hh

Software Requirement Specification Document
V4 Hospital Management System

abdelrahman ehab,ziad mohammed,omar saeed,youssef mohammed

June 2020

Contents

1 Introduction **4**
- 1.1 Purpose of the document . . . 4
- 1.2 Scope of the System . . . 4
- 1.3 System overview . . . 4
- 1.4 the business context . . . 4

2 General Description **5**
- 2.1 Product Functions . . . 5
- 2.2 Similar Systems . . . 5
- 2.3 User Characteristics . . . 5
- 2.4 Problem Statement . . . 8
- 2.5 User Objectives . . . 8
- 2.6 General Constraints . . . 8

3 Functional Requirements **9**

4 Interface Requirements **19**
- 4.1 GUI . . . 19
- 4.2 Hardware Interfaces . . . 28

5 Performance Requirements **28**

6 Design Constraints **28**

7 Non-Functional Requirements **28**
- 7.1 security . . . 28
- 7.2 Maintainability . . . 28
- 7.3 Re-usability . . . 28
- 7.4 Availability . . . 28
- 7.5 reliability . . . 28

8 Operational Scenarios **29**

9 Preliminary Schedule Adjusted **30**

10 References **32**

task name	Finish date
SRS v1	19/3/2019
SRS V2	25/5/2020
SRS V3	2 /6/2020
SRS V4	10 /6/2020

1 Introduction

1.1 Purpose of the document

the purpose of this Software Requirements Specification document is to show our hospital system specifications and how it shall be used and who will be using it[8].

1.2 Scope of the System

Currently Hospital system is using a manual system to handle the hospital process . instead of patients going to the hospital to reserve an arrangement with the doctor. Then the patients diagnosed symptoms related disease details and other necessary details are being recorded and those are being recorded in a file. Calculation of bills are done manually.these files can be easily destroyed by fire and natural disasters, Also admin and receptionist can view data of users and patients. they may lose data and information.the amount of files used to save the information waste a huge space.it takes time to check previous record of patients ,or add new records ,or update records.its also tough.its hard to relay on manual calculations done by the staff and receptionists. our system shall overcome the above mentioned problems.

1.3 System overview

our system shall help patients to arrange an appointment in the hospital through the web application to consult the chosen doctor .firstly the user creates an account then sign in to the system where the user shall find different types of treatment specialists , their descriptions and available times then the user shall choose the doctor they desire after the user reserve the doctor the user shall receive the description with the price, the appointment time and number of patients waiting before him for the reservation. some kind of staff are allowed to view data or Edit, add, and delete it.

1.4 the business context

the purpose of this application is to help patients reserve an appointment fast ,avoid wasting time in the reservation process. manage data of users and patients on the system by adding, deleting, and editing it,financially it reduces the amount of paper work.

2 General Description

2.1 Product Functions

- Doctors Data Management : (View data / update data / Delete data)
- Receptionists Data Management : (View data / update data / Delete data)
- Patient Management : (appointments, reservations history, registrations, discharge, payment history, medicine, view and print prescription record)
- Patient Record Editing / updating / deleting : (Patient History, medicine required, preferred doctor)
- Personnel Management (Doctors) : (Doctors records, Doctors timings, Doctor availability i.e. day date and time)
- Staff management : (update job description of Doctors , add new staff member , tracks recruiting records , track timing records)
- prescription : (view and print diagnoses prescriptions written by doctors for each patient)
- Patient Portal : (patient's information, appointments and prescriptions)

2.2 Similar Systems

systems that our system is similar to:

- Saudi German hospital [1]
- mayo clinic [2]
- Indian University health [3]
- northwestern medicine [7]
- mount Sinai hospital [5]
  But it can be different in functionalities.

2.3 User Characteristics

1. Admin :

   - Admin can control everything in the system(full Access). which means that the admin is able to control any activity on the system.He has the highest privileged of the users of the system.

   Key functions:

   - Manage Staff , patients , and admin profiles

- Manage the web application
- Allocate resources
- Administer the charges
- Provide Technical Support
- Keep track of patient details
- Manage Staff , patients , and admin accounts
- View all patient, Doctor, Receptionist records
- Manage doctors accounts, and their timings
- Keep track of patient's Appointments
- Manage Departments

2. Doctor :

   - Doctors has a very limited access to the system, as he has few control on it.

   Key functions:

   - view patients who reserved appointments
   - Edit account information
   - Manage appointments by approving or deleting them
   - writes prescriptions about the condition of the patient and send to patient so he can view or print it.

3. patient:

    - Doctors has a very limited access to the system, as he has few control on it.

    Key functions:

    - reserve appointment
    - view prescription records
    - Register with site
    - view prescription
    - view appointment history
    - print prescription record

4. Receptionist:

    - Receptionist access few pages on the system and has limited activities on it

    Key functions:

    - reserve appointment
    - view Doctors
    - view Doctor timings
    - create patient
    - view approved appointments
    - edit profile

2.4 Problem Statement

This system provides solutions for all user's problems, the client want the system to move from paper based to electronic medical records to help them to provide premium health care to their patients and improve the medical services offered It provides customer comfort for reservations, viewing prescriptions and view doctors of all departments. All of which should be easy to use, and handle the system crash.

2.5 User Objectives

1. Reservations management :
    - Fast and easy to reserve an appointment
    - Different appointment time to each patient
2. Analysis System :
    - analysis on Each user information
    - analysis on all reservations happened on system
3. Accounts Management :
    - full access on users accounts (Add, Edit or Delete)
    - Manage reservations (Delete and view)

2.6 General Constraints

Each user should has individual loginid and password,
Only the admin can access the whole system (Admins),
the System is only accessible within the hospital premises only and
User should have an account to access the system and make an appointment.

3 Functional Requirements

- The user access system process takes place either by Sign-in or Registration or can sign out from system by clicking on logout[4]

Function	Register
Description	Add new user to the system so he could login with same identity later on and do system's operations which is reserve an appointment
Input	username, password , phone, full name
Output	save data on database and tell user whether the process was accepted or not
action	checks if the username entered by the user is already registered , if it is ,it tells the user to enter another username , if it isn't its proceeds to home page.
precondition	the username must not be already registered on the database, and Server must be connected to the Database.
post-condition	the username information should be saved in the database according to the validations required

Function	sign in
Description	user shall be able to sign in after checking his username and password are in the database
Input	username, password entered by user
Output	directs to the hospital system main page
action	check if username and password are in the database
precondition	the username that the user enters should be registered on the database, and Server must be connected to the Database.
post-condition	the user is signed in, and can do system's operations

Function	sign out
Description	the user shall be able to sign out
Input	the user presses the sign out button
Output	returns the user to the sign in page
precondition	the user should be signed in, and Server must be connected to the Database.
post-condition	the user isn't signed in anymore

- Insert new user information to the system either by sign-up or add the information by the administrator (admin) or Receptionists. This process takes various functions to be done.

