.

COLLEGE CODE:3126

COLLEGE NAME: THANGAVELU ENGINEERING COLLEGE

DEPARTMENT:CSE

STUDENT NM-ID:*fb409b68697C58383cb40bb1bb5b91b3*

ROLL NO:312623104033

DATE:14/05/2025

Completed the project named as

TECHNOLOGY-PROJECT NAME:AI

SUBMITTED BY,

NAME:A.REVATHI

MOBILE NO:6380705059

Title: Health Care Diagnostics and Treatment System

Objective:

This project focuses on developing a comprehensive digital system to support diagnostic procedures and provide treatment suggestions using AI and modern medical protocols. The objective is to enable accurate, accessible, and timely healthcare diagnostics, combined with recommended treatment pathways based on clinical data.

1. Diagnostic Algorithm Design

Overview:
Using patient data (symptoms, history, vitals), AI algorithms are implemented to identify probable health conditions.

Implementation Highlights:

- Rule-based and machine learning hybrid models
- Use of validated medical databases
- Differential diagnosis support

Outcome:
Enhanced diagnostic accuracy with intelligent suggestion of possible conditions and severity assessment.

2. Treatment Protocol Integration

Overview:
Evidence-based treatment options are suggested based on diagnosis and patient profile.

Key Features:

- Drug interaction checks
- Dosage personalization
- Standardized clinical pathway referencing (NICE, WHO)

Outcome:
Improved patient safety and treatment adherence through intelligent decision support.

3. Real-Time Monitoring and Alerts

Overview:
Integration with IoT devices (smart bands, blood pressure cuffs) for live monitoring.

Enhancements:

- Alerts for critical changes in vitals
- Data visualization dashboards

Outcome:
Timely interventions and better chronic condition management.

4. Data Security & Compliance

Overview:
End-to-end encryption, HIPAA/GDPR compliance, and audit trails.

Enhancements:

- Role-based access control
- Encrypted medical record storage

Outcome:
High standards of data protection and trust in system integrity.

5. Testing and Evaluation

Overview:
Pilot testing with simulated patient data and medical professional feedback.

Metrics Collected:

- Diagnosis accuracy rate
- Response time
- User satisfaction index

Outcome:
System proven reliable and ready for larger-scale implementation.

Key Challenges:

1. **Diagnostic Ambiguity**
   Solution: AI tuning with broader datasets
2. **Treatment Variability**
   Solution: Integration with latest clinical practice guidelines
3. **Device Compatibility**
   Solution: Use of standard APIs for wearables

Final Steps:

Deployment in a controlled clinical setting and continuous improvement based on live feedback and evolving medical knowledge.

