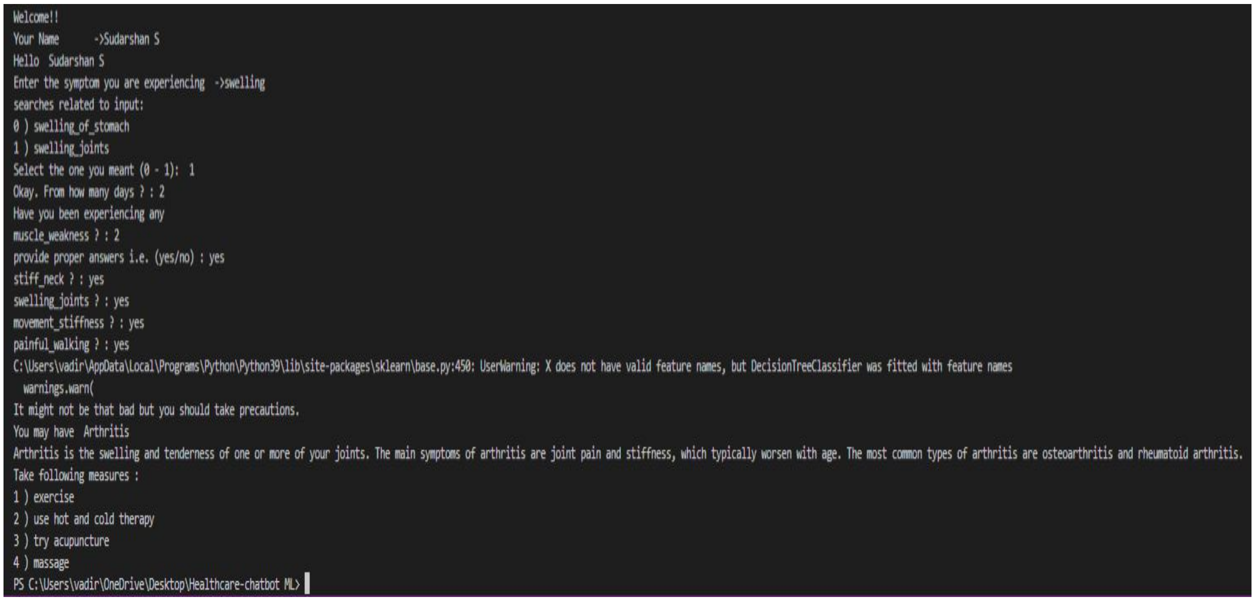
Snapshots :