Function	Insert user
Description	This function is accessed by admin, he can be able to create new user to system
Input	Insert user
Output	User ID of the user to be entered
action	these data shall be added to the database
precondition	the username 'shouldn't be in database already, and Server must be connected to the Database.
post-condition	the username shall be added to the database, username and password will be sent to user

Function	Insert user value
Description	Insert user information in the database table called "Users" according to user's entered automatically ID in database.
Input	The form information and user previous entered automatically ID from database
Output	create new row in database with the new user's information added by administrator (admin)
action	input all required user information and follow the validations
precondition	input of information should match the Validations, and Server must be connected to the Database.
post-condition	User information shall be added to the database, so user can sign-in later

Function	Insert Doctor
Description	This function is accessed by admin, he can be able to add new Doctor to system
Input	Insert user
Output	doctor ID of the doctor to be entered
action	these data shall be added to the database
precondition	the username 'shouldn't be in database already, and Server must be connected to the Database.
post-condition	the username shall be added to the database

Function	Insert Doctor value
Description	Insert Doctor information in the database table called "Doctors" according to user's entered automatically ID in database.
Input	The form information and Doctor previous entered automatically ID from database
Output	create new row in database with the new Doctor's information added by administrator (admin)
action	input all required user information and follow the validations
precondition	input of information should match the Validations, and Server must be connected to the Database.
post-condition	User information shall be added to the database, so user can sign-in later

- Deleting user from the system by the administrator (admin). This process takes various functions to be done.

Function	Delete user from system
Description	only the administrator shall be able to delete any user from the system and database
Input	pressing on delete user button and confirm deletion
Output	Delete user value
precondition	the user shall have an account registered in the database, and Server must be connected to the Database.
post-condition	the user information shall be deleted from the database

Function	Delete user value from system
Description	only the administrator shall be able to delete any user from the system and database
Input	Selecting user ID
Output	a pop-up saying that the user has been deleted successfully
precondition	the user shall have an account registered in the database, Server must be connected to the Database, and user Data must be already existed in the Database.
post-condition	the user account information shall be deleted from the database

Function	Delete Doctor from system
Description	only the administrator shall be able to delete any Doctor from the system and database
Input	pressing on delete user button and confirm deletion
Output	Delete Doctor value
precondition	the Doctor shall have an account registered in the database, and Server must be connected to the Database.
post-condition	the Doctor information shall be deleted from the database

Function	Delete Doctor value from system
Description	only the administrator shall be able to delete any Doctor from the system and database
Input	Selecting Doctor ID
Output	a pop-up saying that the Doctor has been deleted successfully
precondition	the Doctor shall have an account registered in the database, Server must be connected to the Database, and Doctor Data must be already existed in the Database.
post-condition	the Doctor account information shall be deleted from the database

- Update user information either by administrator (admin) or by user itself.
  This process takes various functions to be done.

Function	Update user information(administrator)
Description	Admin can edit the user's information id assigned to each user in table users in the database
Input	User ID and the section is to be edited
Output	the selected row in database is updated
action	Confirm Editing user's information
precondition	pressing confirm button after filling the section selected to be edited, Server must be connected to the Database, and user Data must be already existed in the Database.
post-condition	User's information is updated successfully according to each row validations

Function	Update user profile (user)
Description	User update his own information by himself in his profile and the this information is automatically updated in the database
Input	username, password , phone, full name
Output	pop-up message saying information is updated successfully
action	User updates his own information in profile
precondition	user must be authenticated, Server must be connected to the Database, and user Data must be already existed in the Database.
post-condition	None

Function	Update Doctor information(administrator)
Description	Admin can edit the Doctor's information id assigned to each Doctor in table doctor in the database
Input	Doctor ID and the section is to be edited
Output	the selected row in database is updated
action	Confirm Editing Doctor's information
precondition	pressing confirm button after filling the section selected to be edited, Server must be connected to the Database, and Doctor Data must be already existed in the Database.
post-condition	Doctor's information is updated successfully according to each row validations

Function	Update Doctor profile (Doctor)
Description	Doctor update his own information by himself in his profile and the this information is automatically updated in the database
Input	doctor name , mobile number , department ,login id , education , experience , consultancy charge
Output	pop-up message saying information is updated successfully
action	doctor updates his own information in profile
precondition	doctor must be authenticated, Server must be connected to the Database, and doctor Data must be already existed in the Database.
post-condition	None

- view all users by administrator (admin) or Doctors or Receptionists. admin can select a specific user to view all his information

Function	view user profile
Description	Admin view any user profile he selects on the system
Input	User ID / pressing edit button beside user's information on the system
Output	User information / profile
action	admin views user's profile
precondition	only admin can make this action, and the selected user must not be already deleted, Server must be connected to the Database, and user Data must be already existed in the Database.
post-condition	None

Function	view all users in system
Description	Admin view all users profile on the system
Input	None
Output	Array of all users
action	admin views all users
precondition	only admin can make this action, and the selected user must not be already deleted, and Server must be connected to the Database.
post-condition	None

- view all users only by administrator (admin). admin can select a specific user to view all his information

Function	view Doctor profile
Description	Admin view any Doctor profile he selects on the system
Input	Doctor ID / pressing edit button beside Doctor's information on the system
Output	Doctor information / profile
action	admin views Doctor's profile
precondition	only admin can make this action, and the selected Doctor must not be already deleted, Server must be connected to the Database, and Doctor Data must be already existed in the Database.
post-condition	None

Function	view all doctors in system
Description	Admin view all doctors profile on the system
Input	None
Output	Array of all doctors
action	admin views all doctor
precondition	only admin can make this action, and the selected doctor must not be already deleted, and Server must be connected to the Database.
post-condition	None

- can add new doctor's timings to the system.

Function	add doctor timings
Description	admin add new timings for a specific doctor
Input	doctor name, Time (from..to..) ,status
Output	pop up saying timing added successfully
action	none
precondition	none
post-condition	none

Function	view doctor timings
Description	user shall check the doctors available then select the doctor he wants to reserve with
Input	none
Output	array list of all doctors timings
action	none
precondition	none
post-condition	none

- add can add / view departments

Function	add new department
Description	add add new department to the system
Input	department name, description, and status
Output	pop-up:department added successfully
action	fill form
precondition	none
post-condition	none

Function	view
Description	admin can view all departments in the system
Input	none
Output	array of departments
action	
precondition	none
post-condition	none

- view all doctors on system available with days and times, so it will be easy for the user to choose a suitable appointment with any doctor. This reservation will be added to the user's profile so he can check date and time and doctor information as long as he sign-in.

Function	Reserve appointment
Description	user shall check the doctors available then select the doctor he wants to reserve with
Input	User name and password
Output	the system outputs a receipt with the information of the reservation
action	
precondition	the doctor should be available for reservations, and Server must be connected to the Database.
post-condition	a reservation time should be saved on the database with the doctor name and patient name and time for the reservation

Function	view reservation
Description	user shall be able to view his reservation
Input	pressing reservations button on his profile or in navigation bar
Output	pop-up message with the user's reservations
action	
precondition	user must reserve an appointment with a doctor or no results will appear in the pop-up message , Server must be connected to the Database, and Reservation Data must be already existed in the Database.
post-condition	None

Function	delete reservation
Description	user shall be able to delete his reservation
Input	pressing cancel appointment button on the system
Output	a pop-up telling the user that the reservation has been deleted successfully
action	
precondition	there should be a reservation saved on the database, and Server must be connected to the Database.
post-condition	the reservation should be deleted from the database successfully

- only Administrator (admin) can Add, edit or delete Doctors accounts from system and also from database automatically. Also admin can view the Doctor's list and update it frequently.

