TKR HACK CONQUEST 1.0

Hackathon Solution Submission Template

1. Problem Overview

- **Project Title:** Digital Medico
- **Domain & Sub-Domain:** self inovation
- **Problem Statement:**
-
    - Lack of time during critical operation in medical domain?

How to induce technology into medical domain?

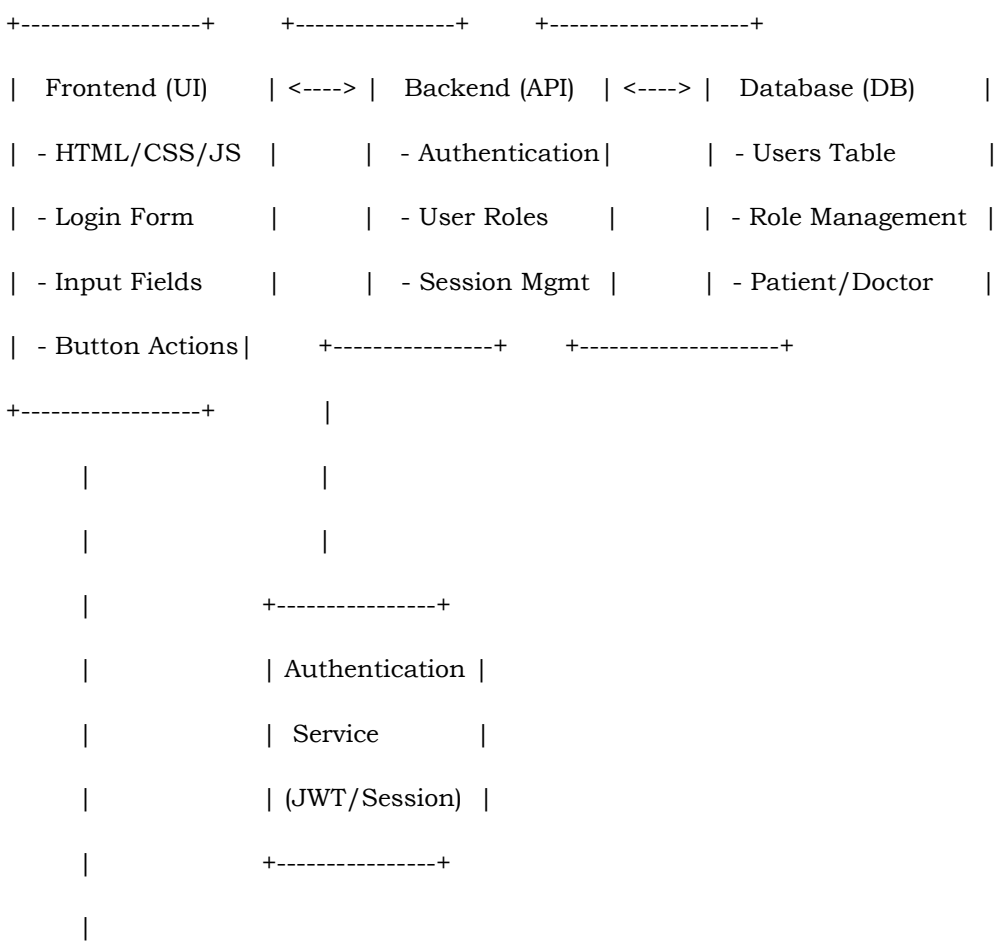
- **Objectives:** to connect tech platform into medical field
- **Target Audience:** doctors and patient

- **2. Solution Design & Implementation**

- **Technology Stack: html, css (frontend nodejs (backend)**

Architecture overview :

High-Level System Diagram



User Input:

- Patient Name
- Password
- Role (Patient/Doctor)

Core Features of the "Digital Medico" System:

1. **User Authentication (Login and Role-based Access)**:
    - Patients and doctors have unique login credentials. The system authenticates users by verifying their username, password, and role.
    - Ensures secure login by hashing passwords and managing sessions or using JWT tokens.
2. **Role-based Dashboard**:
    - After successful authentication, the system provides a different dashboard for **patients** and **doctors** based on their roles.
    - Patients can access their medical history, schedule appointments, and view lab results, while doctors can view their appointments, patient history, and manage prescriptions.
3. **Session Management**:
    - The system maintains user sessions using cookies, JWT tokens, or other methods.
    - Sessions are used to persist the user's login state across multiple requests and ensure that only authenticated users can access protected routes.
4. **Password Recovery and Reset**:
    - Users who forget their password can reset it through an email-based password recovery process.
    - A secure password reset token is generated and sent to the user’s email to allow them to reset their password.
5. **Secure Database Access**:
    - User data (including sensitive information like passwords) is stored securely in the database with encryption and hashing.
    - All user interactions with the system are logged, and actions are tracked for security.

3. Testing & Validation

Testing and Validation for the "Digital Medico" System:
Testing and validation are crucial steps to ensure the system performs as expected, is secure, and meets the requirements of both users (patients and doctors) and the organization. Testing can be broken down into different types, and each should be executed to verify that the system is functioning correctly, securely, and efficiently

Unit Testing:

- Focuses on testing individual units or components of the system (e.g., authentication, password hashing, JWT token generation).
- It ensures that each part of the code works as expected in isolation.

Example: Test the login logic with various inputs (correct/incorrect username/password).

4. Deployment & User Guide

Deployment Guide for "Digital Medico" System:
Deploying the **Digital Medico** system involves setting up the application on a production server and ensuring that all components (backend, frontend, database, etc.) work together in the live environment. Below is a step-by-step guide for deploying the system:

Prerequisites for Deployment
Before you begin, ensure you have the following:

- **Server/Hosting Platform**: A cloud server such as **AWS EC2**, **DigitalOcean**, or **Heroku**.
- **Node.js** and **npm** installed on the server (for backend).
- **Database**: A relational database like **MySQL** or **PostgreSQL**, or a NoSQL database (MongoDB, for example).
- **Frontend Assets**: HTML, CSS, and JavaScript files.
- **Git** installed on the server (for code version control).
- **Domain Name**: A domain name, if you wish to host the app on a custom URL.
- **SSL Certificate**: For secure HTTP (HTTPS), especially important for handling sensitive data like passwords and medical records.

6. Future Enhancements & Impact

- Store medical history you can use database (e.g.,MySQL).
- To implement a login system for doctors.
- Access of digital medico via biometric of patient. In order to keep the privacy of medical history of patient.

Conclusion

The **Digital Medico** system provides a secure, efficient, and user-friendly platform for both patients and doctors, addressing key needs in healthcare management. By implementing secure authentication, role-based access, and a smooth user experience, it ensures that users can easily access relevant information while safeguarding sensitive data. The system's deployment options, either on cloud servers like AWS or Heroku, make it scalable and adaptable for growing user bases. Ultimately, **Digital Medico** simplifies healthcare processes, improves accessibility, and enhances security, ensuring a more effective and streamlined healthcare experience.