Electronic Health Record System

(EHRS-احرص)

Under supervision

Dr. Mohamed Abo Elzahhad

Dr. Safwat Mohamed

Eng. Khaled Mohamed

Web group:

- **Frontend Group**
  1) Amin Mamdouh Mohammed
  2) Islam Najih Abdel Rahim
  3) Israa Ashraf Muhammadin
- **Backend Group**
  4) Michael Majed William
  5) Nourhan Joseph Adel
  6) Yassa Tharwat Mikhail
  7) **Role of web group in project:**

1. Authentication Module

• Login Page: for Patients and Doctors

• Sign-Up Page: new user registration (choose role: Patient / Doctor)

Validation for email, password strength, and role-based redirection

2. Patient Module

• View Patient Data (profile, contact info, history)

• Access Tests and Prescriptions

• Track Chronic Diseases

• View Surgical Operations history

• Book an Appointment with a doctor or hospital

3. Doctor Module

• View Doctor Data (profile, specialization, contact info)

• Search for a Patient by ID

• View Patient Lists assigned to the doctor

Tools

Frontend: HTML, CSS, bootstrap , JS, react

Backend: SQL(DB), ASP.NET, C#, WEB API

Cloud and DevOps group & Hardware group

8) Mostafa Kadry Motawie
9) Nasser Tarek Nasser
10) Hisham Ahmed Abass
11) Mazen Mohamed Elbadry

Role of cloud & DevOps group in project:

We will deploy the entire project on AWS Cloud to ensure scalability, reliability, and security.

Components:

VPC (Virtual Private Cloud):

A custom VPC with public and private subnets to separate the FortiGate Firewall, frontend, backend, and database layers.

EC2 Instances:

Used to host our Kubernetes cluster (control plane and worker nodes).

RDS (Relational Database Service):

For hosting the main project database (MySQL/PostgreSQL). It will be run in a private subnet for enhanced security.

AWS CloudWatch:

for monitoring and observing the FortiGate and EC2 logs and metrics.

Kubernetes Cluster Deployment:

We will use Kubernetes (K8s) to manage and orchestrate our backend and front-end containers.

Setup Details:

Deployed on AWS EC2 using k3s (lightweight Kubernetes distribution).

Pods will be organized as:

Frontend Deployment: React or HTML/CSS web app container.

Backend Deployment: Python Flask or Node.js API container.

Database Connection: Private access only from backend Pods.

Configured Ingress Controller with HTTPS routing through FortiGate.

Containerization with Docker

Each part of the application (frontend, backend) will be containerized for easy deployment.

CI/CD Pipeline (DevOps Implementation)

We will automate the deployment process using DevOps tools and best practices.

Pipeline Stages:

1. Source Control: GitHub repository for the entire project.

2. Continuous Integration (CI):

Triggered on every code commit.

Runs automated tests and builds Docker images.

3. Continuous Deployment (CD):

Deploy the latest version to the Kubernetes cluster automatically.

Role of hardware group in project:

Sensor integration: Connect and calibrate body sensors (MAX30102 for heart rate & SpO_2, MLX90614 for temperature, and humidity sensors) with the ESP32-S microcontroller to ensure accurate real-time readings.

Data acquisition & signal processing: Implement stable analog/digital data acquisition from all sensors and filter noise using software or hardware techniques before transmission.

Display interface: Configure the OLED display (3 × 1.5 cm) to show live health metrics clearly, with optimized refresh rate and power consumption.

Power management: Design and integrate a Li-ion battery-based power circuit ensuring efficient charging, low power consumption, and protection from overcurrent/overcharge.

Hardware assembly & testing: Assemble all components (ESP32-S, sensors, display, power module) on a compact wearable board (~2 × 2.5 cm).

Security group

12) Mohamed Ahmed Mohamed Haridi
13) Marwa Ali Ibrahim

Role of security group in project:

Data protection: Ensure TLS/HTTPS for all traffic and protect data at rest (password hashing with bcrypt/argon2; AES encryption for sensitive fields.

Rate limiting: Implement and verify rate limits on sensitive endpoints (e.g., /login, /API) to mitigate brute-force and abuse.

IPS deployment: Deploy and tune an Intrusion Prevention System (Snort/Suricata) to monitor and block network-level attacks against our internal environment.

Penetration testing: Conduct full DAST/manual penetration tests after security controls are in place; deliver a vulnerability report and remediation guidance.

Cross-team collaboration: Work closely with Backend, Frontend, DevOps/Cloud and Network teams to apply fixes, validate configurations, and monitor alerts/logs

network design, management and monitoring

Firewall:

- Implementing FortiGate firewall

- applying port roles (LAN, WAN , DMZ)

- set the firewall to NAT mode

- configuring firewall profile policies

- enabling different inspection modes

- Antivirus and web filtering

AI group

14) Abanoub Ihab Anwar

Role of AI group in project:

Develop a Medical AI Chatbot that can interact with patients, understand their symptoms, and provide medically reasonable responses

Main Tasks:

1. Data Preparation

Collect and preprocess medical text data (symptoms, diseases, and common questions).

Use medical NLP datasets from trusted sources (e.g., PubMed, MIMIC).

2. Model Development

Fine-tune a pretrained medical language model (e.g., BioBERT, ClinicalBERT) to understand and respond to patients.

Add logic to generate safe, clear, and medically consistent answers.

3. Model Evaluation

Test the chatbot with different medical scenarios.

Check accuracy, clarity, and medical reliability of responses.

4. Web Integration

Build an API using FastAPI or Flask to connect the AI model with the web interface.

Mobile Application (Flutter) group

15) Hossam Al-Sayeh Mohamed
16) Khaled Zidan Mohamed
17) Hamdallah Muhammad Abdul Aziz

Role of application group in project:

The app's functionalities will mirror the website, ensuring both platforms share the same data and structure. Flutter was chosen for its cross-platform capabilities, allowing the app to run efficiently on both Android and iOS with a single codebase, while maintaining a modern and responsive design.