



Smart Health

Submitted By

Mahmuda Akter Nipa

ID: 153-35-1385

Department of Software Engineering

Daffodil International University

Supervised By

Asif Khan Shakir

Lecturer

Department of Software Engineering

Faculty of Science and Information Technology

Daffodil International University

FALL 2019

This Project report has been submitted in fulfillment of the requirements for the Degree of Bachelor of Science in Software Engineering.

APPROVAL

This **Project** titled “**Smart Health**”, submitted by Mahmuda Akter Nipa, **ID No: 153-35-1385** to the Department of software engineering, Daffodil International University has been accepted as satisfactory for the partial fulfillments for the requirements of the degree of B.Sc. in Software Engineering and approved as its style and contents.

BOARD OF EXAMINERS

Chairman

Dr. Touhid Bhuiyan
Professor and Head
Department of Software Engineering
Faculty of Science and Information Technology
Daffodil International University

Internal Examiner 1

Dr. Md. Asraf Ali
Associate Professor
Department of Software Engineering
Faculty of Science and Information Technology
Daffodil International University

Internal Examiner2

Asif Khan Shakir
Lecturer
Department of Software Engineering
Faculty of Science and Information Technology
Daffodil International University

External Examiner

Dr. Md. Nasim Akhter
Professor
Department of Computer Science and Engineering
Faculty of Electrical and Electronic Engineering
Dhaka University of Engineering & Technology, Gazipur

PROJECT DECLARATION

The project entitled “**Smart Health**” is done under the supervision **Asif Khan Shakir, Lecturer**, Department of Software Engineering, Daffodil International University. I declare that this thesis is my original work for the degree of B.Sc.in Software Engineering and that neither the whole work nor any part has been submitted for another degree in this or any other university.

Submitted by



Mahmuda Akter Nipa

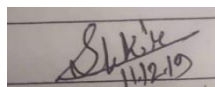
Id: 153-35-1385

Batch:18th

Department of Software Engineering

Daffodil International University

Certified by:



Asif Khan Shakir

Lecturer

Department of Software Engineering

Daffodil International University

Acknowledgement

I am thankful to my creator for giving me the opportunity to complete this research work and learn so much. I am thankful to my project supervisor, Asif Khan Shakir, for providing careful guidance starting from selecting the project scope to successfully finalizing the research work. I also thank Mahamuda rownok for her valuable observation, which was always insightful. Finally, I want like to express my gratitude to Professor Dr. Touhid Bhuiyan, head of the Software engineering faculty, for inspiring us in all means.

Abstract

The experience to working with project is very important for all health conscious people. This system will help Doctors and patients and It's make a smooth environment to ensure better medical service. This is a web based project where user show available doctor's list and make appointment to the doctor in online. So that user need not to go to doctor manually to make appointment and get serial number. User can easily make appointment and get serial number from their home using this website. User have to registration first and then can login and can see the user's profile, make appointment, serial no, patient's medical history and also available doctor's in that time. And admin can manage the whole system and provide serial number and time to the patient's. Also when doctor make prescription for the patient's it will be store in the patient's profile and doctor's dashboard.

Table of Contents

| | |
|--|------------|
| APROVAL..... | ii |
| PROJECT DECLARATION..... | iii |
| ACKNOWLEDGMENT | iv |
| ABSTRACT | v |
| TABLE OF CONTENT..... | vi |
| Chapter 1:INTRODUCTION..... | 1 |
| 1.1 Project Overview..... | 1 |
| 1.2 The Purpose of the project..... | 1 |
| 1.2.1 Background..... | 1 |
| 1.2.2 Benefits..... | 1 |
| 1.2.3 Goals..... | 2 |
| 1.3 Stockholder..... | 2 |
| 1.4 Proposed System Model(Block Diagram)..... | 3 |
| 1.5 Project Schedule..... | 4 |
| 1.5.1 Gant Chart..... | 4 |
| Chapter 2: SOFTWARE REQUIREMENT SPECIFICATION..... | 5 |
| 2.1 Requirement Specification..... | 5 |
| 2.2 Performance Requirement..... | 5 |
| 2.2.1. Speed and Latency Requirements..... | 5 |
| 2.2.2. Precisions or Accuracy Requirements..... | 6 |
| 2.3. Dependability Requirements..... | 6 |
| 2.3.1. Reliability Requirements..... | 6 |
| 2.3.2. Availability Requirements..... | 6 |
| 2.4. Maintainability and supportability Requirements..... | 7 |
| 2.4.1. Maintenance Requirements..... | 7 |
| 2.5. Security Requirements..... | 7 |
| 2.5.1. Access Requirements..... | 6 |
| 2.6. Usability and Human-Interaction Requirements..... | 7 |

| | |
|--|-------|
| 2.6.1. Ease of Use Requirements..... | 7 |
| 2.6.2 Understandability and Politeness Requirements..... | 7 |
| 2.6.3 Accessibility Requirements..... | 7 |
| 2.7. Legal Requirements..... | 8 |
| 2.7.1 Standards Requirements..... | 8 |
| Chapter 3: System Analysis..... | 9 |
| 3.1 Use Case Diagram..... | 9 |
| 3.2 Use Case Description..... | 09-19 |
| 3.2 Activity Diagram..... | 20 |
| 3.3.1 Activity daigram for Login..... | 21 |
| 3.3.2 Activity daigram update profile..... | 22 |
| 3.3.3Activity daigram Doctor search. Detail& Appointment:..... | 22 |
| 3.3.4 Activity daigram Schedule..... | 23 |
| 3.3.5 Activity daigram View serial & Update serial..... | 23 |
| 3.3.6 Activity daigram Prescription..... | 24 |
| 3.2.7 Activity daigram search patient..... | 24 |
| 3.4 Sequence Diagram..... | 25 |
| 3.4.1 Sequence Diagram for Login..... | 25 |
| 3.4.2 Sequence Diagram forU pdate profile..... | 25 |
| 3.4.3 Sequence Diagram for Schedule..... | 26 |
| 3.4.4 Sequence Diagram for Search doctor..... | 26 |
| 3.4.5 Sequence Diagram for Doctor Detail..... | 27 |
| 3.4.6 Sequence Diagram for Search patient..... | 27 |
| 3.4.7 Sequence Diagram for Appointment..... | 28 |
| 3.4.8 Sequence Diagram for Prescription..... | 28 |
| 3.4.9 Sequence Diagram for Serial..... | 29 |
| Chapter 4: SYSTEM DESIGN SPECIFICATION..... | 30 |
| 4.1 Entity Relationship Diagram..... | 30 |
| 4.2 Class Diagram..... | 31 |
| 4.3 Development of tools And Technology..... | 32 |
| 4.3.1 User Interface Technology..... | 32 |
| 4.3.2. Implementation Tools & Platforms..... | 32 |

| | |
|--|-------|
| Chapter 5: SYSTEM TESTING | 31 |
| 5.1. Testing Features | 31 |
| 5.1.1. Features to be tested | 31 |
| 5.1.2. Features not to be tested | 31 |
| 5.2. Testing Strategies | 31 |
| 5.2.1. Test Approach | 31 |
| 5.2.2. Pass/Fail Criteria | 31 |
| 5.2.3. Suspension and Resumption | 31 |
| 5.2.4. Testing Schedule | 32 |
| 5.3 Test case | 33 |
| 5.3.1 Test case 1 | 33 |
| 5.3.2. Test Case: 02 | 34-35 |
| CHAPTER 6: USER MANUAL | 36 |
| 6.1. User Manual (Patient/normal user) | 36 |
| 6.1.1. User login | 36 |
| 6.1.2. User Registration | 37 |
| 6.1.3. User Dashboard | 38 |
| 6.1.4. Patient appointment | 39 |
| 6.1.5 patient profile: | 40 |
| 6.1.6 patient check serial number. | 41 |
| 6.1.7 Doctor write prescription | 41 |
| CHAPTER 7 | 42 |
| 7.1. GitHub Link | 42 |
| 7.2. Limitations | 42 |
| 7.3. Obstacles & Achievements | 42 |
| 7.4. Future Scope | 42 |
| Conclusion | 43 |
| REFERENCES | 44 |

LIST OF FIGURES

| | |
|---|----|
| Figure 1.1: Block diagram (Programmer Forum) | 03 |
| Figure 3.1: Use Case diagram (Smart health)..... | 09 |
| Figure 3.2: Activity diagram for Login (Smart Health)..... | 20 |
| Figure 3.3: Activity diagram for update profile (Smart Health)..... | 21 |
| Figure 3.4: Activity diagram for doctor search, detail, appointment (Smart Health)..... | 21 |
| Figure 3.5: Activity diagram for schedule (Smart Health)..... | 22 |
| Figure 3.5: Activity diagram for view& update serial (Smart Health) | 22 |
| Figure 3.5: Activity diagram for prescription (Smart Health) | 23 |
| Figure 3.5: Activity diagram for search patient (Smart Health)..... | 23 |
| Figure 3.6: Sequence diagram (Smart Health)..... | 24 |
| Figure 3.7: Sequence diagram for login (Smart Health)..... | 24 |
| Figure 3.8: Sequence diagram for update profile (Smart Health)..... | 24 |
| Figure 3.9: Sequence diagram for schedule (Smart Health)..... | 25 |
| Figure 3.10: Sequence diagram for search doctor (Smart Health)..... | 25 |
| Figure 3.10: Sequence diagram for doctor detail (Smart Health)..... | 26 |
| Figure 3.10: Sequence diagram for search patient (Smart Health)..... | 26 |
| Figure 3.10: Sequence diagram for appointment (Smart Health)..... | 27 |
| Figure 3.10: Sequence diagram for prescription (Smart Health)..... | 27 |
| Figure 3.10: Sequence diagram for serial (Smart Health) | 28 |
| Figure 4.1: Entity Relationship diagram (Smart Health)..... | 29 |
| Figure 4.2: Class diagram (Smart Health)..... | 30 |