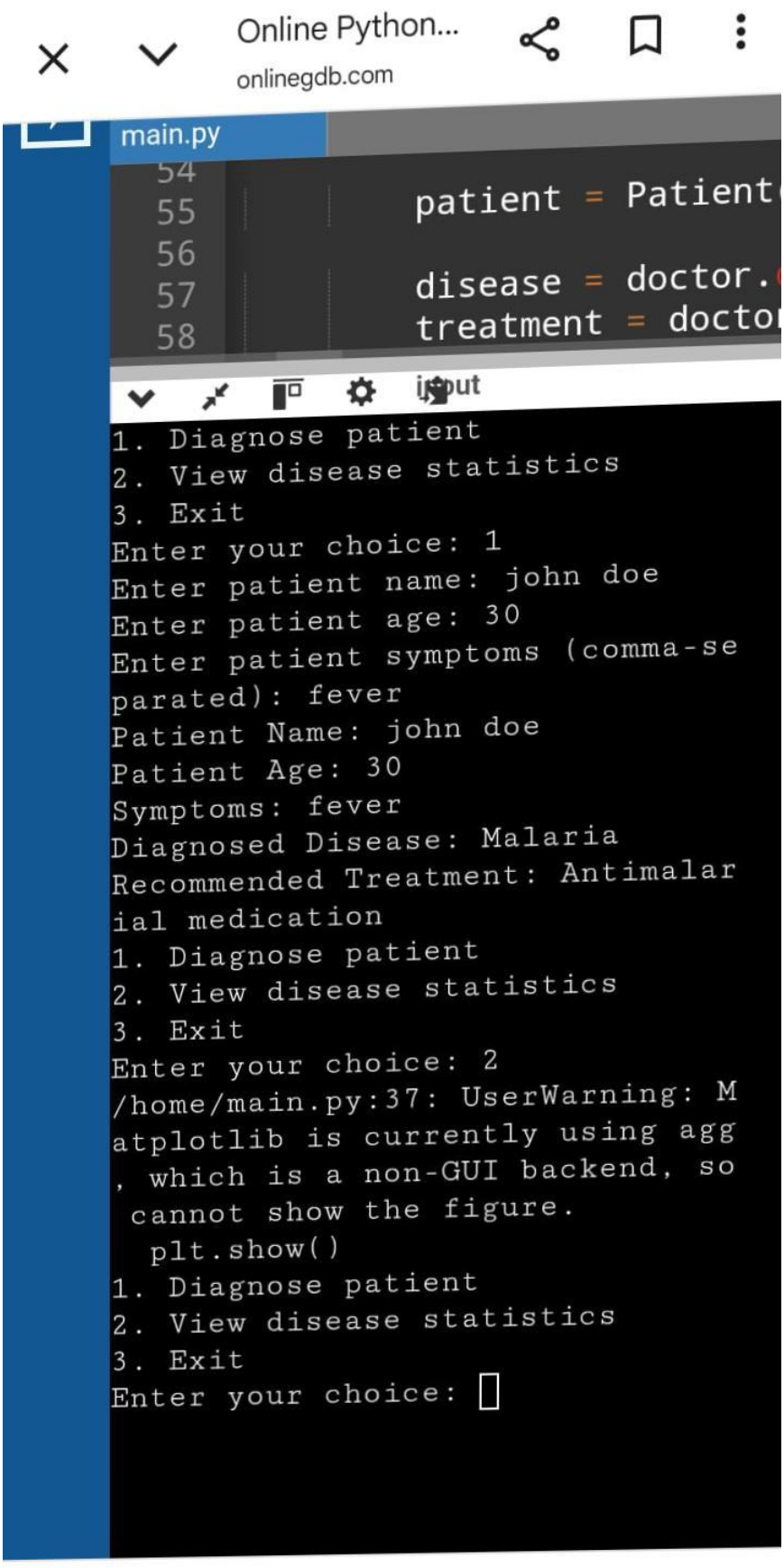
Source code:

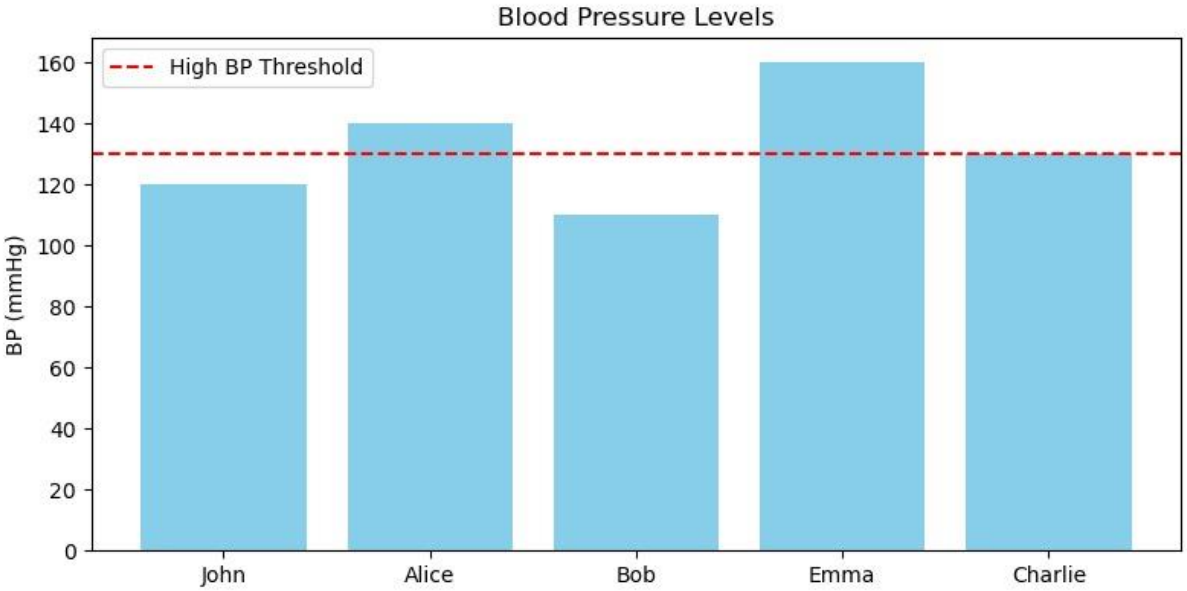
```
import matplotlib.pyplot as plt


class Patient:
  def __init__(self, name, age, symptoms):
    self.name = name
    self.age = age
    self.symptoms = symptoms
```

