PredictWell: Chronic Disease Predictor

Mini Project 2B Report

Submitted in partial fulfillment of the requirement of University of Mumbai

For the Degree of

(Computer Engineering)

By

1) **Manav Rajpurohit** **ID No: TU3F2122034**

2) **Prateek Arote** **ID No: TU3F2122043**

3) **Prasad Shinde** **ID No: TU3F2122047**

Under the Guidance of

Prof. Siddharth Hariharan

Department of Computer Engineering

TERNA ENGINEERING COLLEGE

Plot no.12, Sector-22, Opp. Nerul Railway station,

Phase-11, Nerul (w), Navi Mumbai 400706

UNIVERSITY OF MUMBAI

TERNA ENGINEERING COLLEGE, NERUL, NAVI MUMBAI

Department of Computer Engineering

Academic Year 2023-24

CERTIFICATE

This is to certify that the mini project 2 B entitles " **PredictWell:** Chronic Disease Predictor" is a bonafide work of

1) Manav Rajpurohit	**ID No: TU3F2122034**
2) Prateek Arote	**ID No: TU3F2122043**
3) Prasad Shinde	**ID No: TU3F2122047**

submitted to the University of Mumbai in partial fulfillment of the requirement for the award of the Bachelor of Engineering (Computer Engineering).

Guide	Head of Department	Principal
Prof. Siddharth Hariharan	**Dr. Kishore Sakhure**	**Dr. L. K. Ragha**

Project Report Approval

This Mini Project 2 B Report – entitled "**PredictWell:** Chronic Disease Predictor" by following students is approved for the degree of ***T.E. in "Computer Engineering"***.

Submitted by:

1) Manav Rajpurohit ID No: TU3F2122034

2) Prateek Arote ID No: TU3F2122043

3) Prasad Shinde ID No: TU3F2122047

Examiners Name & Signature:

1.----------

2.----------

Date: ----------

Place: ----------

Declaration

We declare that this written submission represents our ideas in our own words and where others' ideas or words have been included, we have adequately cited and referenced the original sources. We also declare that we have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in our submission. We understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

Manav Rajpurohit	**ID No: TU3F2122034**	----------
Prateek Arote	**ID No: TU3F2122043**	----------
Prasad Shinde	**ID No: TU3F2122047**	----------

Date: _____

Place: _____

Acknowledgement

We would like to express our sincere gratitude towards our guide **Prof. Siddharth Hariharan,** Mini Project Coordinators **Prof. Siddharth Hariharan** for their help, guidance and encouragement, they provided during the project development. This work would have not been possible without their valuable time, patience and motivation. We thank them for making our stint thoroughly pleasant and enriching. It was great learning and an honor being their student.

We are deeply thankful to **Dr. Kishor Sakhure (H.O.D Computer Department)** and entire team in the Computer Department. They supported us with scientific guidance, advice and encouragement, they were always helpful and enthusiastic and this inspired us in our work.

We take the privilege to express our sincere thanks to **Dr. L. K. Ragha** our Principal

Manav Rajpurohit	**ID No: TU3F2122034**	----------
Prateek Arote	**ID No: TU3F2122043**	----------
Prasad Shinde	**ID No: TU3F2122047**	----------

Date: _____

Place: _____

Index

Abstract

Chronic diseases represent a significant global health challenge, imposing immense burdens on individuals, healthcare systems, and society as a whole. **Early detection** and prevention of these conditions are crucial for reducing **morbidity and mortality rates**. In this project, we present a **predictive model** designed to identify individuals at **risk of developing chronic diseases** based on **demographic information, lifestyle factors, medical history, and genetic predispositions**. Through the utilization of **machine learning techniques**, including feature selection, model training, and evaluation, we aim to construct an accurate and reliable predictor capable of assessing an individual's susceptibility to various chronic conditions. Ethical considerations, such as **data privacy** and fairness, are paramount throughout the project lifecycle. By providing an efficient tool for early risk assessment and intervention, this project seeks to contribute to the advancement of personalized medicine and public health initiatives, ultimately enhancing healthcare outcomes and quality of life for individuals worldwide.

List of Figures

List of Tables

Chapter 3

Introduction

1) **Motivation**:

It's important because it can help doctors catch diseases early when they're easier to treat, potentially saving lives and reducing the need for expensive treatments. By looking at lots of information about a person, like their genes, lifestyle, and medical history, the computer can learn to recognize patterns that might indicate a risk for certain diseases

1.1 **Scope of the project:**

The scope of the disease prediction project using machine learning is vast and promising. By leveraging advanced algorithms and large datasets, this project aims to revolutionize healthcare by predicting diseases before symptoms appear, The project's scope includes integrating diverse sources of data such as genetic information, lifestyle factors, and medical history to develop accurate prediction models

1.2 **Need Of the Problem:**

The need for disease prediction using machine learning stems from several critical factors in healthcare project .Health care industry generate tera bytes of data by analyzing of voluminous of data we can predict occurrence of disease. Main purpose of this system is to predict disease and controlling the disease by providing accurate and trustworthy disease risk prediction. Early detection of diseases is often key to successful treatment and management, leading to better patient outcomes and reduced healthcare costs.