Function	Doctors List
Description	a list of the doctors that shall be updated frequently
Input	the admin shall input the information of the doctors and update it
Output	a list of the doctors with information about them
action	
precondition	doctors information should be saved on the database, and Server must be connected to the Database.
post-condition	if the admin updates the list it should be updated in the database

Function	Edit Doctor information
Description	a list of the doctors that shall be updated frequently
Input	the admin shall input the information of the doctors and update it
Output	a list of the doctors with information about them
action	
precondition	doctors information should be saved on the database ,Server must be connected to the Database, and Doctor information must be already existed in the Database.
post-condition	if the admin updates the list it should be updated in the database

Function	Delete Doctor value from system
Description	only the admin shall be able to delete any Doctor from the system and database
Input	Selecting Doctor ID
Output	a pop-up saying that the Doctor information has been deleted successfully
precondition	the Doctor shall have an account registered in the database, and Server must be connected to the Database.
post-condition	the Doctor account information shall be deleted from the database

- front disk Receptions make new profiles for patients who didn't register yet, also they print reports for patients
- More information by pressing buttons in the navigation bar which are contact us, emergency, and location of hospital.

Function	contact us
Description	a way to help the user to contact with someone from the hospital if needed through the website
Input	the user should be signed in
Output	a response from the person assigned to answer the users

Function	emergency
Description	a button that shows the number of the hospital to contact with them if needed
Input	the user should press on the emergency button
Output	a pop-up showing the number of the hospital

Function	location of the hospital
Description	a button that shows the location of the hospital in maps
Input	the user should press the button named location
Output	a pop-up showing the location

- Reports are done by doctors or admin but mainly by doctors

Function	prescription
Input	User ID, and name
Output	Reporting page
action	Write report on the patient profile
precondition	Server must be connected to the Database.
post-condition	report will be available on patient's profile so he can view it

- Receptionists can login as any user to use the system from hospital and also they can edit their profile's information

Function	Receptionist sign in
Description	Receptionist shall be able to sign in after checking that his username and password are in the database
Input	username, password entered by Receptionist
Output	directs to the hospital system main page
action	check if username and password are in the database
precondition	the username that the Receptionist enters should be registered on the database, and Server must be connected to the Database.
post-condition	the Receptionist is signed in, and can do system's operations

Function	Update Receptionist profile
Description	Receptionist update his own information by himself in his profile and the this information is automatically updated in the database
Input	Receptionist name, login id, status
Output	pop-up message saying information is updated successfully
action	Receptionist updates his own information in profile
precondition	Receptionist must be authenticated, Server must be connected to the Database, and Receptionist Data must be already existed in the Database.
post-condition	None