LIST OF TABLE

| | |
|---|----|
| Table 1.1: Gantt chart..... | 04 |
| Table 2.1: Functional Requirements..... | 05 |
| Table 3.1: Login..... | 09 |
| Table 3.2: Update Profile..... | 10 |
| Table 3.3: Appointment..... | 11 |
| Table 3.4: Schedule..... | 12 |
| Table 3.5: Search patient..... | 13 |
| Table 3.6:view Serial..... | 14 |
| Table 3.7: Update Serial..... | 15 |
| Table 3.8: prescription..... | 16 |
| Table 3.9 Search Doctor..... | 17 |
| Table 3.9: Doctor Details | 18 |
| Table 3 ..10 :Appointment | 19 |
| Table 5.1: Testing Schedule..... | 35 |
| Table 5.2 : Test case 1 | 37 |
| Table 5.3: Test case 2 | 38 |

Chapter 1

Introduction

1.1 Project Overview

Smart health is a website where user can easily find out available doctor in emergency time and make appointment at their home. The main purpose of this project are doctor appointment and schedule. Patient can take easily appointment and can update schedule. Patient's medical history with prescription will be store there and user can update his own profile.

1.1.1 Background

We are on the edge of technology and it helps us in every aspects of life. Health is a basic human needs, we can take care our health and medical service smoothly if there is virtual platform where doctors patients can communicate with each other and get chance to treat them without having difficulties. This program will help users in this particular aspects.

1.2 Benefits

This project will enhance the communication between doctors and patients by sharing a common platform.

The user will be-

User: using this website user can easily make appointment in online. So it will save time, reduce complexity, save money. And all history will be stored in the system. It will make our life easier than manual system.

Admin: Admin can manage the system and all medical information about old and new patient will be stored that can help to doctors prescribe.

1.3 Goals

Main goal of our project to ensure better healthy life for everyone. And create a common platform and friendly environment for doctor and patients to give them a better medical service.

1.4 Stockholder

There are four types of stockholders:

1. Admin
2. Doctor
3. Patient
4. Normal User

1.5 Block Diagram

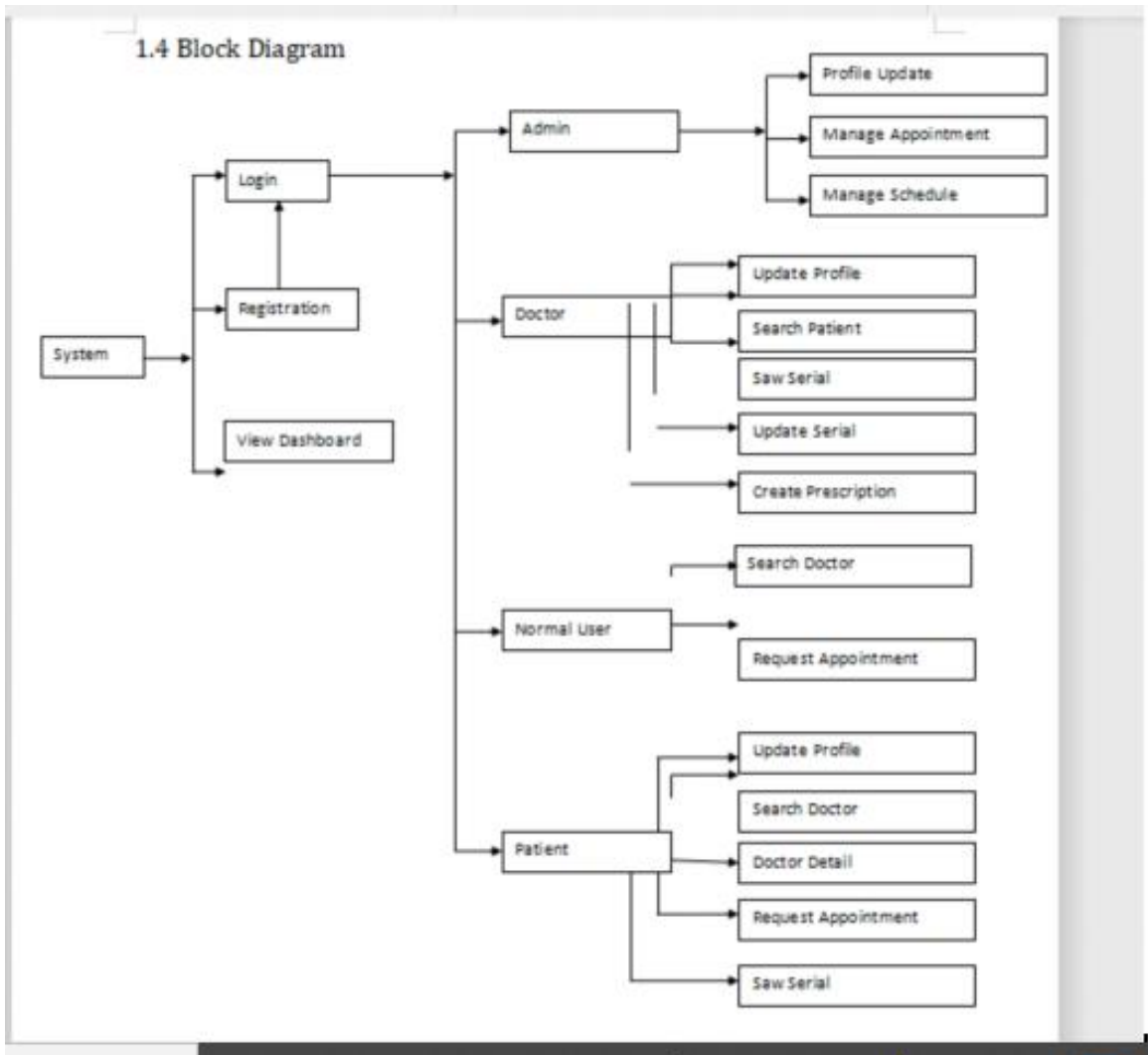


Figure 1.4: Block Diagram

1.5 Project Schedule

1.5.1 Gant Chart

| Task/Date | Start Date | End Date | Status | May | June | July | Aug | Sept | Oct | Nov | Dec | |
|----------------|------------|------------|--------|-----|------|------|-----|------|-----|-----|-----|--|
| Proposal | 07-05-2019 | 17-05-2019 | Comple | | | | | | | | | |
| Requirements | 20-05-2019 | 20-06-2019 | Comple | | | | | | | | | |
| Design | 23-06-2019 | 20-08-2019 | Comple | | | | | | | | | |
| Implementation | 07-09-2019 | 14-10-2019 | Comple | | | | | | | | | |
| Testing | 19-10-2019 | 30-10-2019 | Comple | | | | | | | | | |
| Documentation | 07-11-2019 | 02-12-2019 | Comple | | | | | | | | | |

Table 1.1: Gantt chart

Chapter 2

Software Requirement & Specification

2.1 Functional Requirement

1. Log in
2. Update Profile
3. Manage appointment
4. Doctor Details
5. Search Doctor
6. Search patient
7. Schedule
8. Create Prescription
9. View serial

2.2 Non Functional Requirement

- 1 Amazing Interface
- 2 Easy Language
- 3 User Friendly

2.3. Performance Requirements

2.3.1. Speed and Latency Requirements

1. Data would be inserted in MySQL database within commend like (php artisan make:(model appointment))
2. User interface design depends on the user execute table data set and show the database we have to use this commend (php artisan migrate)

2.3.2. Precisions or Accuracy Requirements

1. After every successful login user would show accurate page where he/she can show see details of appointment details, medical details, booked appointment, medical history, cancel & booking.
2. User can also have managed his/her profile.

2.4. Dependability Requirements

2.4.1. Reliability Requirements

1. Admin/User should log in to the system after input the valid email and password and update it.
2. User can make own profile manage it.
3. User can check update status.

2.4.2. Availability Requirements

1. Before run this application we have to run first xampp.
2. Using command we can run this properly & easily
3. This web application should run on a web browser (Preferable to chrome and Firefox).

2.5. Maintainability and supportability Requirements

2.5.1. Maintenance Requirements

1. This web system define environment did not modify or change
2. Missing or need change data we can recover it using command line.

2.6. Security Requirements

2.6.1. Access Requirements

1. Any type of user register or non-register can visit this web site
2. Manage or check profile user have to registration and login.

2.7. Usability and Human-Interaction Requirements

2.7.1. Ease of Use Requirements

The system UI is user-manual.