Chapter 3
Problem Statement

1. **Problem statement:** Many people suffer from chronic diseases like cancer, diabetes, and heart problems, which can have serious consequences. Detecting these diseases early can help prevent complications. Therefore, there is a need to develop an advanced chronic disease prediction framework that utilizes machine learning techniques to provide accurate and personalized risk assessments for diseases such as cancer, diabetes, and heart problems. Early detection of diseases is often key to successful treatment and management, leading to better patient outcomes and reduced healthcare costs

2. **Features:** This system caters to multiple users, including healthcare providers, patients, researchers, and administrators. Healthcare providers utilize the system to assess disease risks, plan interventions, and recommend treatments, while patients receive personalized risk assessments and health recommendations. The disease prediction system becomes accurate by analyzing large amounts of diverse health data, such as medical history, lifestyle, genetics, and environmental factors. Using smart algorithms, it finds patterns and connections in this data to make predictions about a person's health.

3. **Objectives:** Our aim is to provide quick medical diagnosis to the patients.
   - Predict various chronic diseases based on demographic, lifestyle, and medical data.
   - Enhance early detection and intervention strategies.
   - Create a user-friendly interface for input and result visualization.

4. **Specifications of the system:**
   - Processor: A modern multi-core processor (i3 or equivalent)
   - RAM: Min. 4 GB
   - Programming Language: Python 3.12.0
   - Development Environment: Jupyter Notebook

Chapter 3

Literature Review

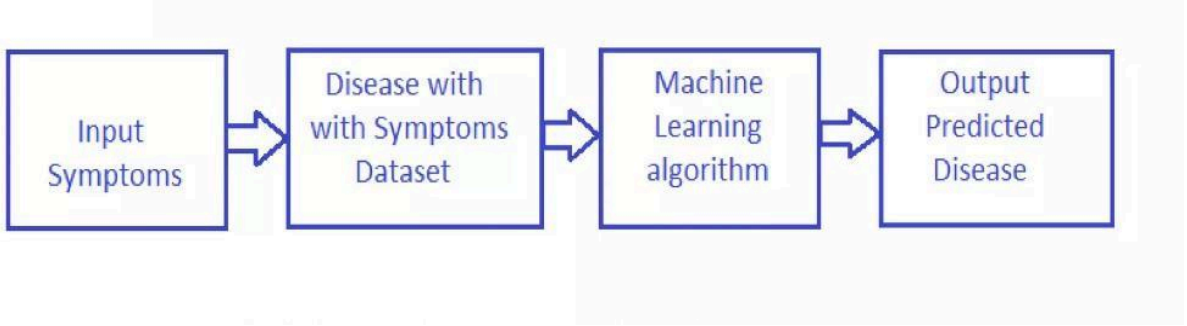
Paper Name	Author Name	Year	Summary
Disease Prediction by Machine Learning Over Big Data From Healthcare Communities	Min Chen Yixue Hao, Kai Hwang	2022	Developed a system which is based on patients data which is used for the prediction of multiple diseases. The sample size selected is less which affected the accuracy of the work. Method applied : Naïve bayes, Support Vector Machine

Paper Name	Author Name	Year	Summary
Identification and Prediction of Chronic Diseases Using Machine Learning Approach	Rayan Alanazi College of Science and Arts in Qurayyat, Jouf University, Sakakah	2022	This section describes the related works that are performed in developing the proposed model for predicting chronic diseases. The following are the discussions made by reviewing the existing literature that helps develop the proposed system efficiently and effectively. Method applied : Naïve bayes, Convolutional Neural Network

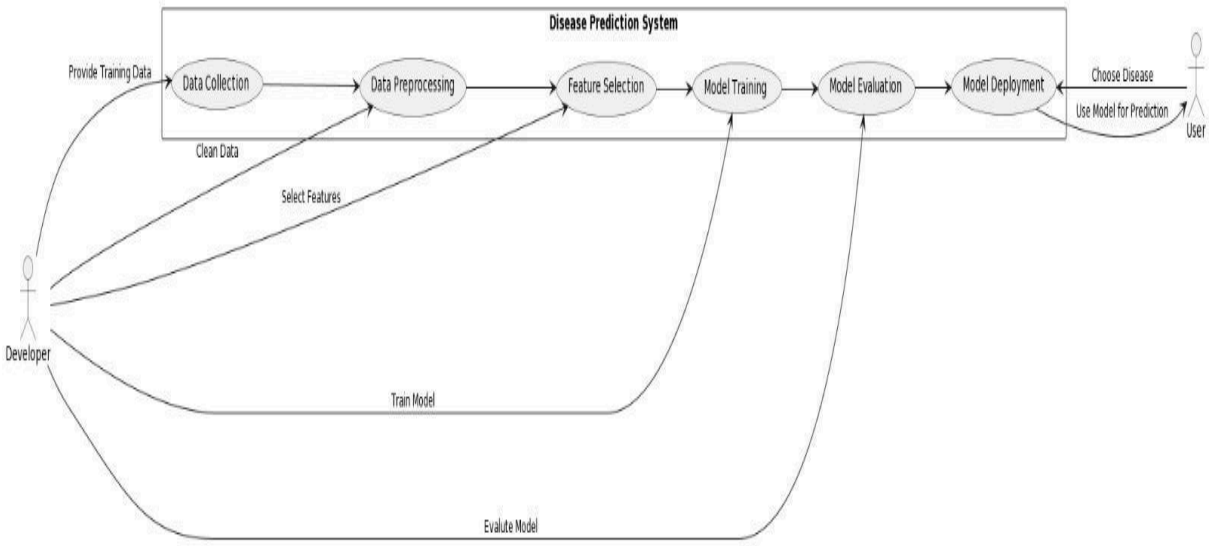
Paper Name	Author Name	Year	Summary
Prediction of Chronic diseases using Machine Learning Techniques. International journal of management, technology	Siddegowda C. J., & A. Jayanthila Devi.	2022	The research finds are very useful as they help health professionals. The limitations of various machine learning algorithms are also studied. Accuracy is about 85%