4 Interface Requirements

4.1 GUI

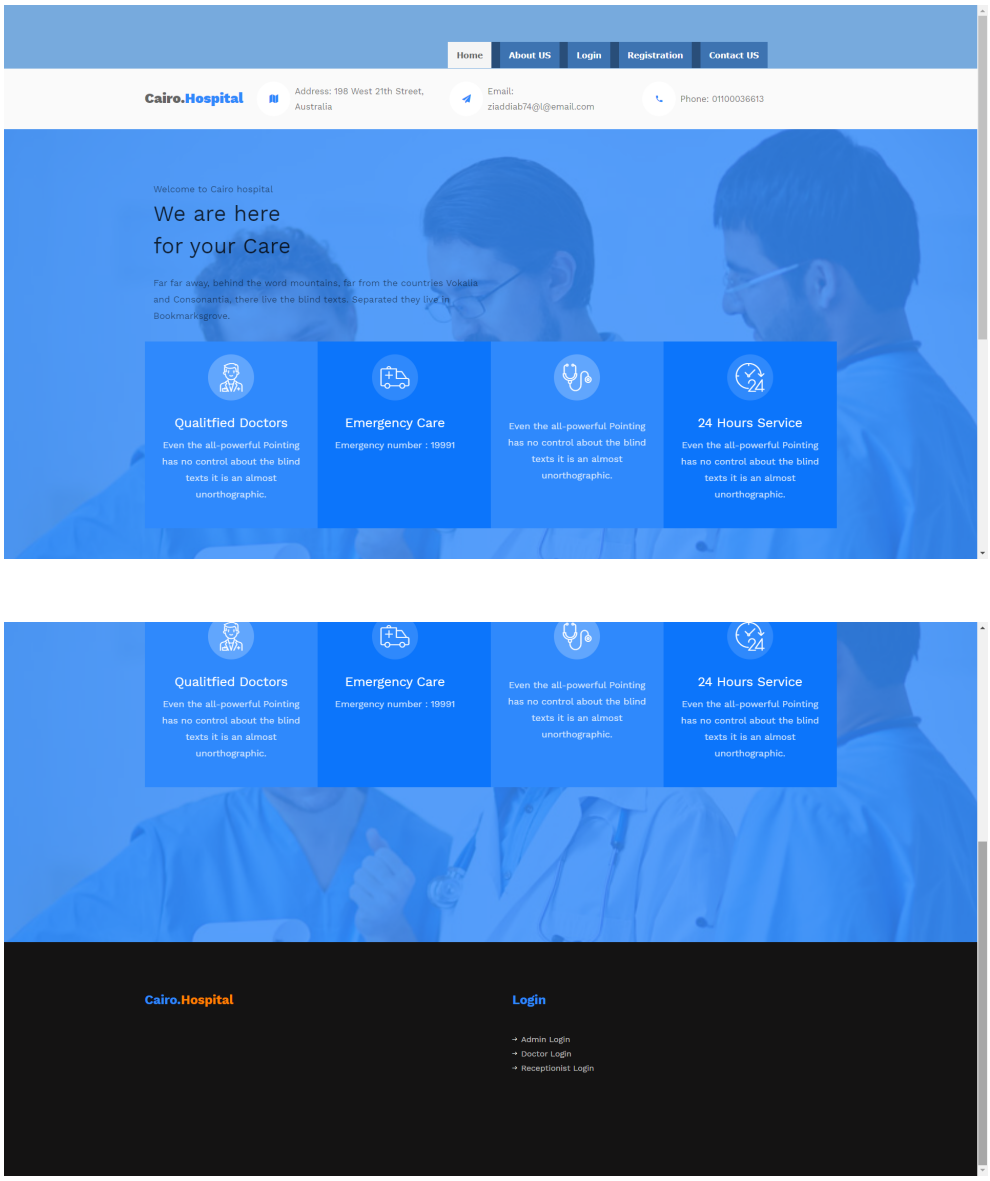

Figure 1: Home

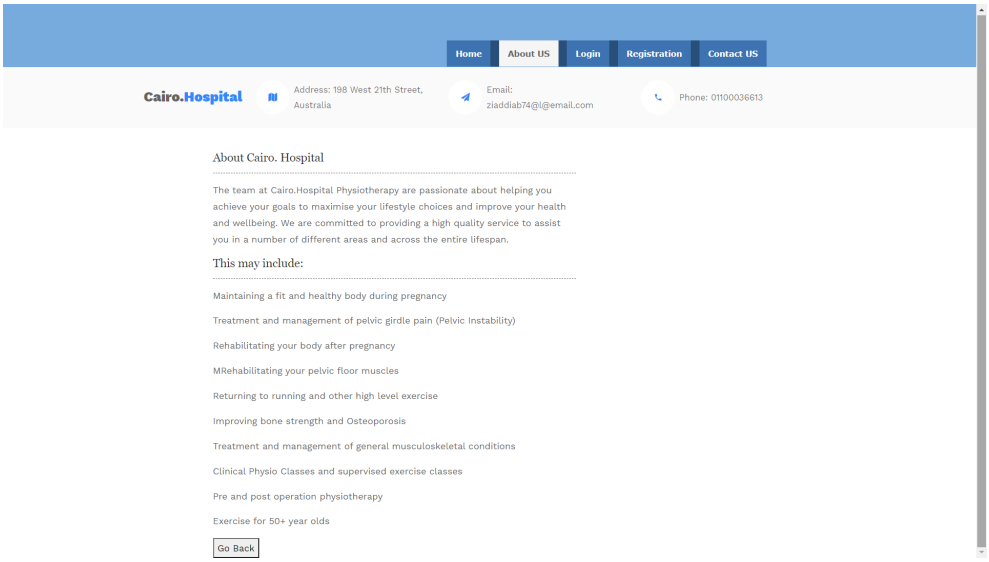

Figure 2: about us

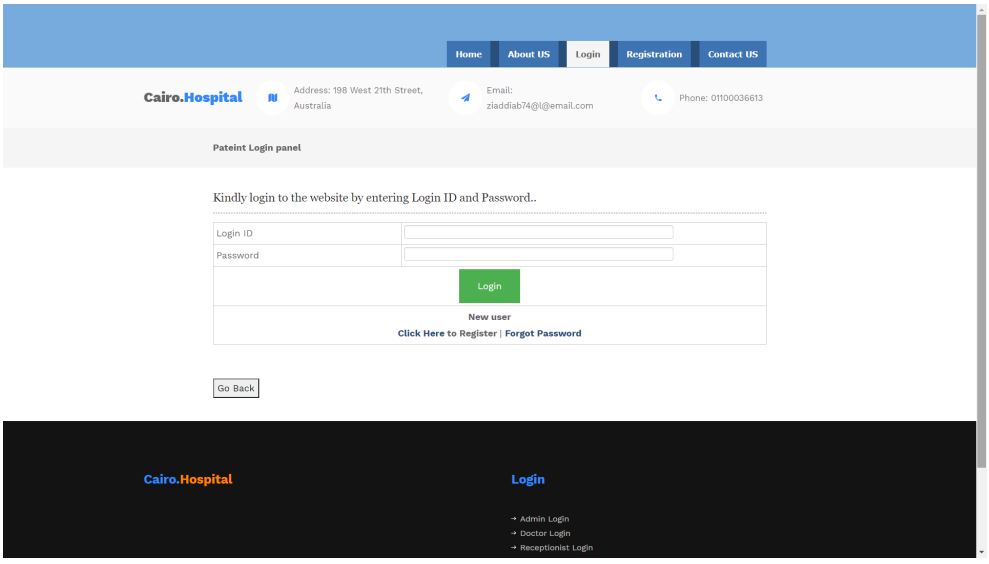

Figure 3: login panel

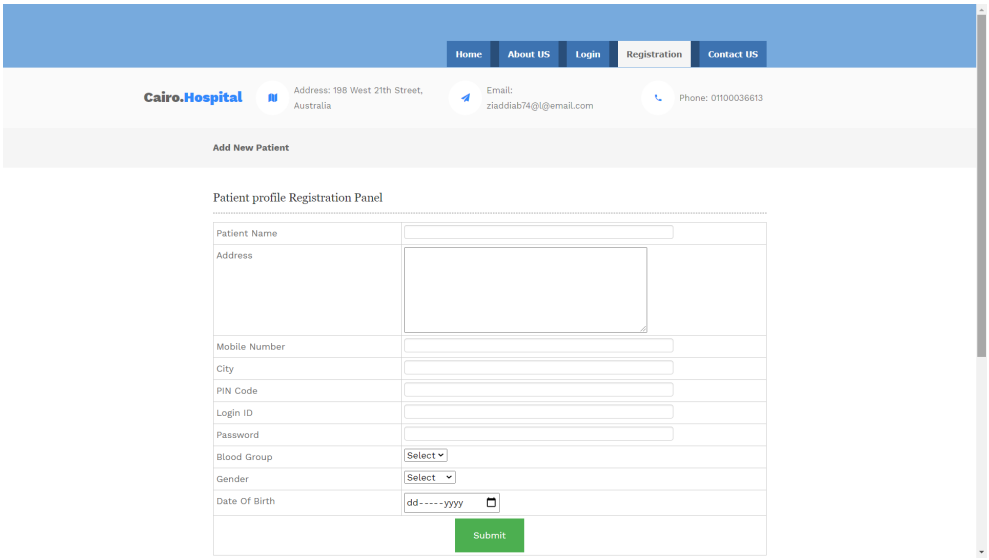

Figure 4: registration

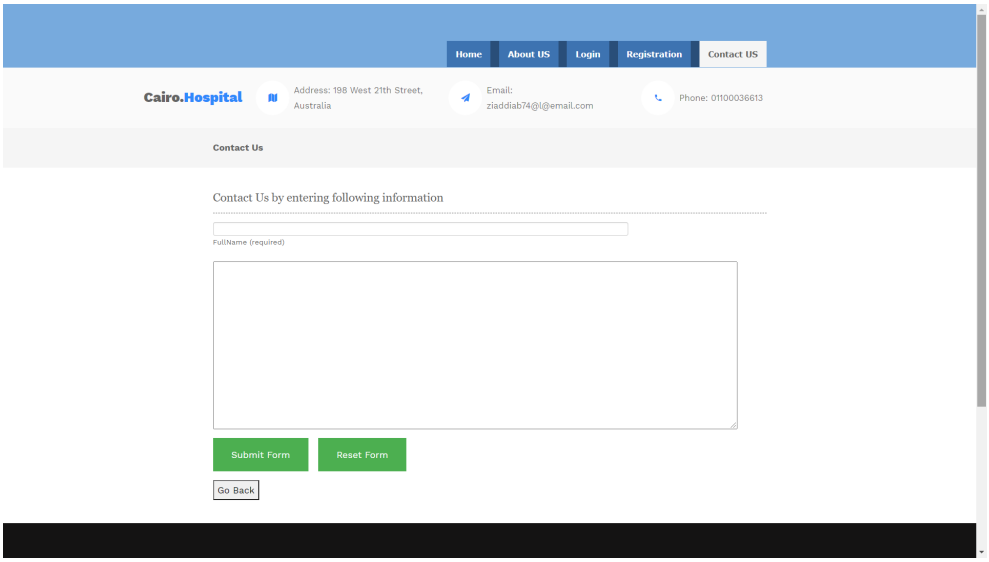

Figure 5: contact us

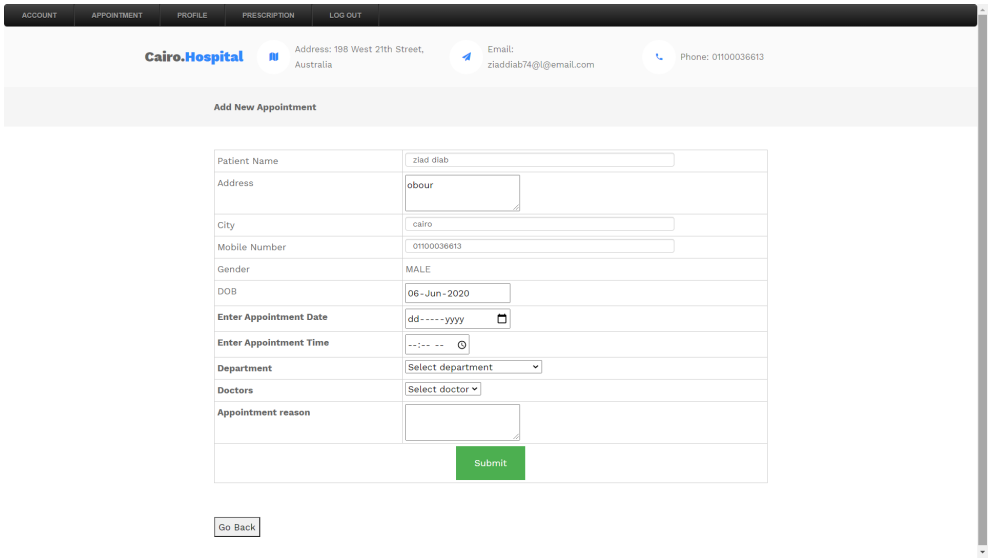

Figure 6: add appointment page

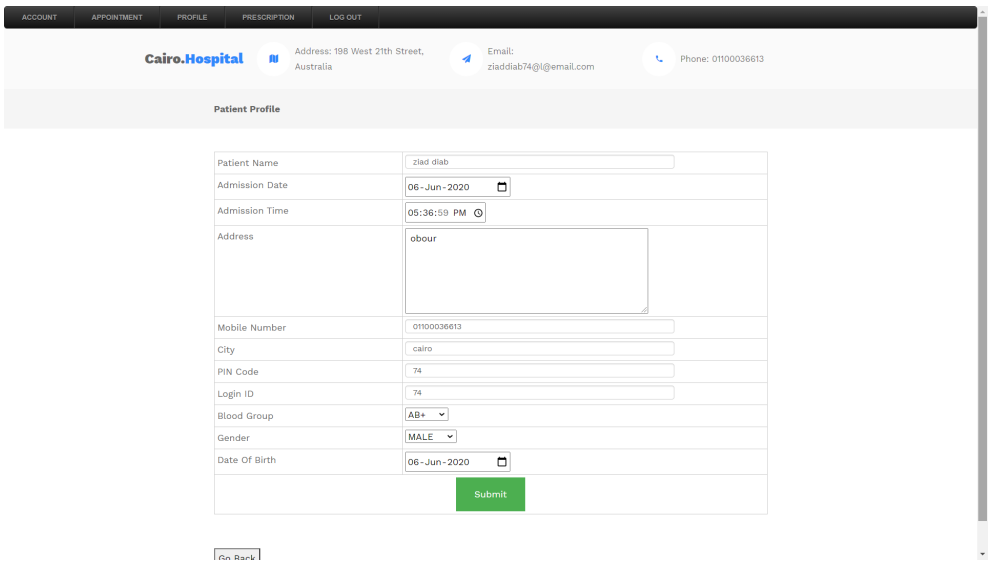

Figure 7: view and edit profile page

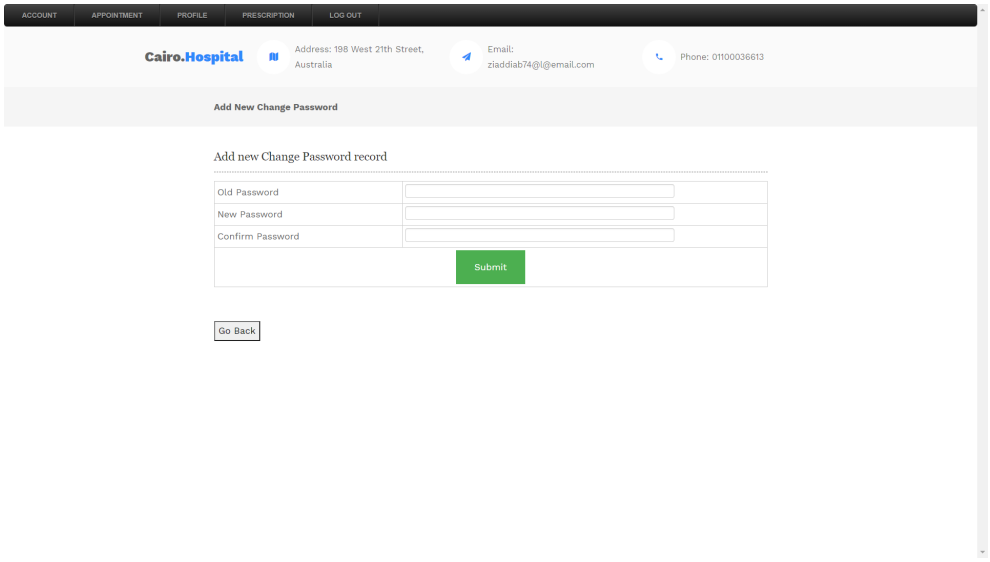

Figure 8: change password

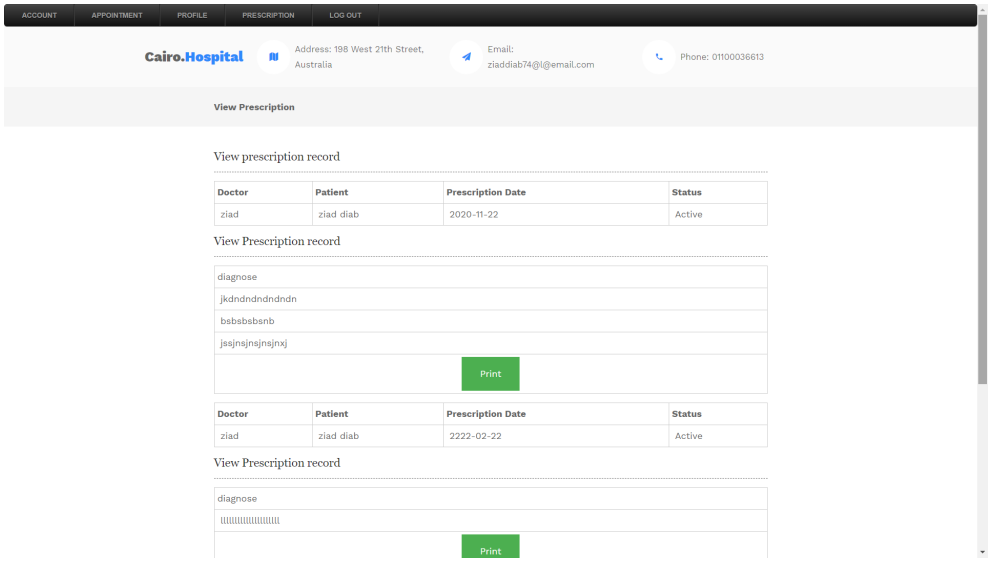

