

INB201: Final Documentation



June 6, 2014

Tutor:

Practical: friday 10 – 12 @ B225

|  |  |
| --- | --- |
| Team Members |  |
| Joseph Salmond | 8823928 |
| Kaan Osmanagaoglu | 8750289 |
| Saud Jehani | 8659486 |
| Syed Ali | 8780595 |
| Abdullah Abdulaziz | 8388083 |

Contents

[Part 1 – Product Overview 3](#_Toc389754250)

[Part 1.1 Features 3](#_Toc389754251)

[Part 1.2 Risks 3](#_Toc389754252)

[Part 2 – Contribution 4](#_Toc389754253)

[Joseph Salmond: 4](#_Toc389754254)

[Kaan Osmanagaoglu: 4](#_Toc389754255)

[Saud Jehani: 4](#_Toc389754256)

[Abdullah Abdulaziz: 4](#_Toc389754257)

[Syed Ali: 4](#_Toc389754258)

[Part 3 - Functional Requirements 5](#_Toc389754259)

[Part 3.1 Function 5](#_Toc389754260)

[Part 3.2 Primary Users 5](#_Toc389754261)

[Part 3.3 Prototype Layout of User Interface 5](#_Toc389754262)

[Part 3.3.1 Saud Jehani 5](#_Toc389754263)

[Part 3.3.2 Syed Ali 6](#_Toc389754264)

[Part 4 – System Documentation / Design Specification 7](#_Toc389754265)

[Part 4.1 Technologies and Tools 7](#_Toc389754266)

[Part 4.2 Developers 7](#_Toc389754267)

[Part 4.3 Use Cases 7](#_Toc389754268)

[Part 5 – Database Design 8](#_Toc389754269)

[Part 6 – Code Quality Assurance 10](#_Toc389754270)

[Part 7 – Acceptance Test Plans 11](#_Toc389754271)

[Part 7.1 User Interface Acceptance Tests 11](#_Toc389754272)

[Part 7.2 Functional Requirements Acceptance Test 11](#_Toc389754273)

[Part 8 – Operations documentation 11](#_Toc389754274)

[Part 9 – User Guides 11](#_Toc389754275)

[Part 9.1 Login screen 11](#_Toc389754276)

[Part 9.2 System Administrator 12](#_Toc389754277)

[Part 9.3 Staff members 13](#_Toc389754278)

[Part 9.3.1 Doctors 13](#_Toc389754279)

[Part 9.3.2 Nurses 14](#_Toc389754280)

[Part 9.3.3 Receptionists 16](#_Toc389754281)

[Part 9.3.4 Hospital Admin 18](#_Toc389754282)

[Part 10 – Sprint backlogs, burn down charts. 19](#_Toc389754283)

[Part 11 – Misc. 21](#_Toc389754284)

[Part 11.1: Source tree, Primary Git Management Software. 21](#_Toc389754285)

[Part 11.2 Github Contribution Chart 22](#_Toc389754286)

[Part 11.3 Git Log 22](#_Toc389754287)

# Part 1 – Product Overview

The Project is to design and implement a digital hospital management system. Thus it must contain feature that enhance the ability of hospital staff to do their job. The product will be user friendly and simple to use.

## Part 1.1 Features

Speed of lookup:

Able to lookup results and patient results in an immediate fashion.

Up to date Results:

The system enables near instantaneous distribution of data across the hospital infrastructure.

Security:

Strict user control and sha256 hashing of passwords with their user name as the salt.

## Part 1.2 Risks

Technical:

One of the risks is assuming that the hospital is running windows with the required .net framework installed.

Concurrency for any database system is always a challenge an there was no tests to indicate how this will affect the user base.

# Part 2 – Contribution

## Joseph Salmond:

Created MidleWare and the code required to traverse the forms, recreated database week 12 due to inefficiencies in the original database, created the pdf’s using PdfSharp, created the logic for authentication and registration, Source Control management and updating and documentation.

## Kaan Osmanagaoglu:

Data-Access Layer:

Design and creation of the database, later moved onto creation and testing of the sql needed for the interaction of the forms.

## Saud Jehani:

Presentation Layer:

Creation of the user interface.

## Abdullah Abdulaziz:

Data-Access Layer:

Design and creation of the database, later moved onto user interface and creation of the user interaction diagram.

## Syed Ali:

Presentation Layer:

Creation of the user interface.

Signed:

# Part 3 - Functional Requirements

This defines what the system does. This includes user roles, typical and significant user stories, and may include prototype layouts of essential user interfaces.