Paper Name	Author Name	Year	Summary
A Literature Review of Predicting Cancer Disease	Mr.A.Deivendran1 , Ms.K.Yemuna Rane M.Sc., M.Phil2 ., M.Phil Research Scholar, Dept of Computer Science, Kongunadu Arts and Science college, Coimbatore-29	2019	Prediction of chronic diseases has gained prominence among the research community as it opens the lots of scope for research and development. Parameter of selection is not elaborate. Method applied : Naïve bayes, Decision Trees, Artificial neural networks and their Multilayer Perception model

Chapter 4
Proposed system



4.1 Flowchart



4.2 UML Diagram

Code:

```
import streamlit as st
import pandas as pd
import tensorflow as tf
from tensorflow import keras
import numpy as np
from sklearn.preprocessing import StandardScaler
from sklearn.model_selection import train_test_split
import pickle

@st.cache_resource
def load_models():
    with open('models/RandomForest_Kidney_model.pkl', 'rb') as file:
        kidney_model = pickle.load(file)

    with open('models/Decision_tree_diabetes_model.pkl', 'rb') as file:
        diabetes_model = pickle.load(file)

    heart_model = keras.models.load_model('models/heart_disease_model.hdf5')

    return kidney_model, diabetes_model, heart_model

kidney_model, diabetes_model, heart_model = load_models()


# Streamlit app
def main():
    st.title('PredictWell: Chronic Disease Predictor')

    # Sidebar navigation
    st.sidebar.title('PredictWell')
    selected_disease = st.sidebar.radio('Select Disease:', ['Kidney Disease', 'Heart Disease',
'Diabetes'])

    if selected_disease == 'Kidney Disease':
        st.subheader('Kidney Disease Prediction')
        # Add input fields and prediction logic for kidney disease prediction
        # Age input
        kd_age = st.slider("Age", min_value=0, max_value=100, value=30)

        gender = st.selectbox("Gender", ('Male','Female'))

        # Blood pressure input
        kd_bp = st.slider("Blood Pressure", min_value=50, max_value=180, value=70)

        # Albumin input
        kd_al = st.slider("Albumin", min_value=0, max_value=5, value=0)
```

```
    # Sugar input
    kd_su = st.slider("Sugar", min_value=0, max_value=5, value=0)

    # Bacteria input
    ba_options = ['Not Present', 'Present']
    k_ba = st.selectbox("Bacteria", ba_options)

    if k_ba == 'Not Present':
        kd_ba = 0
    elif k_ba == 'Present':
        kd_ba = 1


    # Blood glucose random input
    kd_bgr = st.slider("Blood Glucose", min_value=22.0, max_value=500.0, value=145.0)

    # Blood urea input
    kd_bu = st.slider("Blood Urea", min_value=0, max_value=400, value=56)

    # Serum creatinine input
    kd_sc = st.slider("Serum Creatinine", min_value=0.0, max_value=16.0, value=2.997)

    # Sodium input
    kd_sod = st.slider("Sodium", min_value=80, max_value=180, value=135)

    # Potassium input
    kd_pot = st.slider("Potassium", min_value=0, max_value=60, value=4)

    # Hemoglobin input
    kd_hemo = st.slider("Hemoglobin", min_value=0.0, max_value=25.0, value=12.5)

    # Packed cell volume input
    kd_pcv = st.slider("Packed Cell Volume", min_value=0.0, max_value=70.0,
value=29.8)

    # White blood cell count input
    kd_wc = st.slider("White Blood Cell Count", min_value=3000, max_value=40000,
value=6500)

    # Red blood cell count input
    kd_rc = st.slider("Red Blood Cell Count", min_value=0.0, max_value=10.0, value=4.5)

    # Hypertension input
    htn_options = ['No', 'Yes']
    k_htn = st.selectbox("Hypertension", htn_options)

    if k_htn == 'No':
        kd_htn = 0
    elif k_htn == 'Yes':
        kd_htn = 1
```

```
        # Diabetes mellitus input
        dm_options = ['No', 'Yes']
        k_dm = st.selectbox("Diabetes Mellitus", dm_options)

        if k_dm == 'No':
            kd_dm = 0
        elif k_dm == 'Yes':
            kd_dm = 1

        #Coronary Artery Disease input
        cad_options = ['No', 'Yes']
        k_cad = st.selectbox("Coronary Artery Disease", cad_options)

        if k_cad == 'No':
            kd_cad = 0
        elif k_cad == 'Yes':
            kd_cad = 1

        appet_options = ['Good', 'Poor']
        k_appet = st.selectbox('Appetite:', appet_options)

        if k_appet == 'Poor':
            kd_appet = 0
        elif k_appet == 'Good':
            kd_appet = 1

        def
predict_kidney(kd_age,kd_bp,kd_al,kd_su,kd_ba,kd_bgr,kd_bu,kd_sc,kd_sod,kd_pot,kd_he
mo,kd_pcv,kd_wc,kd_rc,kd_htn,kd_dm,kd_cad,kd_appet):
            input_data =
np.array([[kd_age,kd_bp,kd_al,kd_su,kd_ba,kd_bgr,kd_bu,kd_sc,kd_sod,kd_pot,kd_hemo,k
d_pcv,kd_wc,kd_rc,kd_htn,kd_dm,kd_cad,kd_appet]])
            data_2d = input_data.reshape(1, -1)
            kd_prediction = kidney_model.predict(data_2d)
            return kd_prediction

        if st.button("Predict"):
            kd_prediction =
predict_kidney(kd_age,kd_bp,kd_al,kd_su,kd_ba,kd_bgr,kd_bu,kd_sc,kd_sod,kd_pot,kd_he
mo,kd_pcv,kd_wc,kd_rc,kd_htn,kd_dm,kd_cad,kd_appet)
            print(kd_prediction)
            if kd_prediction == 0:
                st.write("Based on the input data, it is likely that you do not have Kidney
Disease.")
            elif kd_prediction == 1:
                st.write("Based on the input data, it is likely that you have Kidney Disease.")



    elif selected_disease == 'Heart Disease':
        st.subheader('Heart Disease Prediction')
```