SCREENSHOT 1



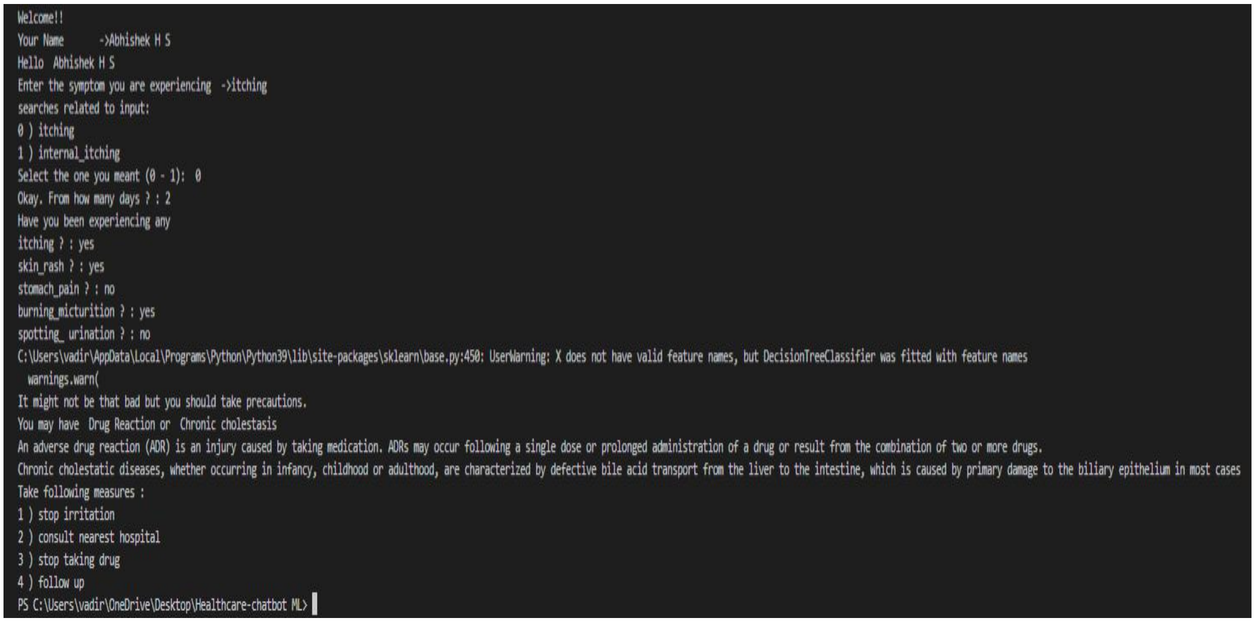
```
Welcome!!
Your Name       ->Vadiraj B R
Hello  Vadiraj B R
Enter the symptom you are experiencing  ->headache
searches related to input:
0 ) headache
Okay. From how many days ? : 2
Have you been experiencing any
back_pain ? : no
weakness_in_limbs ? : no
neck_pain ? : yes
dizziness ? : yes
loss_of_balance ? : yes
C:\Users\vadir\AppData\Local\Programs\Python\Python39\lib\site-packages\sklearn\base.py:450: UserWarning: X does not have valid feature names, but DecisionTreeClassifier was fitted with feature names
  warnings.warn(
It might not be that bad but you should take precautions.
You may have  Cervical spondylosis
Cervical spondylosis is a general term for age-related wear and tear affecting the spinal disks in your neck. As the disks dehydrate and shrink, signs of osteoarthritis develop, including bony projections along the edges of bones (bone spurs).
Take following measures :
1 ) use heating pad or cold pack
2 ) exercise
3 ) take otc pain reliver
4 ) consult doctor
PS C:\Users\vadir\OneDrive\Desktop\Healthcare-chatbot ML>
```

SCREENSHOT 2



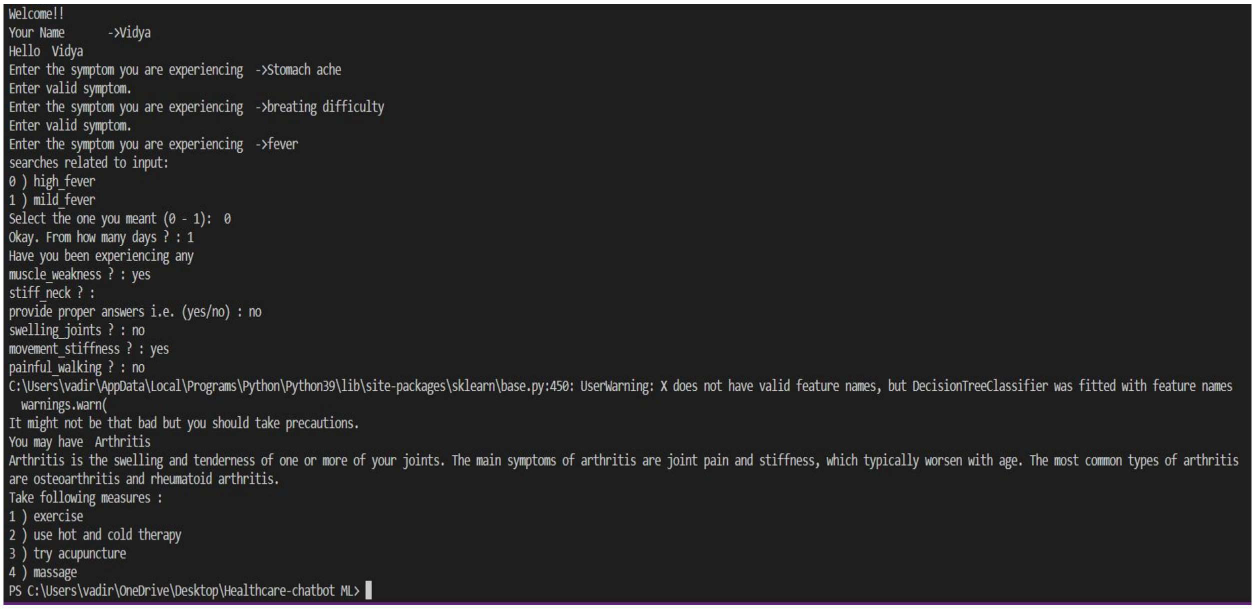
```
Welcome!!
Your Name       ->Sudarshan S
Hello  Sudarshan S
Enter the symptom you are experiencing  ->swelling
searches related to input:
0 ) swelling_of_stomach
1 ) swelling_joints
Select the one you meant (0 - 1):  1
Okay. From how many days ? : 2
Have you been experiencing any
muscle_weakness ? : 2
provide proper answers i.e. (yes/no) : yes
stiff_neck ? : yes
swelling_joints ? : yes
movement_stiffness ? : yes
painful_walking ? : yes
C:\Users\vadir\AppData\Local\Programs\Python\Python39\lib\site-packages\sklearn\base.py:450: UserWarning: X does not have valid feature names, but DecisionTreeClassifier was fitted with feature names
  warnings.warn(
It might not be that bad but you should take precautions.
You may have  Arthritis
Arthritis is the swelling and tenderness of one or more of your joints. The main symptoms of arthritis are joint pain and stiffness, which typically worsen with age. The most common types of arthritis are osteoarthritis and rheumatoid arthritis.
Take following measures :
1 ) exercise
2 ) use hot and cold therapy
3 ) try acupuncture
4 ) massage
PS C:\Users\vadir\OneDrive\Desktop\Healthcare-chatbot ML>
```