2.7.2 Understandability and Politeness Requirements

1. User can easily understand the system
2. Any user can use the system

2.7.3 Accessibility Requirements

1. The system should accessible from any other devices.
2. User should access their account within a request.

2.8 Legal Requirements

2.8.1 Standards Requirements

1. The admin should have good knowledge of different types of programming language.

Chapter 3
System Analysis

3.1 Use Case Diagram

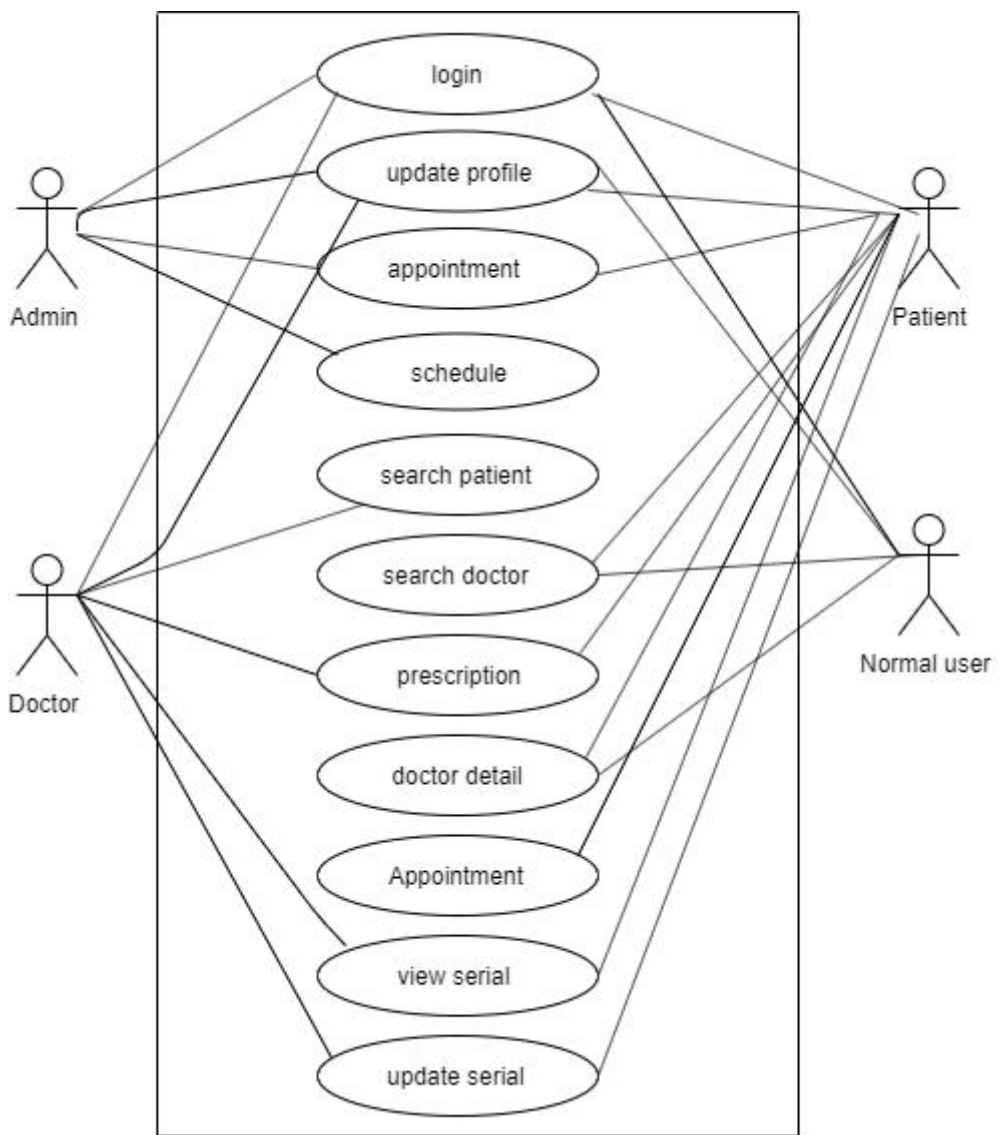


Figure 3.1: Use Case

3.2 Use Case Description:

3.2.1 Login/ Logout from system:

Table 3.1: Admin Login

| | | |
|--|---|--|
| Use case name: | Login/logout | |
| Use case no | 01 | |
| Goal | Selected user id and password then entire the system and | |
| Preconditions | Must be email and valid password for login. logout the system | |
| Primary Actors: | Admin | |
| Secondary Actors: | Doctor , Patient , Normal User | |
| Description / Main Success Scenario | Step | Action |
| | 1 | Confirm valid email address . |
| | 2 | Confirm valid password |
| Expectation | | Without internet you can launch web application. |
| Priority | | High |
| Frequency of user | | Just go to email and password login the system |

3.2.2 Update Profile:

Table 3.2: Update profile

| | | |
|--|--|--|
| Use case name: | Update Profile | |
| Use case no | 02 | |
| Goal | User can request update profile and admin can approving. User can get an email notification from admin | |
| Preconditions | Must email address and password | |
| Primary Actors: | Admin, | |
| Secondary Actors: | Doctor, Patient , normal user | |
| Description / Main Success Scenario | Step | Action |
| | 1 | Confirm valid email address . |
| | 2 | Confirm valid password . |
| Expectation | | |
| | | Without internet you can launch web application. |
| Priority | | High |
| Frequency of user | | Just go email address approving& disapproving result |

3.2.3 Appointment:

Table 3.3 Appointment

| | | |
|--|---|--|
| Use case name: | Appointment | |
| Use case no | 03 | |
| Goal | User can request appointment then admin approve the appointment | |
| Preconditions | Fill up the registration page in the system | |
| Primary Actors: | Patient | |
| Secondary Actors: | Admin | |
| Description / Main Success Scenario | Step | Action |
| | 1 | Check patient information |
| | 2 | Confirm patient appointment |
| | 3. | Confirm registration |
| Expectation | | |
| | | Without internet you can launch web application. |
| priority | | High |
| Frequency of user | | Just go email address approving& disapproving result |

3.2.4 Schedule:

Table 3.4 Schedule

| | | |
|--|----------------------|--|
| Use case name: | Schedule | |
| Use case no | 04 | |
| Goal | Update serial number | |
| Preconditions | Maintain Scheduling | |
| Primary Actors: | Admin | |
| Secondary Actors: | None. | |
| Description / Main Success Scenario | Step | Action |
| | 1 | Maintain scheduling for patient benefit |
| | 2 | Admin maintain the schedule |
| Expectation | | Without internet you can launch web application. |
| Priority | | Medium |
| Frequency of user | | Just go email address approving& disapproving result |

3.2.5 Search patient:

Table 3.5: Patient Search

| | | |
|--|--------------------------------------|--|
| Use Case | Search patient | |
| Use case no | 05 | |
| Goal | Doctor search the appointing patient | |
| Preconditions | Logged in | |
| Primary Actors: | Doctor | |
| Secondary Actors: | None. | |
| Description / Main Success Scenario | Step | Action |
| | 1 | Doctor search the patient and show patient details |
| Expectation | | Without internet you can launch web application. |
| Priority | | Medium |
| Frequency of user | | Doctor check patient information |

3.2.6 View Serial

Table 3.6: View Serial

| | | |
|--|---|--|
| Use Case | View Serial | |
| Use case no | 06 | |
| Goal | See patient serial number& doctor see serial number | |
| Preconditions | Logged in | |
| Primary Actors: | Doctor | |
| Secondary Actors: | Patient | |
| Description / Main Success Scenario | Step | Action |
| | 1 | Doctor call patient serial number |
| | 2 | Patient view his serial number |
| Expectation | | |
| | | Without internet you can launch web application. |
| Priority | | High |
| Frequency of user | | Just go email address saw the result |

3.2.7 Update Serial

Table 3.7: Update Serial

| | | |
|--|--|-----------------------------|
| Use Case | Update Serial | |
| Use case no | 07 | |
| Goal | The goal is to provide access to the doctor update serial number | |
| Preconditions | Logged in | |
| Primary Actors: Secondary Actors: | Doctor Admin , Patient | |
| Description / Main Success Scenario | Step | Action |
| | 1 | Doctor enters the system |
| | 2 | Doctor update serial number |
| | 3 | Admin Approve the serial |
| | 4 | Patient view update serial |
| Expectation | | |
| | Without internet you can launch web application. | |
| priority | Medium | |
| Frequency of user | Just go email address saw the result | |

3.2.8 Prescription

Table 3.8: Prescription

| | | |
|--|---|--|
| Use Case | prescription | |
| Use case no | 8 | |
| Goal | The goal is to provide access to the patient . doctor create prescription | |
| Preconditions | Logged in. | |
| Primary Actors: Secondary Actors: | Doctor Admin | |
| Description / Main Success Scenario | Step | Action |
| | | |
| | 1 | Doctor provide the prescription |
| | 2 | Patient get this prescription |
| Expectation | | Without internet you can launch web application. |
| priority | | High |
| Frequency of user | | Just go email address saw update the serial |

3.2.9 Search Doctor

Table 3.9: Search Doctor

| | | |
|--|--|--|
| Use Case | Doctor search | |
| Use case no | 9 | |
| Goal | The goal is to provide access to the search doctor | |
| Preconditions | Logged in | |
| Primary Actors: Secondary Actors: | Patient Normal user | |
| Description / Main Success Scenario | Step | Action |
| | 1 | Patient enters the system |
| | 2 | Confirm valid email address |
| | 3 | Confirm valid password |
| Expectation | | |
| | | Without internet you can launch web application. |
| priority | | Medium |
| Frequency of user | | Go to the website check detail |

3.2.10 view Doctor Details:

Table 3.10: Doctor Detail

| | | |
|--|---|--|
| Use Case | Doctor details | |
| Use case no | 10 | |
| Goal | The goal is to provide access to the user to modify their doctor details. | |
| Preconditions | Logged in. | |
| Primary Actors: | Patient | |
| Secondary Actors: | Normal User. | |
| Description / Main Success Scenario | Step | Action |
| | 1 | Patient enters the system |
| | 2 | Patient see doctor information |
| Expectation | | |
| | | Without internet you can launch web application. |
| priority | | Medium |
| Frequency of user | | Go to the website check detail |

3.2.11 Appointment

Table 3.11: Appointment

| | | |
|--|---|--|
| Use Case | Appointment | |
| Use case no | 11 | |
| Goal | The goal is to provide access to the user to modify their request appointment | |
| Preconditions | Logged in. | |
| Primary Actors: | patient | |
| Secondary Actors: | Admin | |
| Trigger | this will be a default function for the system | |
| Description / Main Success Scenario | Step | Action |
| | 1 | Patient enters the system |
| | 2 | Confirm valid email address |
| | 3 | Confirm valid password |
| Expectation | | |
| | | Without internet you can launch web application. |
| priority | | N/A |
| Frequency of user | | Just go email address saw the result |

3.3 Activity Diagram:

3.3.1 Login