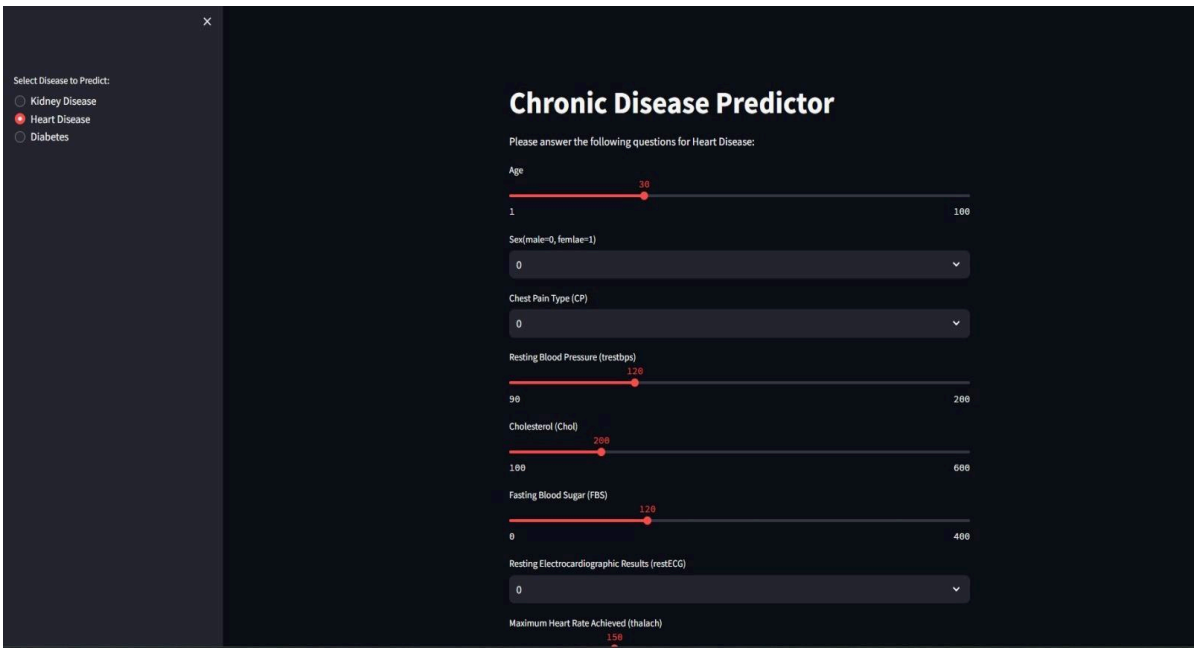
```
   #Heart diaseases prediction Starts
      # Add questions specific to Heart Disease
      age = st.slider("Age", min_value=1, max_value=100, value=30)
      gender = st.selectbox("Gender", ('Male','Female'))

      if gender == 'Male':
         sex = 0
      elif gender == 'Female':
         sex = 1

      cp = st.selectbox("Chest Pain Type (CP)", [0, 1, 2, 3])
      trestbps = st.slider("Resting Blood Pressure (trestbps)", min_value=90,
max_value=200, value=120)
      chol = st.slider("Cholesterol (Chol)", min_value=100, max_value=600, value=200)
      fbs_value = st.slider("Fasting Blood Sugar (FBS)", min_value=0, max_value=400,
value=120)

      if fbs_value < 120:
         fbs = 0
      else:
         fbs = 1

      restecg = st.selectbox("Resting Electrocardiographic Results (restECG)", [0, 1, 2])
      thalach = st.slider("Maximum Heart Rate Achieved (thalach)", min_value=70,
max_value=420, value=150)
      exang = st.selectbox("Exercise-Induced Angina (exang)", [0, 1])
      oldpeak = st.slider("ST Depression (oldpeak)", min_value=0.0, max_value=6.2,
value=1.0)
      slope = st.selectbox("Slope of the Peak Exercise ST Segment (slope)", [0, 1, 2])
      ca = st.selectbox("Number of Major Vessels Colored by Fluoroscopy (Ca)", [0, 1, 2, 3])
      thal = st.selectbox("Thallium Stress Test (thal)", [0, 1, 2, 3, 4, 5, 6, 7])
      # Add more questions as needed


      df = pd.read_csv("Heart_Disease_Prediction.csv")

      x = df.iloc[:, :-1]  # Select all columns except the last one (features)
      y = df.iloc[:, -1]  # Select the last column (target variable)
      x_train, x_test, y_train, y_test = train_test_split(x, y, random_state=0, test_size=0.35)

      def predict_heart_disease(age, sex, cp, trestbps, chol, fbs, restecg, thalach, exang,
oldpeak, slope, ca, thal):
         input_data = np.array([[age, sex, cp, trestbps, chol, fbs, restecg, thalach, exang,
oldpeak, slope, ca, thal]])
         sc_x = StandardScaler()
         sc_x.fit(x_train)
         user_input_scaled = sc_x.transform(input_data)
         prediction = heart_model.predict(user_input_scaled)
         return prediction

      if st.button("Predict"):
```