## Part 3.1 Function

The system is designed to manage and manipulate patient data for a digital hospital system.

## Part 3.2 Primary Users

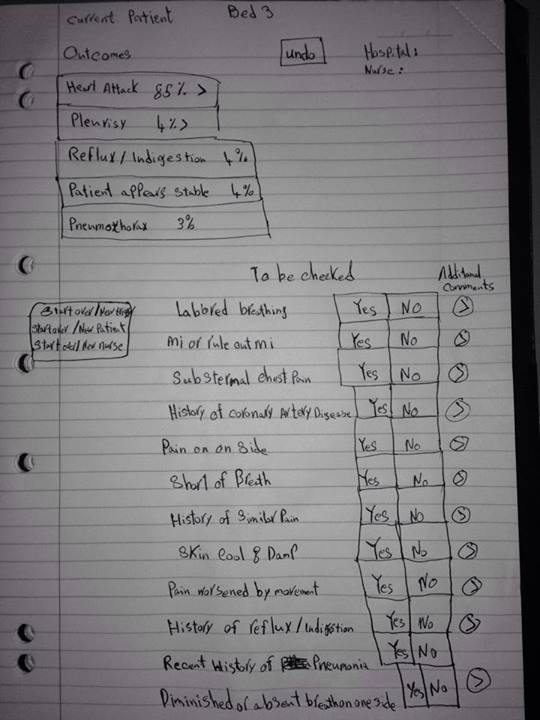
Hospital Staff

* Doctors
* Nurses
* Hospital/System Administrators
* Medical Technicians

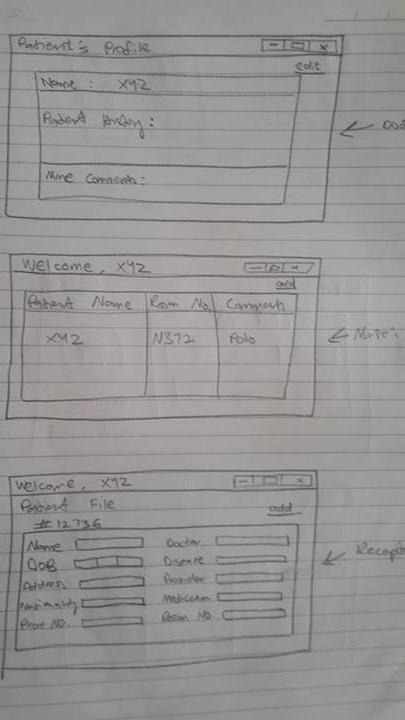
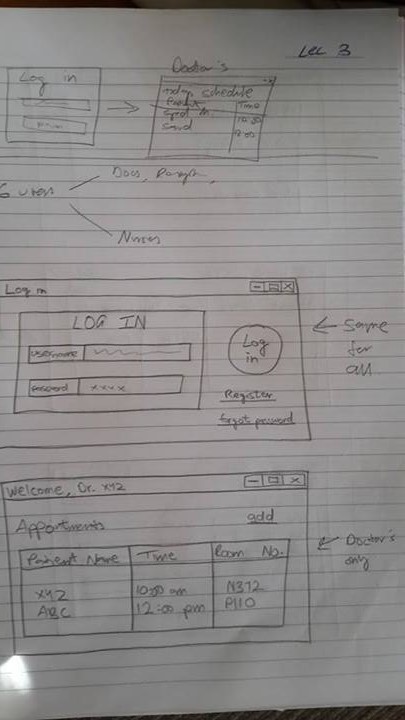
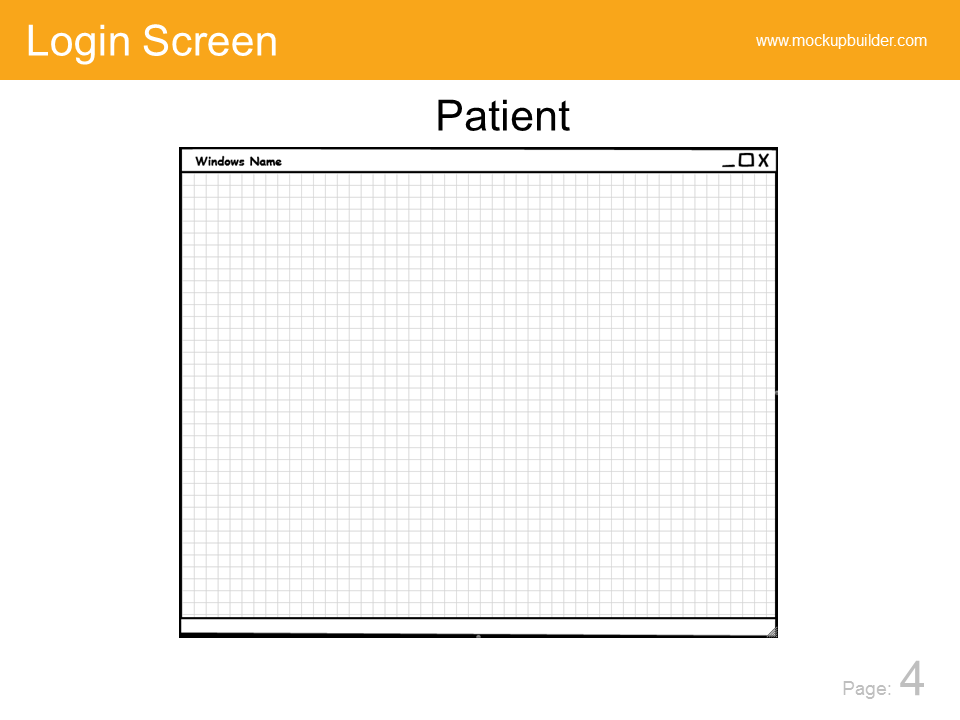
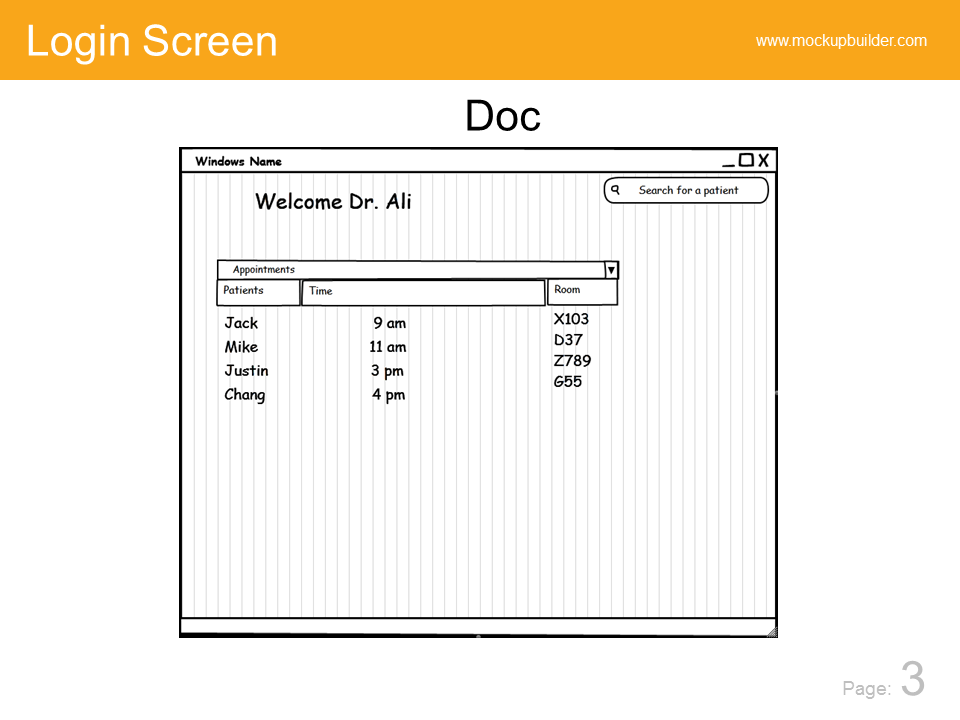