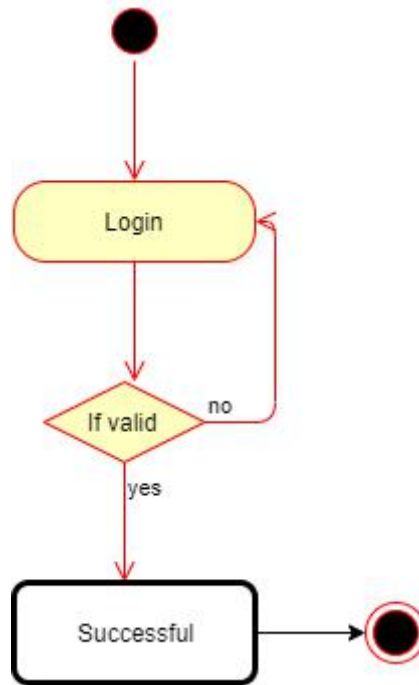


Figure 3.3.1: Login

3.3.2 Update Profile

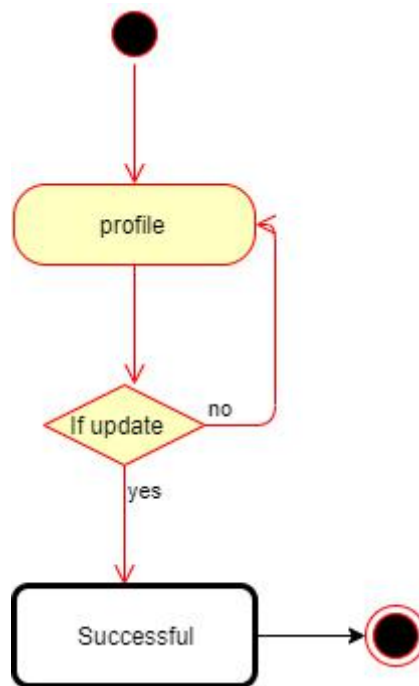


Figure 3.3.2: Update profile

3.3.3 Doctor search. Detail & Appointment

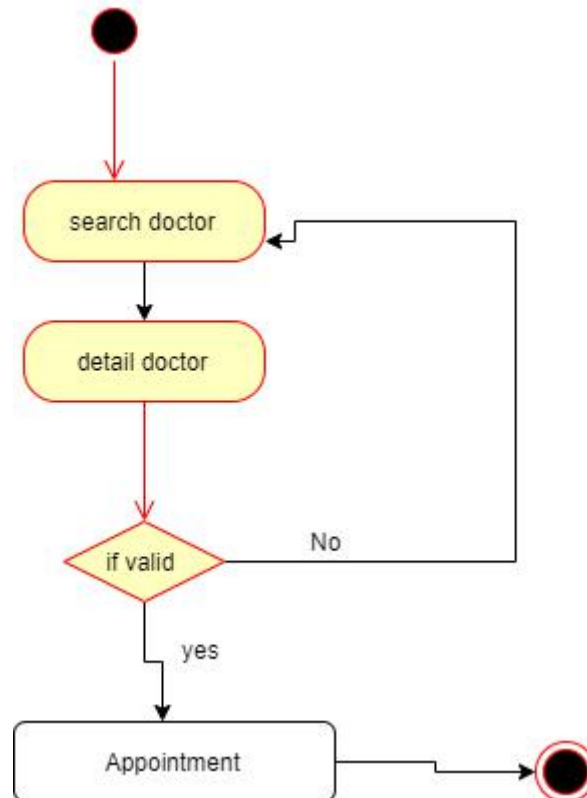


Figure 3.3.3 Doctor Search, Detail & Appointment

3.3.4 Schedule

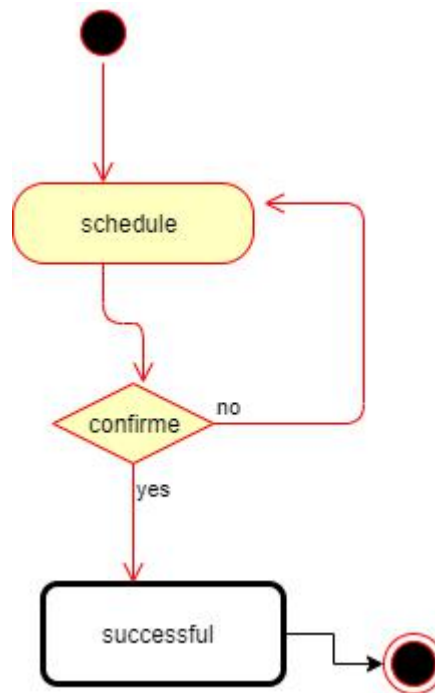


Figure 3.3.4: Schedule

3.3.5 View serial & Update serial

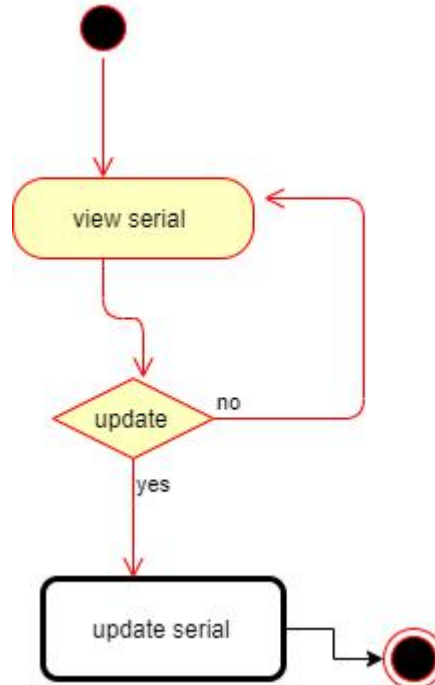


Figure 3.3.5 serial

3.3.6 Prescription

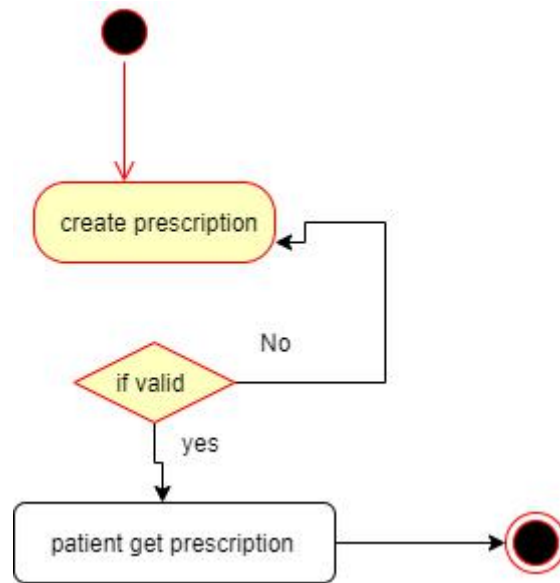


Figure 3.3.6: prescription

3.3.7 Search Patient

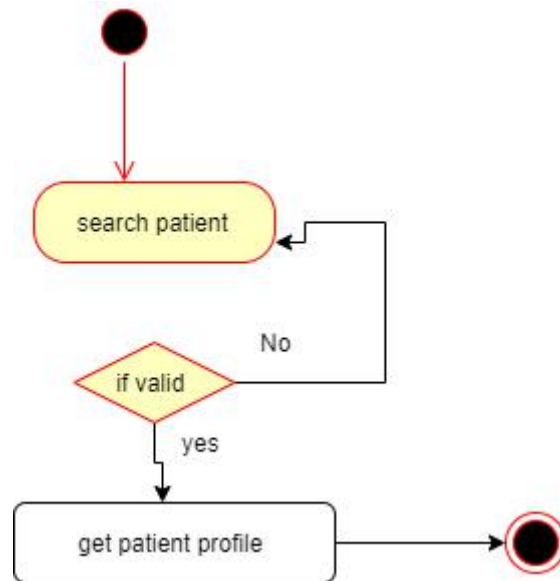


Figure 3.3.7: Search Patient

3.4 Sequence Diagram

3.4.1 Login

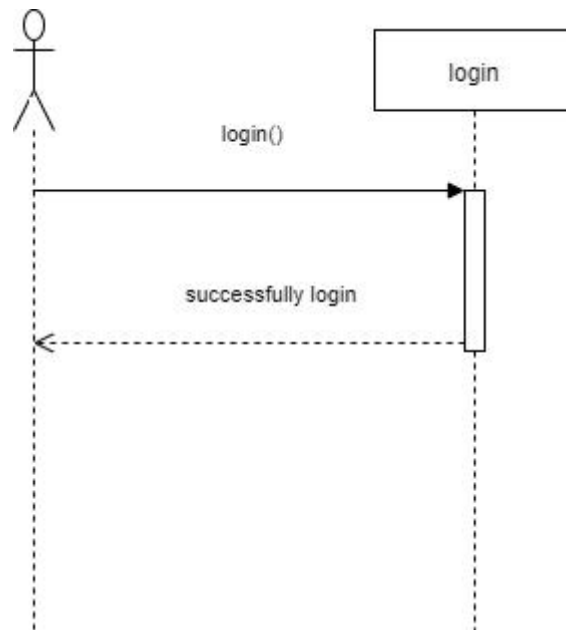


Figure3.4.1: login

3.4.2 Update profile

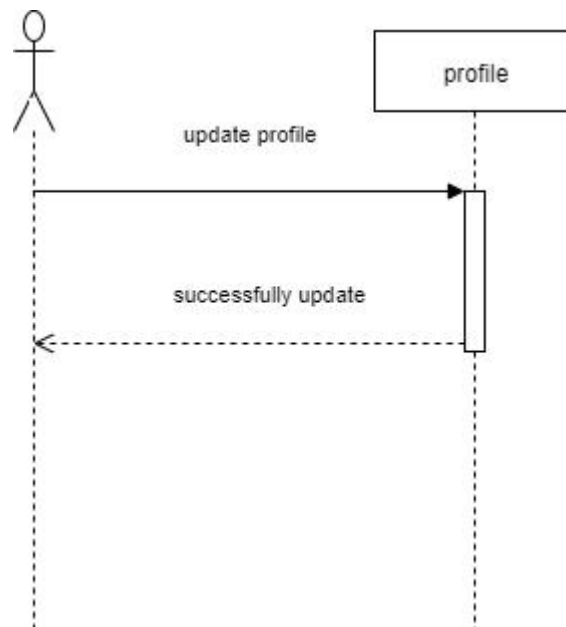


Figure 3.4.2: update profile

3.4.3 Schedule

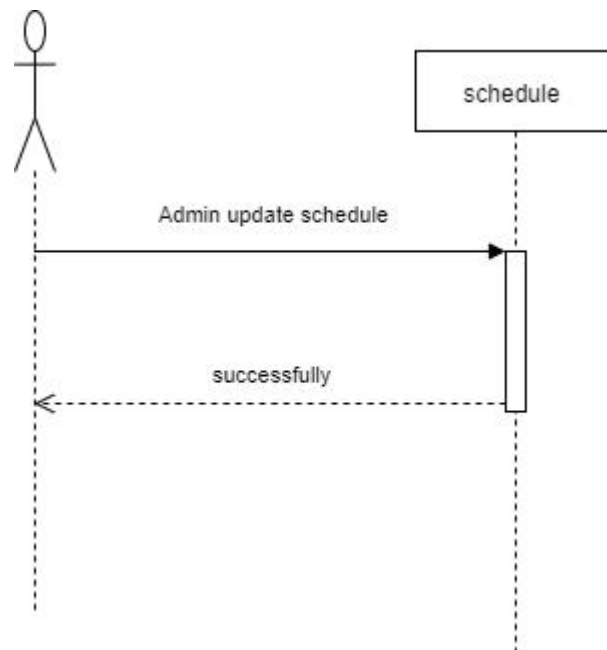


Figure 3.4.3: schedule

3.4.4 Search doctor

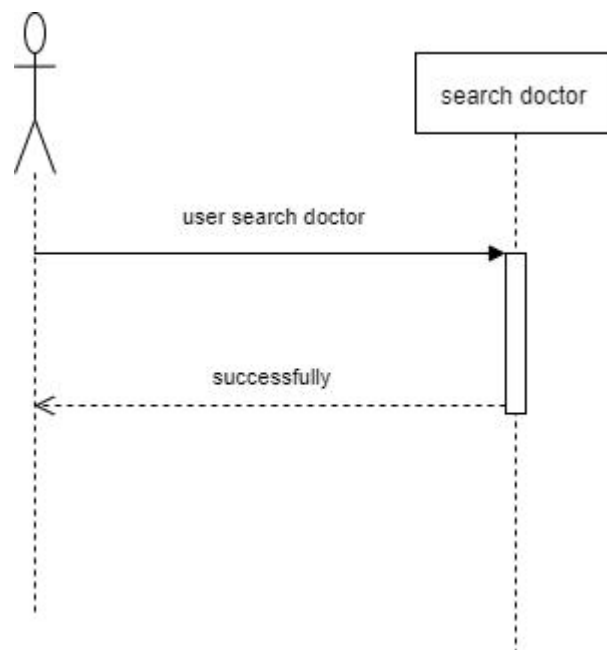


Figure 3.4.4 search doctor

3.4.5 Doctor Detail

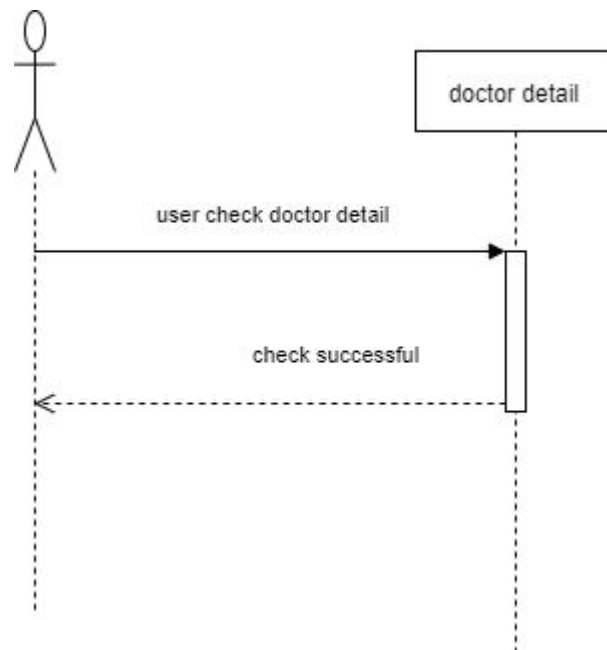


Figure 3.4.5: Doctor Detail

3.4.6 Search patient

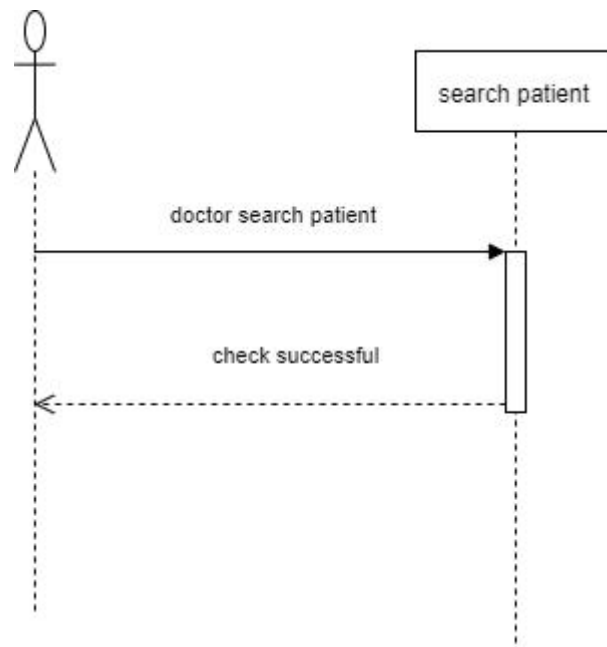


Figure 3.4.6: search patient

3.4.7 Appointment

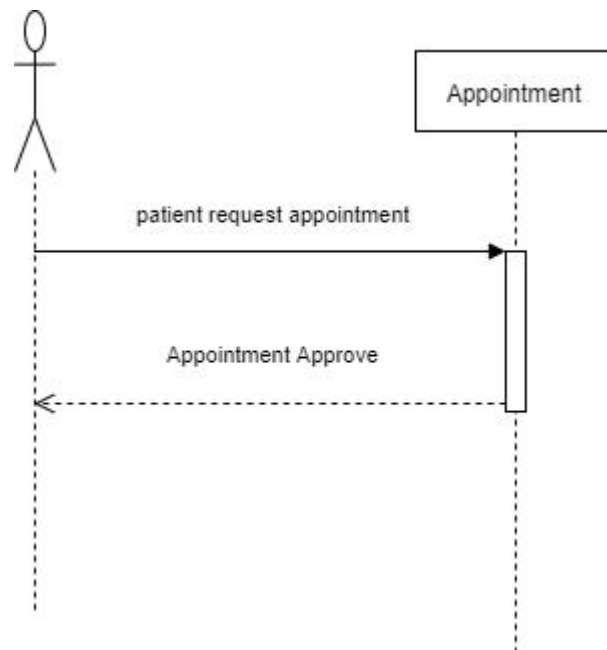


Figure 3.4.7: Appointment