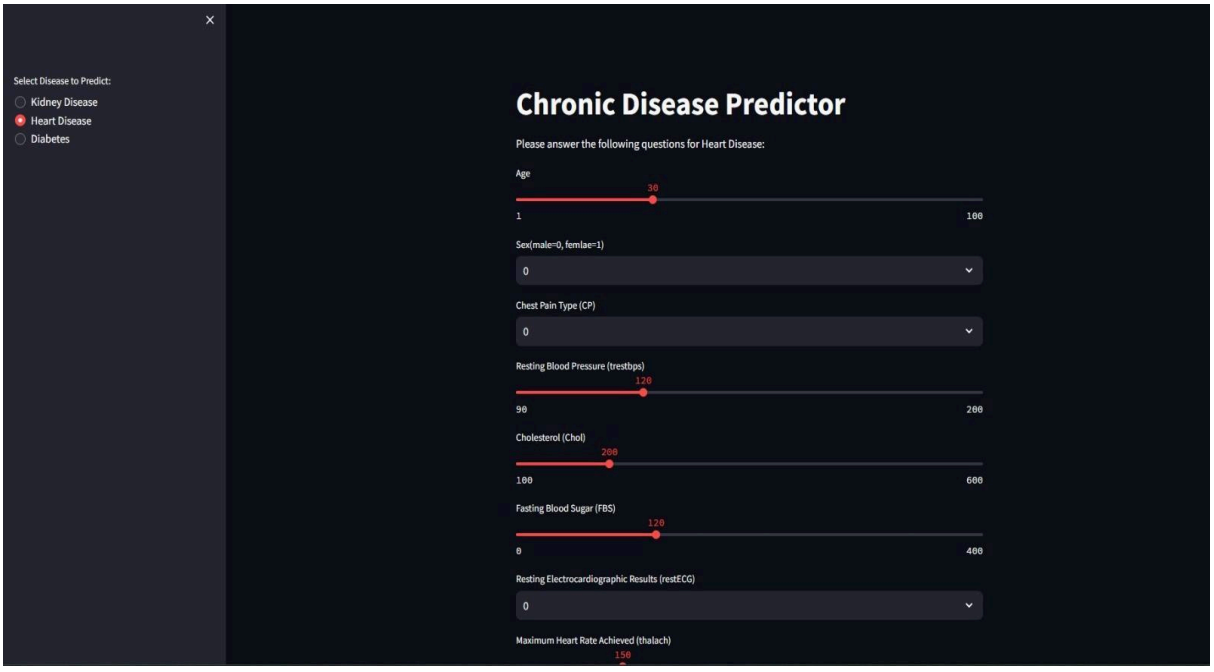
```
            prediction = predict_heart_disease(age, sex, cp, trestbps, chol, fbs, restecg, thalach,
exang, oldpeak, slope, ca, thal)
            print(prediction)
            if prediction >= 0.5:
                st.write("Based on the input data, it is likely that you have heart disease.")
            else:
                st.write("Based on the input data, it is likely that you do not have heart disease.")

    #Heart diaseases prediction Ends

    elif selected_disease == 'Diabetes':
        st.subheader('Diabetes Prediction')
        # Add input fields and prediction logic for diabetes prediction
        db_gender = st.selectbox("Gender", ('Male','Female','Other'))

        if db_gender == 'Male':
            db_sex = 0
        elif db_gender == 'Female':
            db_sex = 1
        elif db_gender == 'Other':
            db_sex = 2

        db_age = st.slider("Age", min_value=1, max_value=100, value=30)
        db_hyper = st.selectbox("Do you have Hypertension?", ('Yes','No'))

        if db_hyper == 'No':
            db_ht = 0
        elif db_hyper == 'Yes':
            db_ht = 1

        db_heart_diseases = st.selectbox("Do you have any Heart Diseases?", ('Yes','No'))

        if db_heart_diseases == 'No':
            db_hd = 0
        elif db_heart_diseases == 'Yes':
            db_hd = 1

        db_smoke = st.selectbox("Smoking Habits",
('Daily','Often','Occasionally','Quited','Never','Other'))

        if db_smoke == 'Daily':
            db_sh = 0
        elif db_smoke == 'Often':
            db_sh = 1
        elif db_smoke == 'Occasionally':
            db_sh = 2
        elif db_smoke == 'Quited':
            db_sh = 3
        elif db_smoke == 'Never':
            db_sh = 4
        elif db_smoke == 'Other':
```

```
            db_sh = 5

        db_bmi = st.slider("BMI(Body Mass Index)", min_value=0, max_value=100,
value=24)
        db_hgb = st.slider("Hemoglobin A1C (HbA1c) levels(in %)", min_value=0.0,
max_value=15.0, value=4.0)

        db_fbs = st.slider("Fasting Blood Sugar (FBS)", min_value=0, max_value=400,
value=120)

        def predict_diabetes(db_sex, db_age, db_ht, db_hd, db_sh, db_bmi, db_hgb, db_fbs):
            input_data = np.array([[db_sex, db_age, db_ht, db_hd, db_sh, db_bmi, db_hgb,
db_fbs]])
            data_2d = input_data.reshape(1, -1)
            db_prediction = diabetes_model.predict(data_2d)
            return db_prediction

        if st.button("Predict"):
            db_prediction = predict_diabetes(db_sex, db_age, db_ht, db_hd, db_sh, db_bmi,
db_hgb, db_fbs)
            print(db_prediction)
            if db_prediction == 0:
                st.write("Based on the input data, it is likely that you do not have Diabetes.")
            elif db_prediction == 1:
                st.write("Based on the input data, it is likely that you have Diabetes.")



if __name__ == '__main__':
    main()
```

