# Directory: ai\_disease\_prediction/

# ---------- train\_model.py ----------

import pandas as pd

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

from sklearn.ensemble import RandomForestClassifier

from sklearn.metrics import accuracy\_score

import joblib

# Load dataset (you can replace this with your own patient data)

data = pd.read\_csv("data/patient\_data.csv")

# Preprocess

X = data.drop("disease", axis=1)

y = data["disease"]

X\_train, X\_test, y\_train, y\_test = train\_test\_split(X, y, test\_size=0.2, random\_state=42)

# Train model

model = RandomForestClassifier()

model.fit(X\_train, y\_train)

# Evaluate

predictions = model.predict(X\_test)

print("Accuracy:", accuracy\_score(y\_test, predictions))

# Save model

joblib.dump(model, "models/model.pkl")

# ---------- app.py ----------

import streamlit as st

import pandas as pd

import joblib

# Load model

model = joblib.load("models/model.pkl")

st.title("AI-Powered Disease Prediction")

# Input fields

def user\_input():

age = st.number\_input("Age", min\_value=0, max\_value=120, value=30)

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

bp = st.number\_input("Blood Pressure", min\_value=60, max\_value=200, value=120)

cholesterol = st.number\_input("Cholesterol", min\_value=100, max\_value=300, value=180)

glucose = st.number\_input("Glucose Level", min\_value=50, max\_value=300, value=100)

gender\_binary = 1 if gender == "Male" else 0

data = pd.DataFrame({

"age": [age],

"gender": [gender\_binary],

"blood\_pressure": [bp],

"cholesterol": [cholesterol],

"glucose": [glucose]

})

return data

input\_df = user\_input()

# Prediction

if st.button("Predict"):

prediction = model.predict(input\_df)

st.write("Predicted Disease:", prediction[0])

# ---------- data/patient\_data.csv ----------

# Sample structure:

# age,gender,blood\_pressure,cholesterol,glucose,disease

# 45,1,130,230,150,Heart Disease

# 29,0,120,180,90,Healthy

# 54,1,140,240,200,Diabetes

# ---------- requirements.txt ----------

streamlit

pandas

scikit-learn

joblib

# ---------- README.md ----------

# AI-Powered Disease Prediction

## Setup

```bash

pip install -r requirements.txt

```

## Train the Model

```bash

python train\_model.py

```

## Run the App

```bash

streamlit run app.py

```

## Notes

- Replace `data/patient\_data.csv` with your own data.

- Ensure the columns match expected inputs in `app.py` and `train\_model.py`.