3.4.8 Prescription

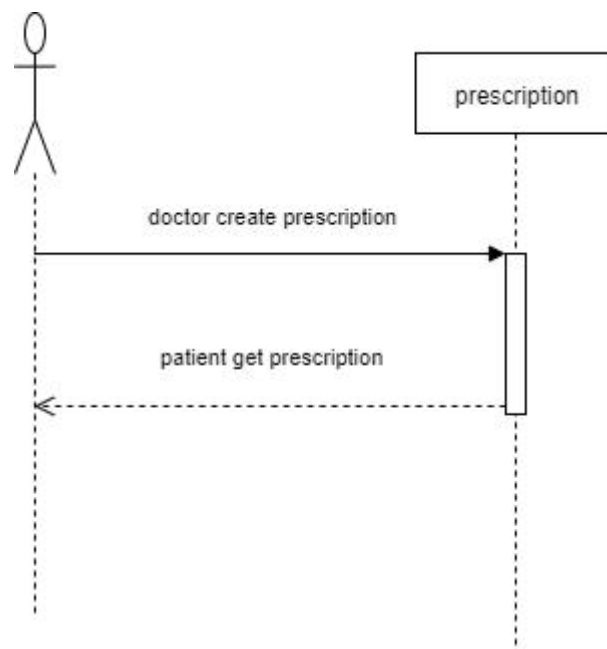


Figure 3.4.8: Prescription

3.4.9 Serial

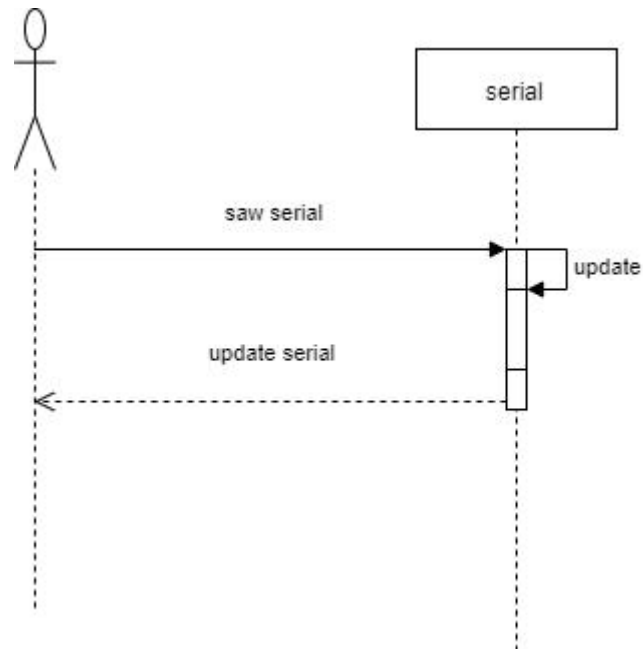


Figure 3.4.9: Serial

Chapter 4

System Design Specification

4.1 Entity Relationship Diagram:

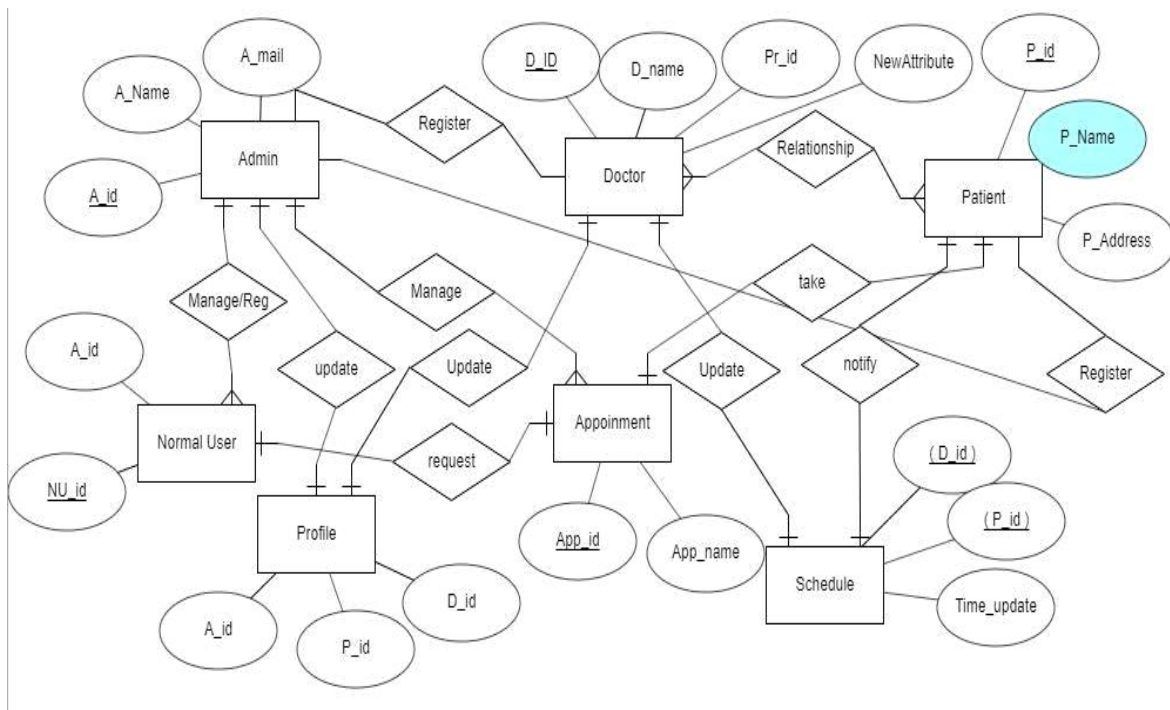


Figure 4.1: Entity Relationship Diagram

4.2 Class Diagram:

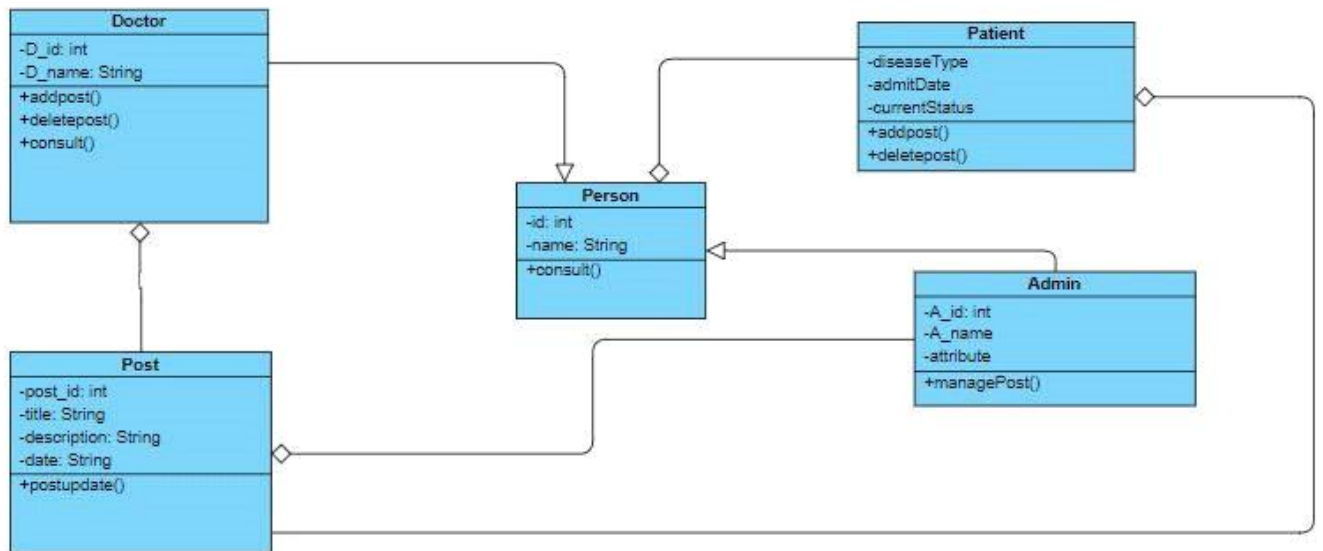


Figure 4.2: Class diagram

4.3 Development of tools and Technology

4.3.1 User Interface Technology.

1. HTML5
2. CSS3
3. Bootstrap 4
4. JavaScript
5. Mysqli

4.3.2. Implementation Tools & Platforms

1. Laravel Framework
2. Xampp
3. Sublime text editor 3
4. Windows CMD

Chapter 5

System Tasting

5.1. Testing Features

5.1.1. Features to be tested

1. Registration
2. Login

5.1.2. Features not to be tested

1. Rating System
2. Project Comment
3. IDE System

5.2. Testing Strategies

5.2.1. Test Approach

1. The whole system will be tested manually
2. System testing based on User acceptance.
3. System testing based on admin activities

5.2.2. Pass/Fail Criteria

1. Component Pass/Fail criteria – The test will pass if the case meets the object design requirement or fail if not.

2. Integration Pass/Fail criteria – The test will pass if the case meets the object design architecture requirement or fail if not.