Figure 9: prescription

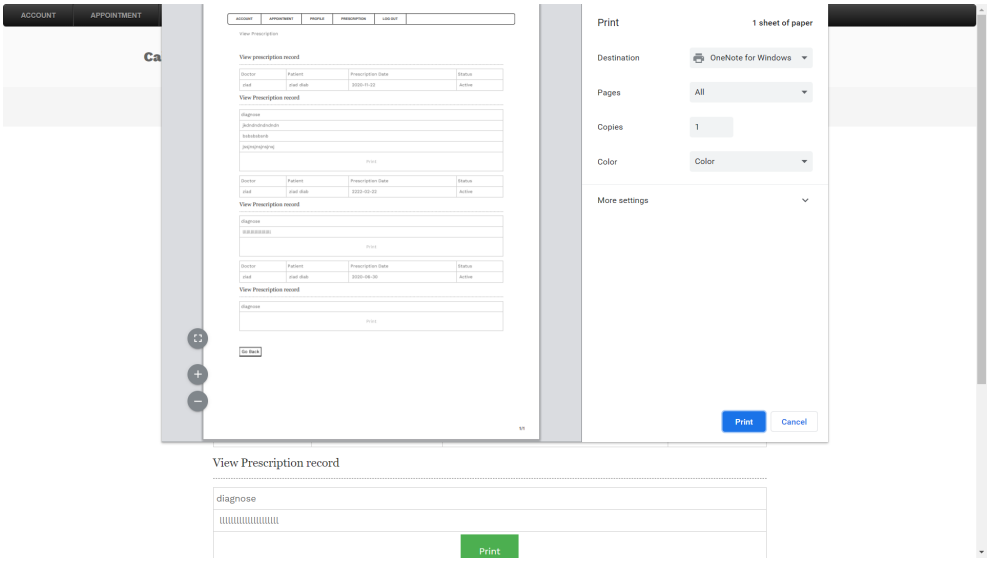

Figure 10: print prescription

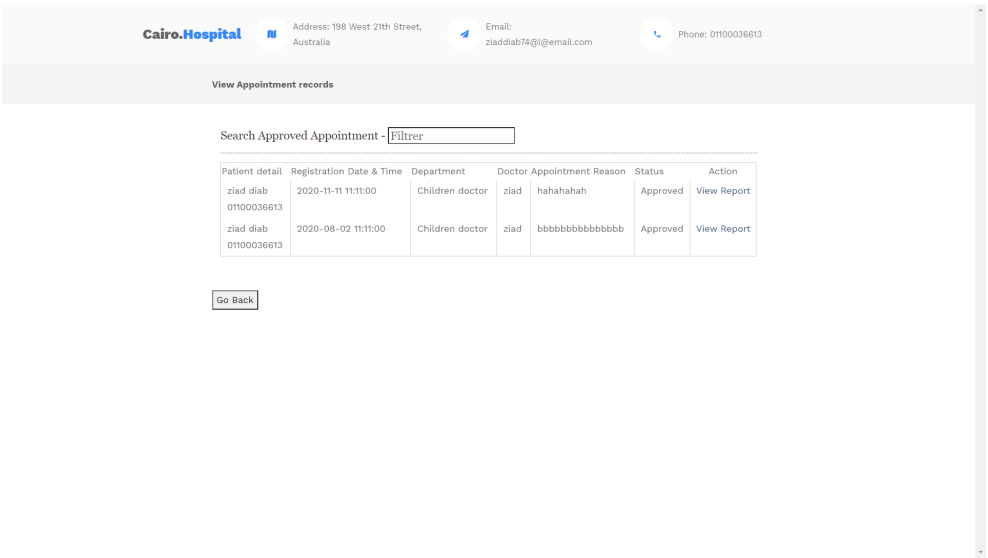

Figure 11: view approved appointments

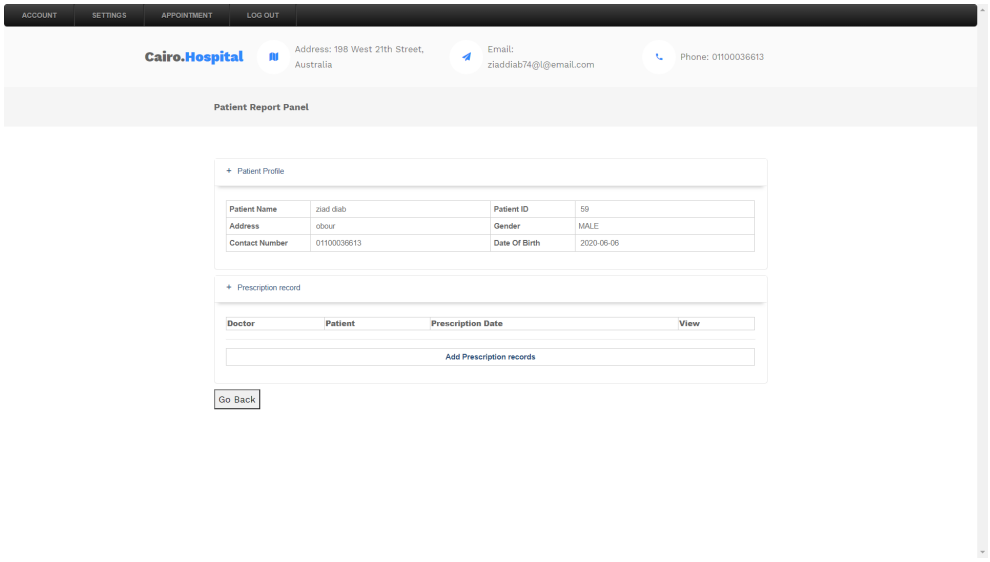

Figure 12: view approved appointments

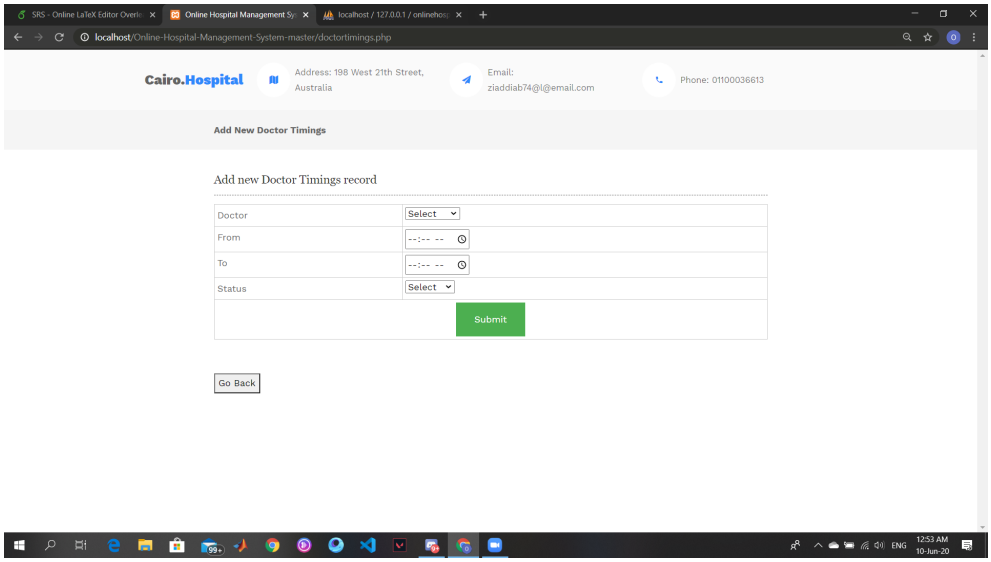

Figure 13: view approved appointments

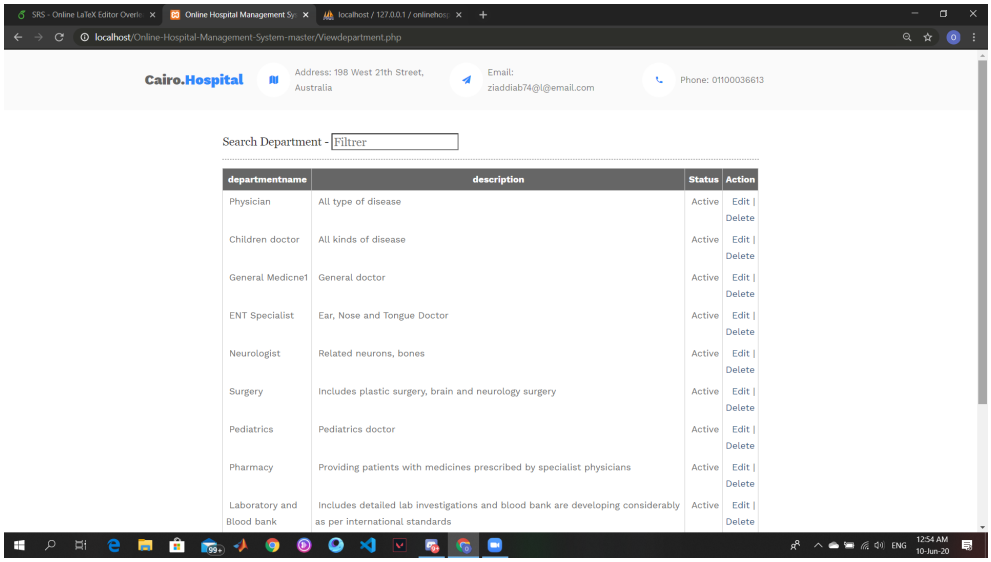

Figure 14: view Departments

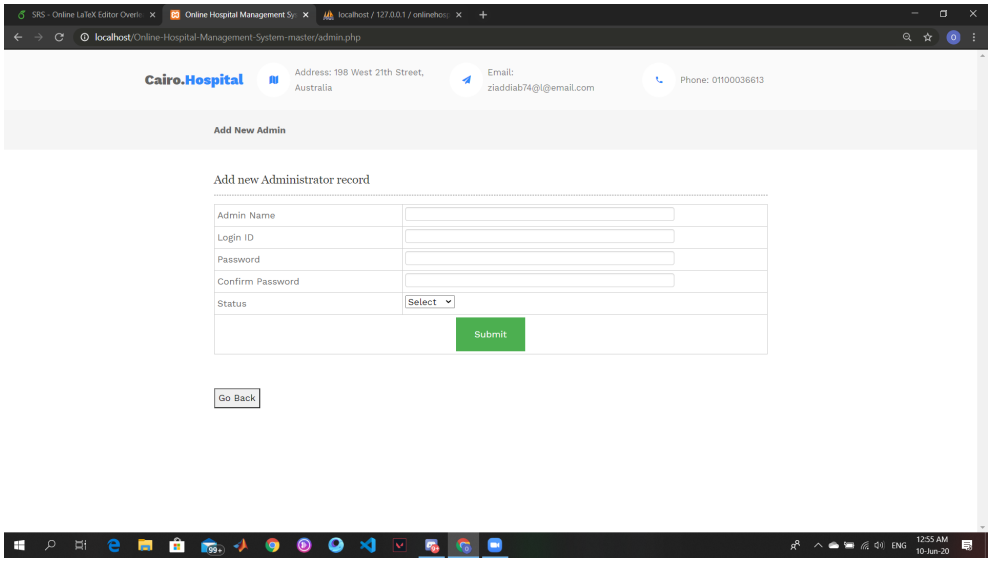

Figure 15: add new admin

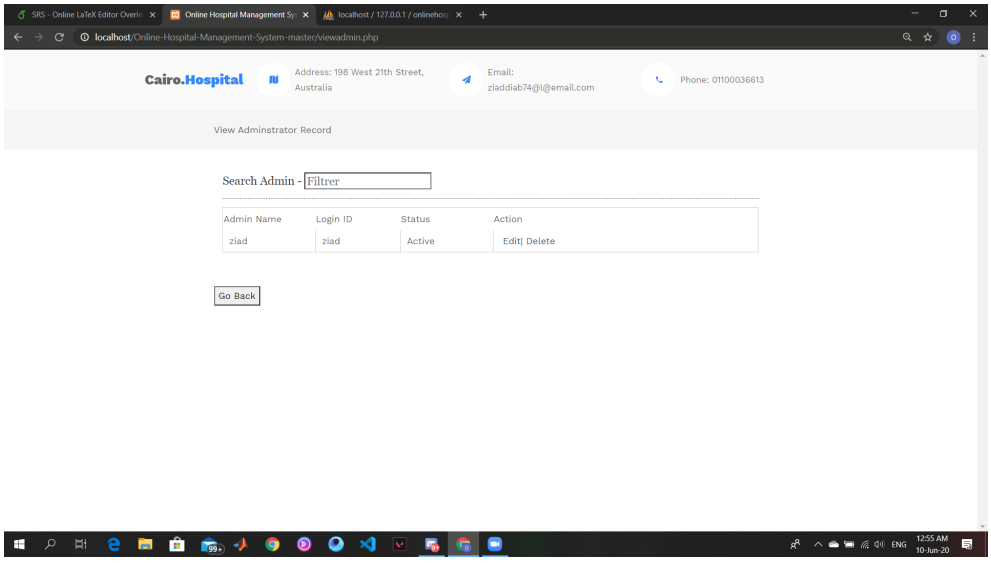

Figure 16: view admins in system

4.2 Hardware Interfaces

this system can run on ordinary computer and a printer is used to print bills and Reports for users from the system

5 Performance Requirements

the system should be using an accepted amount of memory from the client server also it should be reliable in performance and that its processes doesn't take much time.

6 Design Constraints

the system can run on any device .

7 Non-Functional Requirements

7.1 security

admins should have a unique username so that the system knows whose logged in to prevent intruders from having the administrator privileges

7.2 Maintainability

the system can maintain any Data despite how big it is. the code shall be written in an understandable way to help in the ability to maintain ,modify information, update and fix problems of the system,the system shall also provide capability to back-up the data

7.3 Re-usability

the system can be used again as much as the user and the admin wants without distortion ,the system shall be easy to use so that the user can learn how to use it fast and should be organised in such a way that user errors are minimized.

7.4 Availability

the system should be available at all the time,except a few minutes to maintain the system monthly .