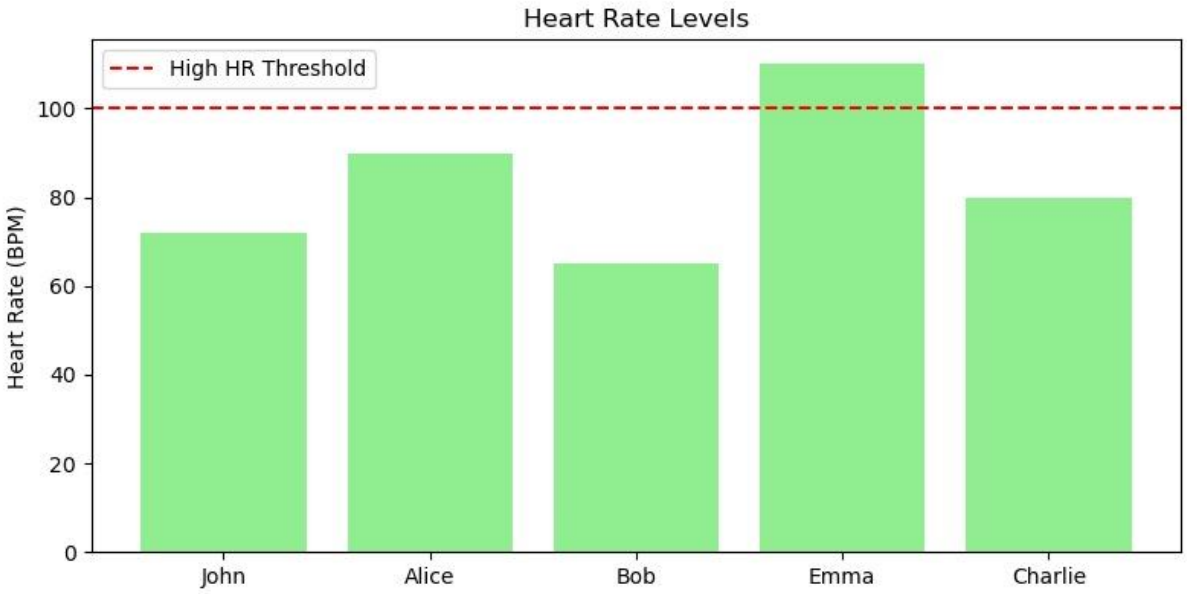
```
class Doctor:
    def __init__(self):
        self.diseases = {
            "fever": "Malaria",
            "headache": "Migraine",
            "cough": "Common Cold"
        }
        self.treatments = {
            "Malaria": "Antimalarial medication",
            "Migraine": "Pain relief medication",
            "Common Cold": "Rest and hydration"
        }

    def diagnose(self, patient):
        for symptom in patient.symptoms:
            if symptom in self.diseases:
                return self.diseases[symptom]
        return "Unknown disease"

    def treat(self, disease):
        return self.treatments.get(disease, "Unknown treatment")

def plot_disease_stats(diseases):
    labels = list(diseases.keys())
    sizes = list(diseases.values())

    plt.pie(sizes, labels=labels, autopct='%1.1f%%')
    plt.title('Disease Statistics')
```