5.2.3. Suspension and Resumption

1. Regression Testing – The system should work properly after each change on the system.

2. Database change: The system will not work properly if we change database name.

3. Build Acceptance Test – The system will pass the test if every build is successful if not then try the build again.

4. System Design Changes – The system should work properly after each change in the design.

5.2.4. Testing Schedule

Table 5.1: Testing Schedule (Programmer Form)

| Test Phase | Time |
|--------------------------------|--------|
| Test Plan Creation | 1 week |
| Test Specification Creation | 1 week |
| Test Specification Team Review | 2 week |
| Component Testing | 2 week |
| Integration Testing | 2 week |
| System Testing | 3 week |

5.3 Test case

5.3.1 Test case 1

Table 5.2: Test Case 1 (Patient Register)

| | |
|---------------------------------------|----------------------------|
| Test case #:01 | Test case name : sing up |
| System : Patient register | Subsystem : N/A |
| Design by : Mahmuda nipa(153-35-1385) | Design Date : 25-9-2019 |
| Execute By : Mahmuda Nipa | Execution date : 7-10-2019 |

Table 5.2: Patient Register

| Step | Action | Expected Response | System | Pass/fail | comment |
|------|---|--|--------|-----------|--|
| 1 | When a user fills up only Name field and clicks the register | Other fields are required | | pass | Other fields are required |
| 2 | When a user clicks only register button without a fill-up any field | Fill up the required field | | pass | Fill up the required field |
| 3 | When a user enters email like abc.com | The system should display the email field is not a valid e-mail address. | | pass | The email field is not a valid e-mail address. |

| | | | | |
|---|---|---|------|------------------------------------|
| 4 | When a user enters email like abc@gmail.com | The system will display the email field is invalid. | pass | The valid email needs to register. |
| 5 | When a user enters blood group like o+ | Other field are required | pass | Fill up the required field |

5.3.2. Test Case: 02

Table 5.3: Test Case 2 (Appointment)

| | |
|--|-----------------------------------|
| Test case #: 02 | Test case name : Make Appointment |
| System: Appointment | Subsystem :N/A |
| Designed By :Mahmuda Nipa(153-35-1385) | Design Date:25-9-2019 |
| Executed By: Mahmuda Nipa | Execution Date :7-9-2019 |

Table 5.3: Appointment

| Step | Action | Expected System Response | Pass/fail | comment |
|------|--|----------------------------|-----------|----------------------------|
| 1 | When a user fills up the only number | Other fields are required | pass | Other fields are required |
| 2 | When a user clicks only date without a fill-up any field | Fill up the required field | pass | Other fields are required |
| 3 | When user only choose button | Other fields are required | pass | Fill up required fields |
| 4 | When user only click gender button | Other fields are required | pass | Other fields are required |
| 5 | When user symptoms button without fill up any field | Fill up the required field | Pass | Fill up the required field |

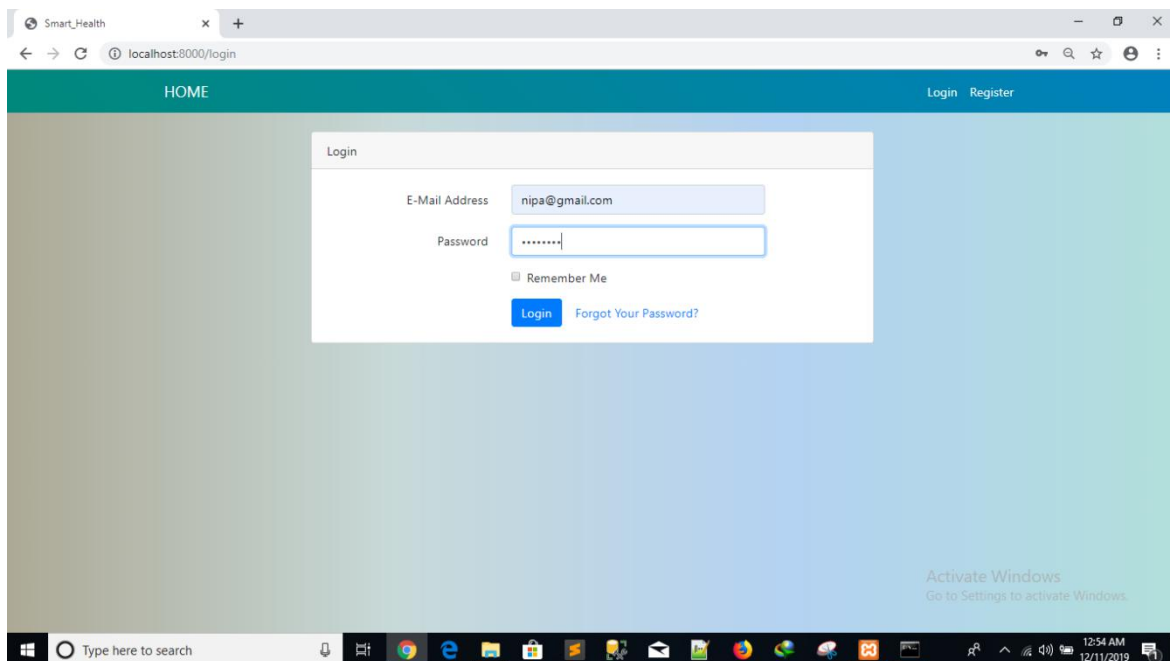
CHAPTER 6

USER MANUAL

6.1. User Manual (Patient/normal user)

6.1.1. User login

User needs to log in this system to get access. User get email & password and this email & password valid user click logging button .



6.1.2. User Registration

This is a user registration Page. User fill up the all requirement. Then registration done and patient & normal user get next page.

Smart_Health x +

localhost:8000/register

HOME Login Register

Register

Patient_name Nipa

Gender Female

Age 23

Blood_group 0+

E-Mail Address nipa@gmail.com

Password

Confirm Password

Register

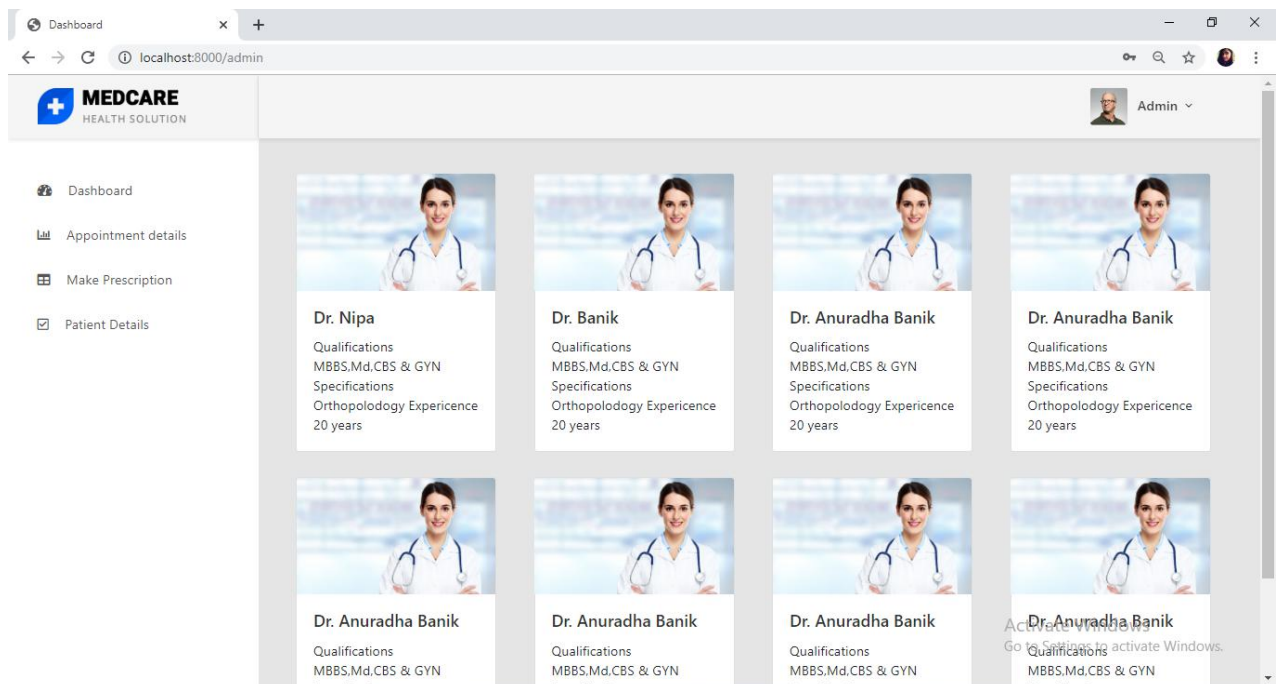
Activate Windows
Go to Settings to activate Windows.