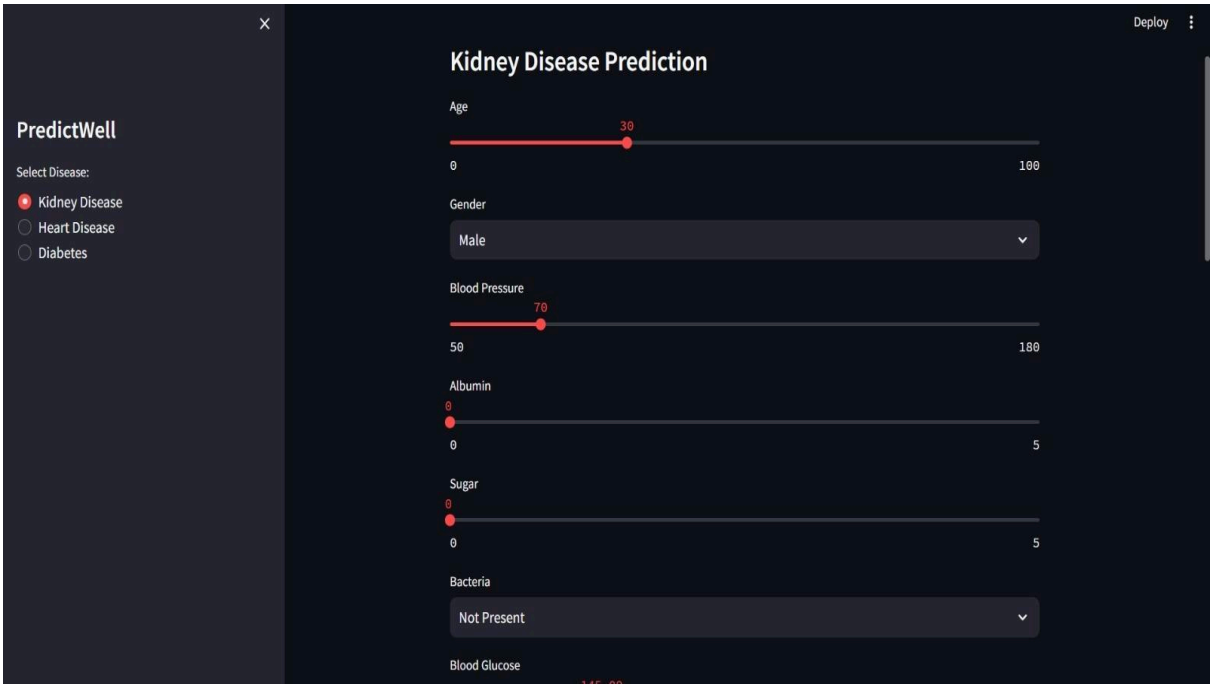
Chapter 5

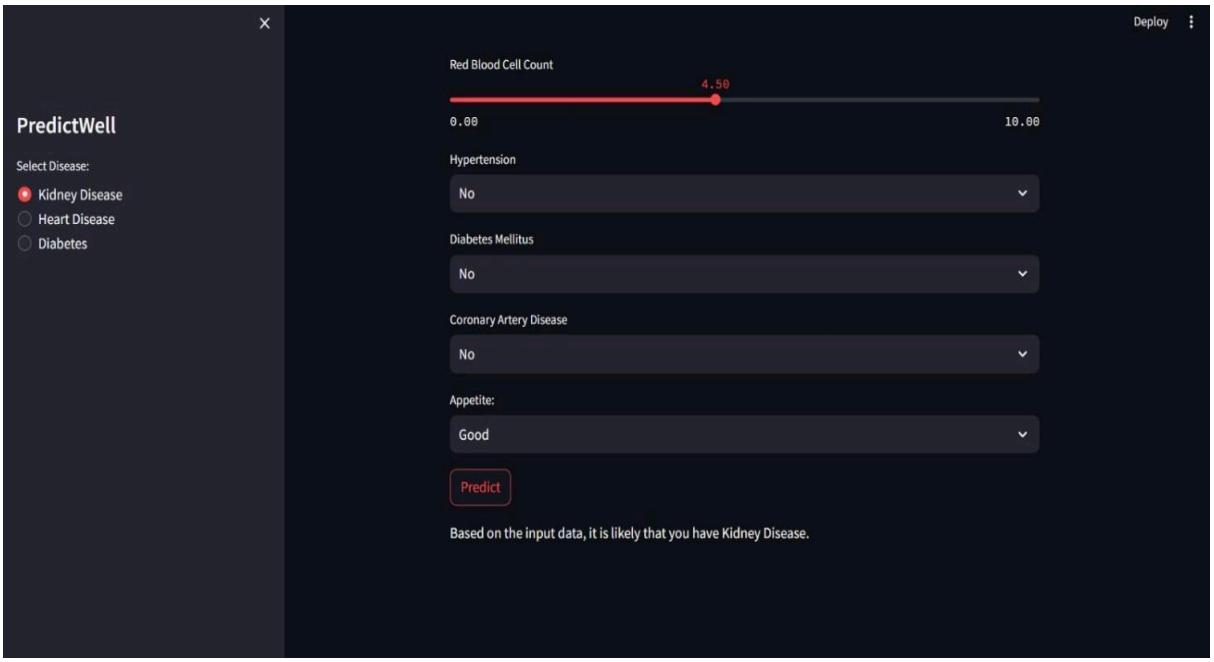
User interface



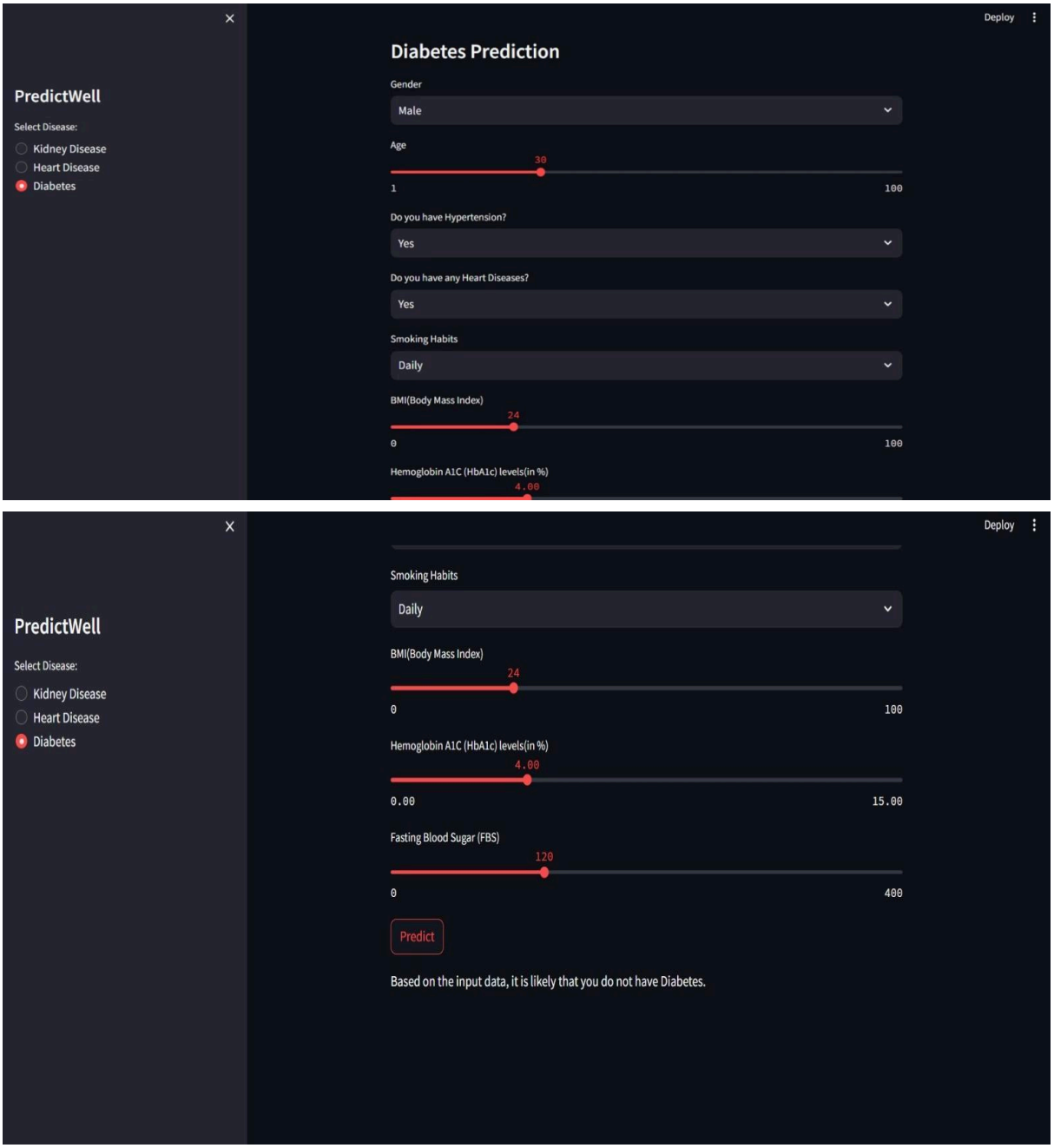


Heart Disease





Kidney Disease


PredictWell
Select Disease:
Kidney Disease
Heart Disease
Diabetes
Deploy
Diabetes Prediction
Gender
Male
Age
30
1
100
Do you have Hypertension?
Yes
Do you have any Heart Diseases?
Yes
Smoking Habits
Daily
BMI(Body Mass Index)
24
0
100
Hemoglobin A1C (HbA1c) levels(in %)
4.00
PredictWell
Select Disease:
Kidney Disease
Heart Disease
Diabetes
Deploy
Smoking Habits
Daily
BMI(Body Mass Index)
24
0
100
Hemoglobin A1C (HbA1c) levels(in %)
4.00
0.00
15.00
Fasting Blood Sugar (FBS)
120
0
400
Predict
Based on the input data, it is likely that you do not have Diabetes.

Diabetes

Chapter 6 Conclusion and Future Scope

Conclusion