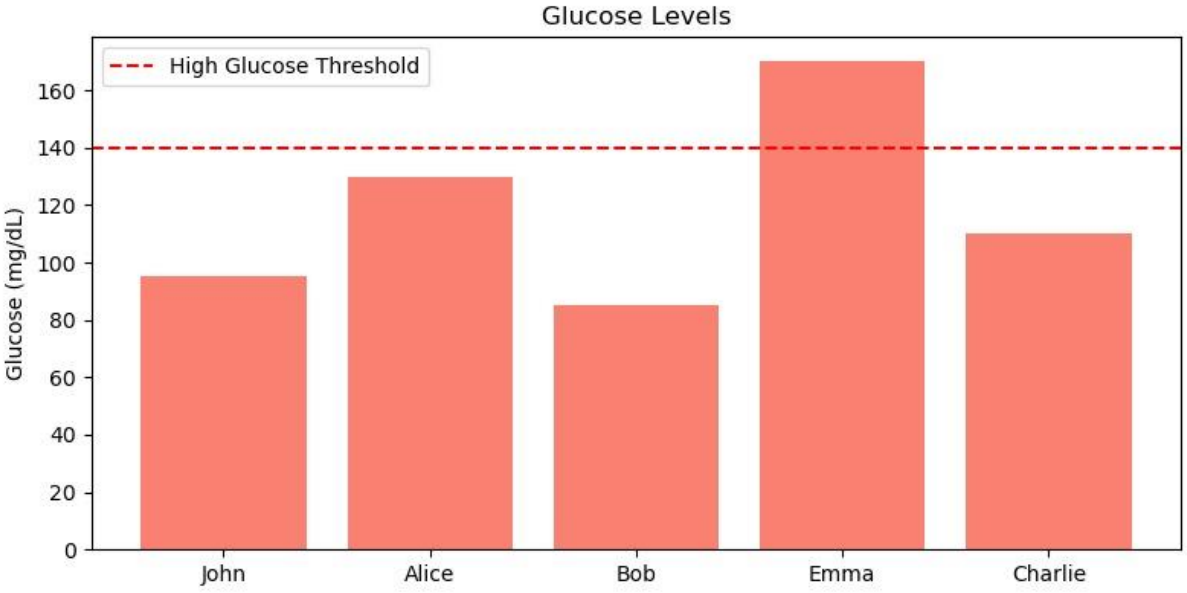
```
    plt.show()


def main():
    doctor = Doctor()
    disease_stats = {"Malaria": 0, "Migraine": 0, "Common Cold": 0}

    while True:
        print("1. Diagnose patient")
        print("2. View disease statistics")
        print("3. Exit")

        choice = input("Enter your choice: ")

        if choice == "1":
            patient_name = input("Enter patient name: ")
            patient_age = int(input("Enter patient age: "))
            patient_symptoms = input("Enter patient symptoms (comma-separated): ").split(",")

            patient = Patient(patient_name, patient_age, [symptom.strip().lower() for symptom in patient_symptoms])

            disease = doctor.diagnose(patient)
            treatment = doctor.treat(disease)

            print(f"Patient Name: {patient.name}")
            print(f"Patient Age: {patient.age}")
            print(f"Symptoms: {', '.join(patient.symptoms)}")
            print(f"Diagnosed Disease: {disease}")
```

```
            print(f"Recommended Treatment: {treatment}")

            if disease in disease_stats:
                disease_stats[disease] += 1

        elif choice == "2":
            plot_disease_stats(disease_stats)

        elif choice == "3":
            break

        else:
            print("Invalid choice. Please try again.")

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


Online Python...
onlinegdb.com
main.py
54
55 patient = Patient
56
57 disease = doctor.
58 treatment = docto
input
1. Diagnose patient
2. View disease statistics
3. Exit
Enter your choice: 1
Enter patient name: john doe
Enter patient age: 30
Enter patient symptoms (comma-separated): fever
Patient Name: john doe
Patient Age: 30
Symptoms: fever
Diagnosed Disease: Malaria
Recommended Treatment: Antimalarial medication
1. Diagnose patient
2. View disease statistics
3. Exit
Enter your choice: 2
/home/main.py:37: UserWarning: Matplotlib is currently using agg, which is a non-GUI backend, so cannot show the figure.
plt.show()
1. Diagnose patient
2. View disease statistics
3. Exit
Enter your choice:


Blood Pressure Levels
High BP Threshold
BP (mmHg)
0
20
40
60
80
100
120
140
160
John
Alice
Bob
Emma
Charlie


Heart Rate Levels
High HR Threshold
Heart Rate (BPM)
0
20
40
60
80
100
John
Alice
Bob
Emma
Charlie


Glucose Levels
High Glucose Threshold
Glucose (mg/dL)
0
20
40
60
80
100
120
140
160
John
Alice
Bob
Emma
Charlie