Type here to search

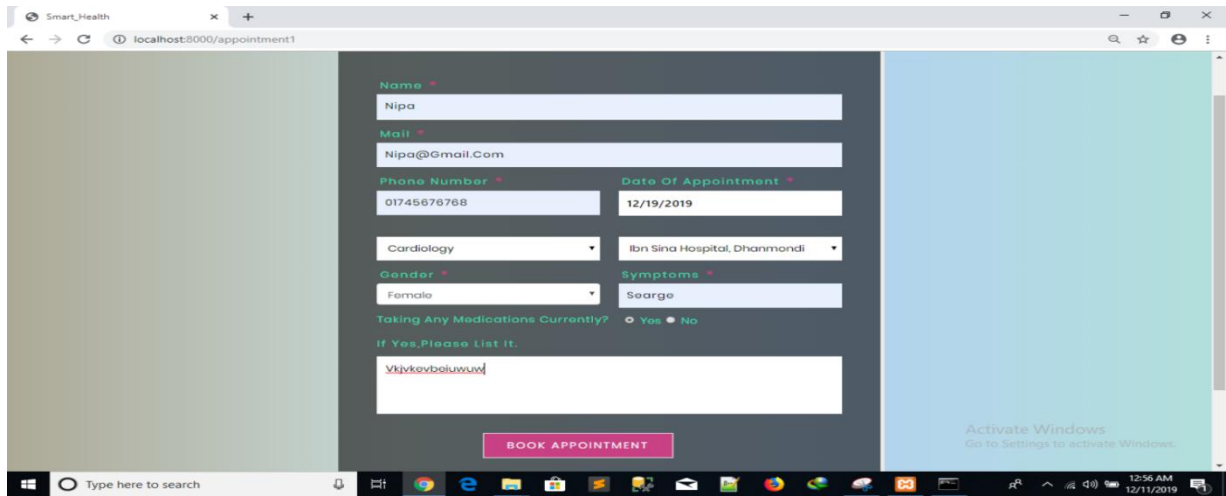
12:53 AM
12/11/2019

6.1.3. User Dashboard

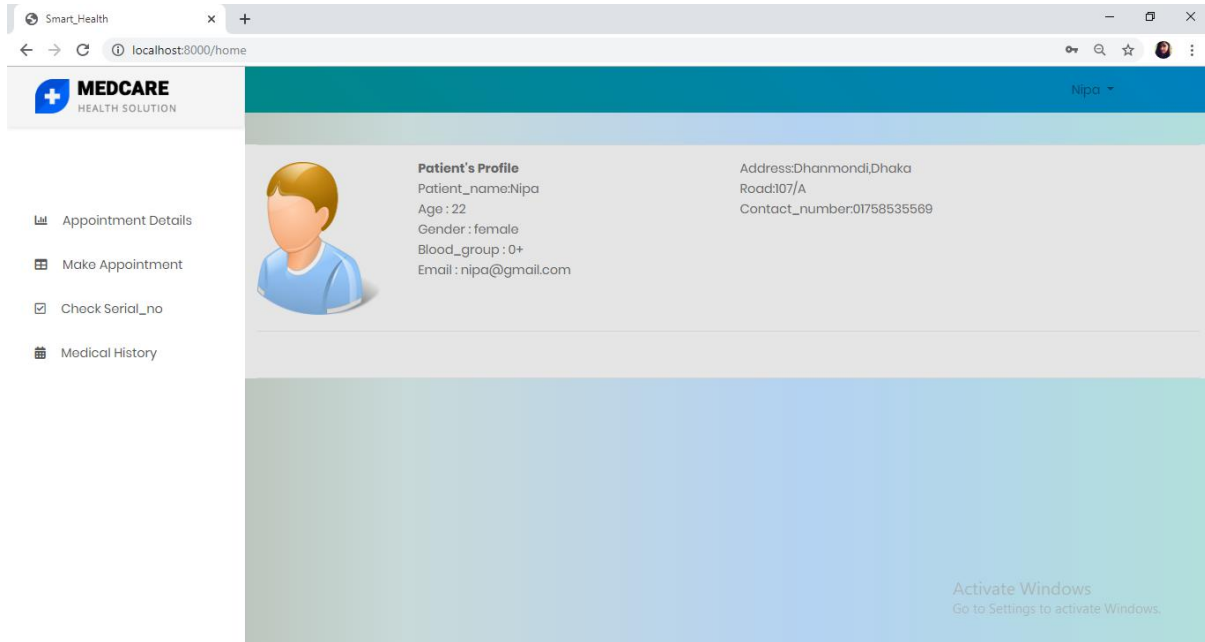
After successful login and registration, a user gets access to the system & using the Start new topic button.



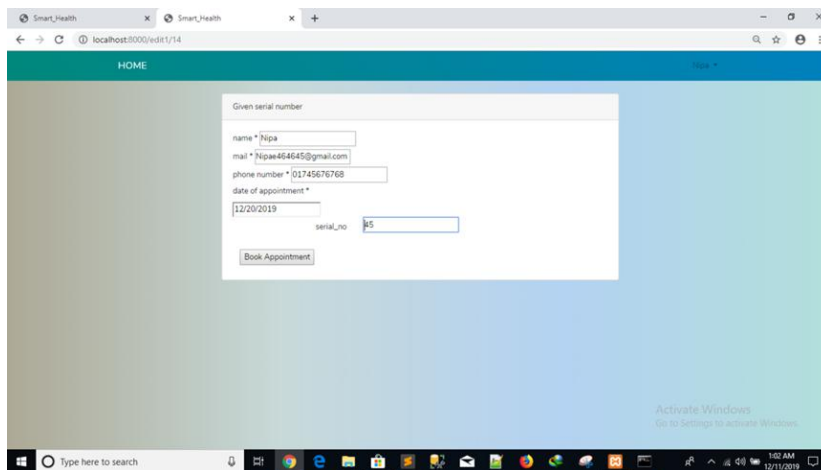
1.4 Patient Appointment: If patient appointment doctor patient fill up the appointment criteria. When it's done then patient get appointment.



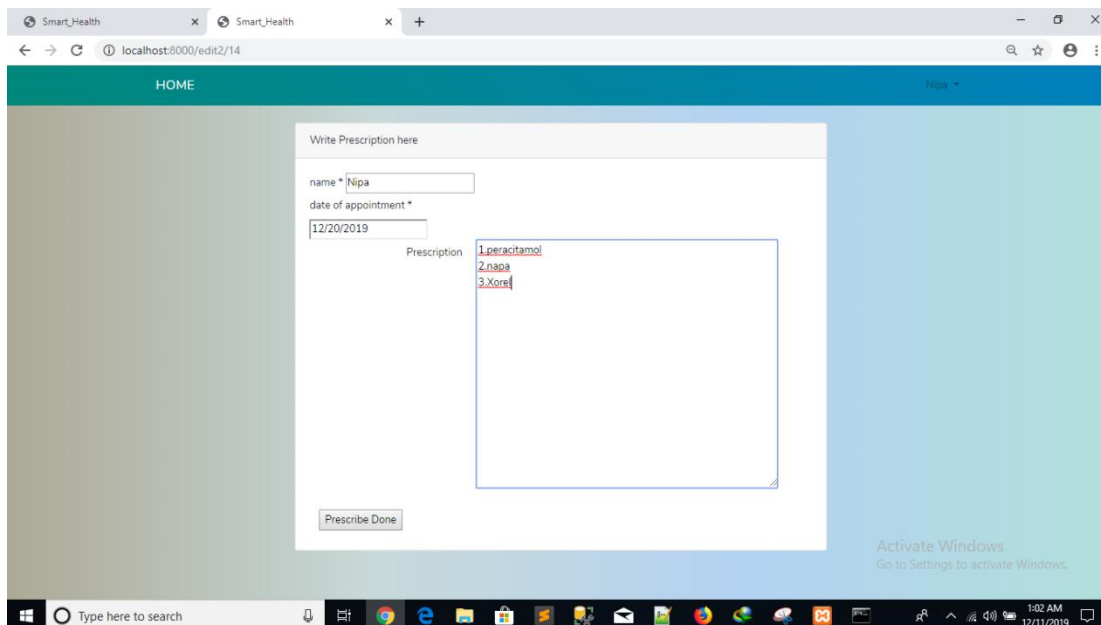
1.6 .5 patient profile: This profile patient all information. Patient get appointment a check serial number. All information save this page.



1.6.6 Automatically update your serial number.



6.1.7 Doctor write prescription: Doctor write a prescription. This prescription view doctor and patient patient .both



CHAPTER 7

PROJECT SUMMARY

7.1. GitHub Link: <https://github.com/Mahmuda001/smart-health>

7.2. Limitations

1. Patient can't registration, he/she can't appointment

7.4. Obstacles & Achievements

Day by day technologies are updated so the Laravel Framework is updating every day. It is quite tough to work with new technology but not so hard. New packages and new library functions are added regularly. So, I feature there are some technology can be eliminating and the project can be implemented with new technology. There are too many errors while I am developing the system.

7.5. Future Scope

Every project has some future scope so that in near future the application adds some new feature

1. Patient Consult with doctor and discusses his condition in online by message or video call.
2. Patient could treatment by online in emergency case.
3. Patient could consult or online discussion with foreign doctor.

Conclusion:

- As the analysis and examples in this project have demonstrated, a wide user influences an online health service, and patients can share their health status for maintaining and improving their health. The concept of Smart Health will grow in parallel with smart cities with more sensing capabilities in SC comes more and diverse Smart Health application. But serious and focused research is also required.

REFERENCES

- [1]. Database design and diagram [Access on 18 August 2019 10.21am] Link: <https://erdplus.com/#/diagrams>
- [2]. Activity Diagram, Use case Diagram, Class Diagram [Access on 20 June 2019 12.21am] Link: <https://diagrams.visual-paradigm.com/#>
- [3]. Laravel 5 Essentials Paperback – April 28, 2015 by [Martin Bean](#)
- [4]. https://laravel.com/?fbclid=IwAR1rHqpJoetbh6trCMOrQ3tKLSy6R3NNR4wOybAQjNBOvl_wYFKclZ4_pS0
- [5]. https://www.google.com/search?q=bootstrap+download&rlz=1C1CHBF_enBD855BD855&oq=bootstarp+dow&aqs=chrome.1.69i57j0l7.15651j0j7&sourceid=chrome&ie=UTF-8
- [6]. <https://www.youtube.com/coderstapehttps://www.youtube.com/coderstape>
- [7]. https://www.youtube.com/watch?v=5dIHU1jeb_s
- [8]. https://www.google.com/search?q=bootstrap+design&rlz=1C1CHBF_enBD855BD855&oq=bo&aqs=chrome.0.69i5912j69i57j0j69i6014.1508j0j7&sourceid=chrome&ie=UTF-8