7.5 reliability

the output of information should be accurate

8 Operational Scenarios

click to see the use case diagram in figure 19 [6]

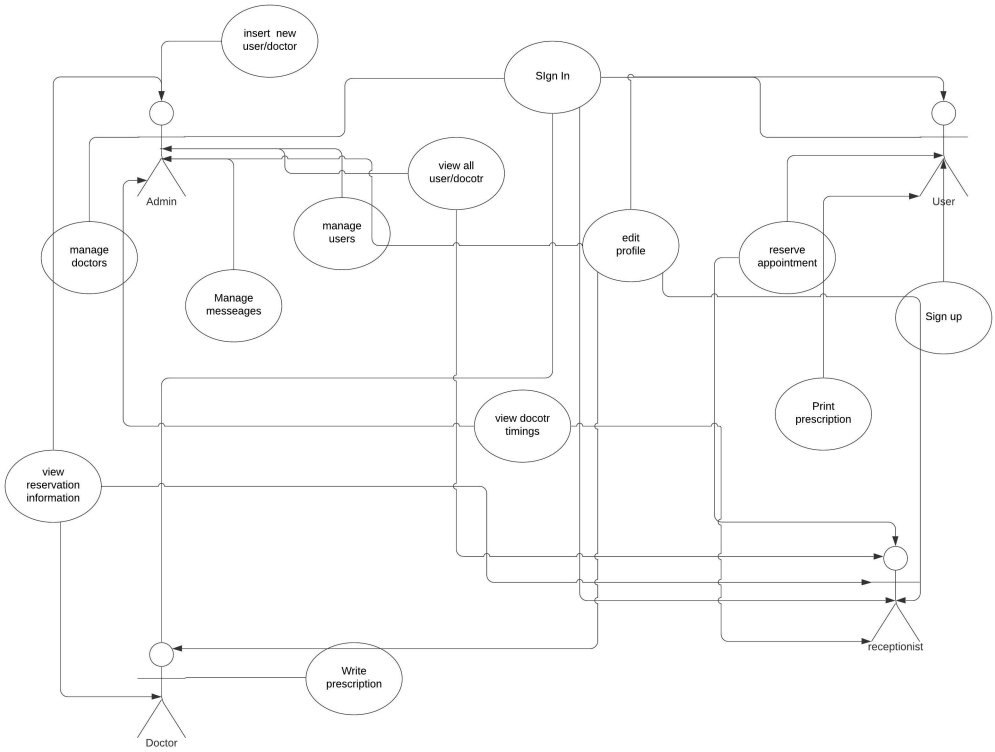

Figure 17: use case

9 Preliminary Schedule Adjusted

click to see the use case diagram in figure 19

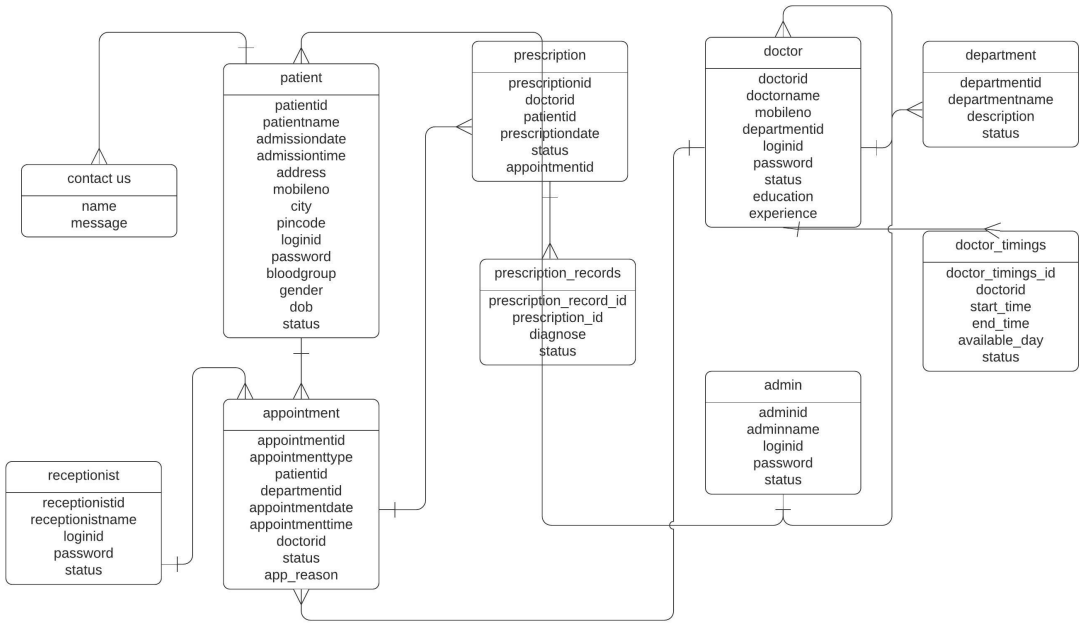

Figure 18: Database Diagram

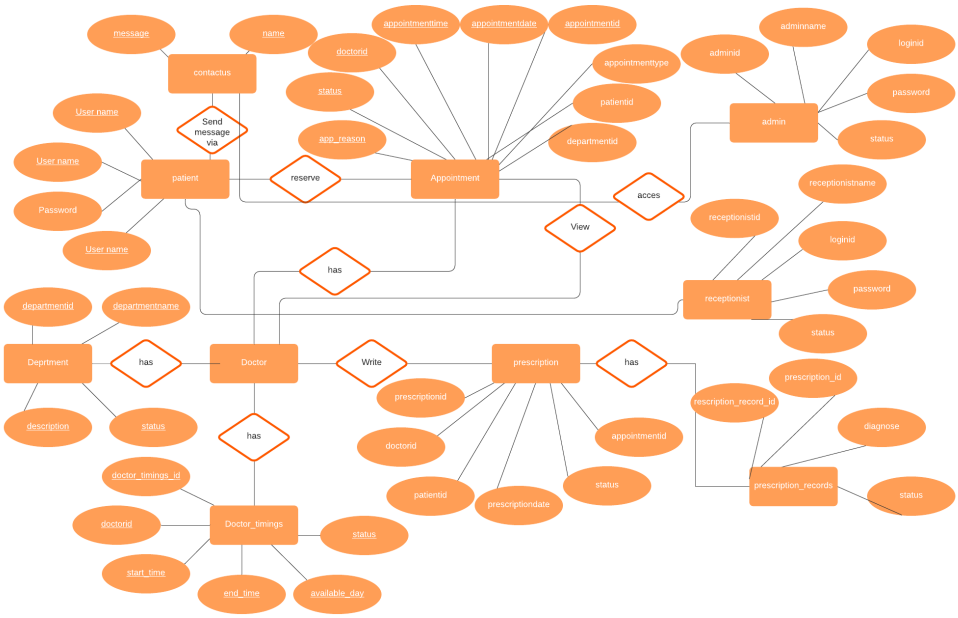

Figure 19: ER Diagram

10 References

References

[1] (2016). saudi german hospital cairo - welcome to sgh - cairo.pp.1-6.

[2] (2019).mayo clinic - mayo clinic.

[3] (2020). iu health.

[4] (2020). miu computer science graduation projects – computer science graduation projects portofolio.

[5] (2020). mount sinai health system - new york city | mount sinai - new york.

[6] (2020).log in.

[7] (2020).northwestern medicine.

[8] Ravi bandakkanavar.(2015). krazytech,pp.1-4.