SCREENSHOT 3



```
Welcome!!
Your Name       ->Abhishek H S
Hello  Abhishek H S
Enter the symptom you are experiencing  ->itching
searches related to input:
0 ) itching
1 ) internal_itching
Select the one you meant (0 - 1):  0
Okay. From how many days ? : 2
Have you been experiencing any
itching ? : yes
skin_rash ? : yes
stomach_pain ? : no
burning_micturition ? : yes
spotting_ urination ? : no
C:\Users\vadir\AppData\Local\Programs\Python\Python39\lib\site-packages\sklearn\base.py:450: UserWarning: X does not have valid feature names, but DecisionTreeClassifier was fitted with feature names
  warnings.warn(
It might not be that bad but you should take precautions.
You may have  Drug Reaction or  Chronic cholestasis
An adverse drug reaction (ADR) is an injury caused by taking medication. ADRs may occur following a single dose or prolonged administration of a drug or result from the combination of two or more drugs.
Chronic cholestatic diseases, whether occurring in infancy, childhood or adulthood, are characterized by defective bile acid transport from the liver to the intestine, which is caused by primary damage to the biliary epithelium in most cases
Take following measures :
1 ) stop irritation
2 ) consult nearest hospital
3 ) stop taking drug
4 ) follow up
PS C:\Users\vadir\OneDrive\Desktop\Healthcare-chatbot ML>
```

SCREENSHOT 4



```
Welcome!!
Your Name       ->Vidya
Hello  Vidya
Enter the symptom you are experiencing  ->Stomach ache
Enter valid symptom.
Enter the symptom you are experiencing  ->breating difficulty
Enter valid symptom.
Enter the symptom you are experiencing  ->fever
searches related to input:
0 ) high_fever
1 ) mild_fever
Select the one you meant (0 - 1):  0
Okay. From how many days ? : 1
Have you been experiencing any
muscle_weakness ? : yes
stiff_neck ? :
provide proper answers i.e. (yes/no) : no
swelling_joints ? : no
movement_stiffness ? : yes
painful_walking ? : no
C:\Users\vadir\AppData\Local\Programs\Python\Python39\lib\site-packages\sklearn\base.py:450: UserWarning: X does not have valid feature names, but DecisionTreeClassifier was fitted with feature names
  warnings.warn(
It might not be that bad but you should take precautions.
You may have  Arthritis
Arthritis is the swelling and tenderness of one or more of your joints. The main symptoms of arthritis are joint pain and stiffness, which typically worsen with age. The most common types of arthritis
are osteoarthritis and rheumatoid arthritis.
Take following measures :
1 ) exercise
2 ) use hot and cold therapy
3 ) try acupuncture
4 ) massage
PS C:\Users\vadir\OneDrive\Desktop\Healthcare-chatbot ML>
```

BIBLIOGRAPHY

https://github.com/itachi9604/healthcare-chatbot.git

https://www.w3schools.com/python/python_json.asp

https://www.w3schools.com/python/python_pip.asp

https://www.w3schools.com/excel/excel_tables.php

About my TEAM

ROHITH.E [Team Leader] has done majority part of the Database in the project anddeployment of python code.