Graduation Project Proposal Form

1. Project Information

- **Project Title:** Shefa
- **Course/Track:** FullStack.net
- **Team Members:**

1. Omar Mohammed Eldesawy Gomaa
2. Abdelrhman Waleed Mohammed Singer
3. Hussien Osman Mohammed Abdelrhman
4. Marwan El Sayed Al Doaif Mohammed
5. Ahmed Magdy Reda Sayed

2. Project Overview

❖ **Objective:**

- enhance accessibility and communication with clinics, ensuring a seamless and efficient interaction between patients and healthcare providers.
- prioritize patient care and health by providing a system for organizing medication schedules, offering reminders through notifications, and ensuring accurate dosage management.
- enhance patient health management by providing a clear medical history for each patient, facilitating effective communication with doctors, and streamlining the diagnostic process.
- provide accredited online medical consultations, offering patients convenient access to healthcare services during specific times of need.
- improve patient experience by recommending suitable doctors through targeted questions designed to understand the patient's health issue.

❖ **Scope of Work:**

- **User Interface Development:**
  Design and develop user-friendly interfaces tailored for patients, doctors, clinics and administrators.
  Provide direct communication channels within the system to facilitate interaction between patients and doctors.
- **Appointment and Medication Management:**
  Develop a system to organize medication schedules with automatic

notifications to remind patients.
Ensure accuracy in timing and dosage appointments for each medication.

- **Medical History Recording and Documentation:**
  Build a well-structured database for each patient containing their complete medical history.
  Provide secure and fast access for doctors to patient records to support accurate diagnosis.

- **Online Medical Consultations:**
  Develop a platform for accredited online medical consultations with suitable appointment scheduling.
  Provide tools for follow-up and electronic delivery of medical recommendations.

- **Doctor Recommendation System:**
  Design an intelligent questionnaire model that assists in recommending suitable doctors based on the patient's health condition.

❖ **Expected Outcomes:**

- increase accessibility to online consultations by simplifying appointment booking for patients and reducing waiting times.
- Facilitated diagnosis through clear and comprehensive medical records
- Improved accuracy in medication adherence and reduced dosage errors.
- Increased patient satisfaction with the healthcare services provided.
- Expanded access to online medical consultations

3.Problem Statement

Many patients face difficulties in adhering to medication schedules and maintaining effective communication with doctors, leading to delayed diagnoses and deterioration of their health conditions. This problem is exacerbated in crowded clinics lacking a centralized system to organize medication schedules and send automatic reminders, increasing the risk of medical errors such as missed or incorrect dosages. Furthermore, poor communication between patients and doctors negatively impacts the quality of healthcare and raises treatment costs due to worsening conditions. Therefore, there is a critical need for an integrated system that facilitates treatment adherence and enhances communication to improve patient outcomes.

4. Proposed Solution

- **Technologies Used:** React or angular (frontend) , ASB.Net (backend) , Microsoft SQL Server (database) , cloud hosting (AWS/Azure), Twilio/Local SMS provider, payment gateway(local).

- **System Architecture** :

Layered Architecture :

- **API:** This is the presentation or API layer. It's the entry point for the application, handling incoming requests and returning responses. In a C# backend, this is often a web API that exposes endpoints for clients to consume.
- **DAL:** This is the data access layer (DAL). Its sole responsibility is to interact with the database. It contains the code for performing CRUD (Create, Read, Update, Delete) operations, abstracting the data source from the business logic.
- **Entity:** This is the domain or entities layer. It holds the core business objectives and rules of the application. These entities are often simple Plain Old C# Objects (POCOs) that represent the real-world concepts the application deals with.

5. Resources Needed

- **Hardware/Software:**

Hardware:

- We will need Computer with good CPU and RAM
- Managed Database: A cloud service(AWS, Azure SQL Database) that handles database.

Software:

- Operating System: Windows.
- Programming Language & Frameworks: we will use ASP.NET Core for building web APIs.
- Database: A database management system (DBMS) like Microsoft SQL Server to store and manage data. You'll also need a tool like SQL Server Management Studio to interact with it.
- Development Tools: Visual Studio community , a version control system (Git)

6. Approval

- **Instructor/Advisor:** Islam Helmy
- **Signature:** .....