This project proposed a method of identification and prediction of the presence of chronic disease in an individual using the machine learning algorithms by Random forest and SVM. The advantage of the proposed system is the use of both structured and unstructured data from real life for data set preparation, which lacks in many of the existing approaches. It is highly believed that the proposed system can reduce the risk of chronic diseases by diagnosing them earlier and also reduces the cost for diagnosis, treatment, and doctor consultation.

Future Scope

Future of disease prediction using machine learning is bright, with advancements expected to revolutionize healthcare delivery, improve patient outcomes, and enhance our understanding of complex diseases. In future work, the creation of more complex ML algorithms is much needed to increase the efficiency of disease prediction.

References

- Ambekar, S., & Phalnikar, R. (2018, August). Disease risk prediction by using convolutional neural network. In 2018 Fourth international conference on computing communication .An improved method for disease prediction using fuzzy approach. In 2015 www.event management.org

- Wadhwa, S., &Babber, K. (2020, July). Artificial intelligence in healthcare: predictive analysis on diabetes using machine learning algorithms. In International Conference on Computational Science and Its Application

- Disease Prediction by Machine Learning Over Big Data From Healthcare Communities

- A Literature Review of Predicting Cancer Disease by Rayan Alanazi College of Science and Arts Qurayyat, Jouf University, Sakakah.

- A Literature Review of Predicting Cancer Disease Mr.A.Deivendran1 , Ms.K.Yemuna Rane M.Sc., M.Phil2 ., M.Phil

- Prediction of Chronic diseases using Machine Learning Techniques. International journal of management, technology