**DR.ASSISTANT**

***[Noman Vadsariya, 19K-1432& Section: F]***

***[M.Subhan Khan, 19K-1285& Section: F]***

***[Zohaib Rasool, 19K-0228 & Section: F]***

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**0. Acknowledgment**

Firstly, we would like to thank Sir Zain-Ul Hassan, Our Object-Oriented-Programming professor who taught us the course and introduced us to the concepts and principles of Object-Oriented-Programming that made it quite possible for us to implement those principles in our project. Secondly we would like to thank “Stack Overflow” which has answers for all your queries. Since we were working on C# and Window forms for the very first time it was nearly impossible to complete the project without any assistance. We mostly referred Stack Overflow for correcting syntax errors and for discovering C# libraries.

**1. Introduction**

Dr.Assistant is a user friendly (windows form desktop application) and sophisticated Health Care Management system (E-Clinic), a great need during this global pandemic. The project comprises of three major modules: doctor, patient and admin. The project will have email verified signups and protected logins. Dr.Assistant is designed to provide easy disease diagnosis (through disease prediction engine) and appointment booking; proper management of health reports, medical history and doctor prescriptions for patients. The system will notify doctors and patient about appointment details through email. Moreover doctors would be able to view the medical records of patients and would write prescription for patients. This project also makes drug management quite effortless, displaying data in the form of tables. Thus, Dr.Assistant will provide health care services on your finger tips.

1. **ADMIN ROLE:**

* Admin have access to pharmacy.
* Admin can perform CRUD operations with the drugs; one special feature is that everything is in the tabular format. Admin can directly update drug info by directly editing in the table.
* Admin can view every doctor’s basic info like; usernames and name and their specialty.
* For admin’s feasibility, doctor general information is also displayed in the text boxes by selecting usernames from the drop down list.

1. **PATIENT’S ROLE:**

* Firstly patient registers himself after signing up in the system; this process is completed after patient enters the verification code (otp) sent to his/her email.
* Patient has right to edit his/her info, but validations have been applied on the info.
* Going into the details of “Disease Prediction Engine”, Firstly it does not only give the differential diagnosis of the patient’s disease based on the symptoms selected by the patient from a drop down list, but also solves the problem of a patient to find a specialized doctor for that disease.
* Patient can also book an appointment with the specialist of his own choice after describing his symptoms.
* The system sends notifications to both doctor and patient about the details of the appointment after fixing the appointment.
* The patient’s medical records and prescription is properly managed and is displayed in tabular format for convenience.
* Patient can view and print the report of the appointment along with prescription.

1. **DOCTOR’S ROLE:**

* For doctors, the signing up and login process is same with same security level as that for and validations as that for patient.
* Doctor can view and edit his personal information, but validations have been applied.
* Doctor can write the prescription to for the patient whose appointment has been fixed and for any kind of mishap or lack of information about a patient, doctor would be notified with an email.
* Doctor can even go through the past medical reports of a patient in order to be completing confident while writing the prescription for the patient’s disease.
* Doctor has right to suggest drugs that would later be accepted or rejected by the admin, for adding into the inventory.

**2. Tools and Technologies Used:**

**Language:** C#

**IDE:** Microsoft Visual Studio,

**Databases:** Not used

**Libraries:**

**System Namespace:**

1. using System;

**Namespace for Windows based Applications:**

1. using System.Windows.Forms;

**For File I/O:**

1. using System.IO;

**For Sending Emails (Email verification and Appointment notifications)**

1. using System.Net;
2. using System.Net.Mail;
3. using System.Web;

**For the use of Background Worker in C#**

1. using System.Threading;

**For Text-to-Speech Conversion:**

1. using System.Speech.Synthesis;

**For Serialization and Object I/O:**

1. using System.Runtime.Serialization;
2. using System.Runtime.Serialization.Formatters.Binary;

**Framework:** .Net Framework 4

**3. Class diagram**

[The class diagrams and their brief description is in the attached pdf file]

**4. Link to source**

[A Google Drive or GitHub link to your code repository. The link must be working]

**5. Future work**

We started working on this project just after the acceptance of our project proposal, so we have incorporated maximum features that an E-Clinic system can have. Another feature we worked on was the use MS SQL in our project, but unfortunately we were not able to include the database in our project, so we are keen to learn about databases and to include it in this project.