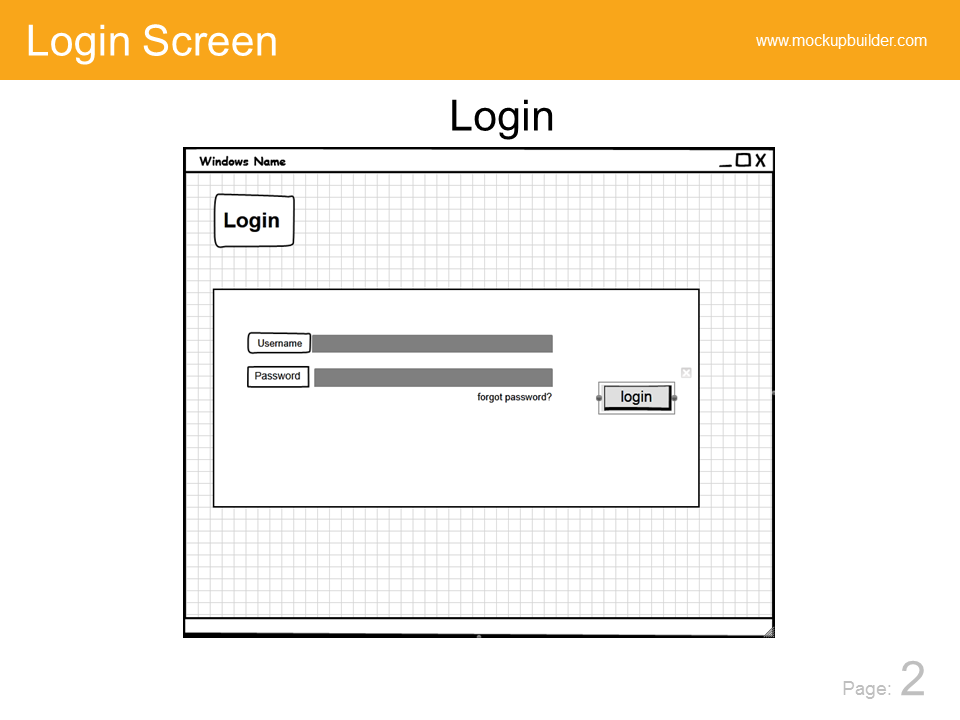
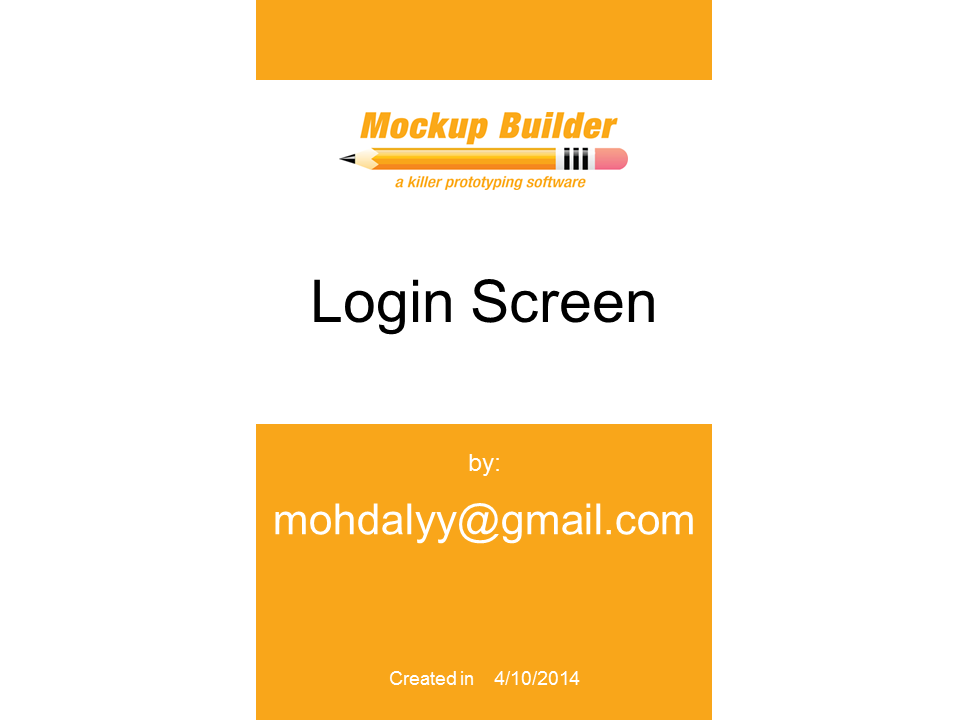
## Part 3.3 Prototype Layout of User Interface

User interface Designed by Saud and Syed, prototypes designs delivered by.

### Part 3.3.1 Saud Jehani



### Part 3.3.2 Syed Ali



# Part 4 – System Documentation / Design Specification

## Part 4.1 Technologies and Tools

The Technologies that were used are:

Visual Studio 2012: Coding and User Interface Creation.

Microsoft Sql Server 2012: For the database creation.

NOTE: due to the fact that initially only two members and eventually only one had Sql server installed the database was added to Visual studio and interaction was done on Visual Studio’s cut down version of Sql server.

PdfSharp : was a 3rd party assembly that managed the creation of pdf’s

## Part 4.2 Developers

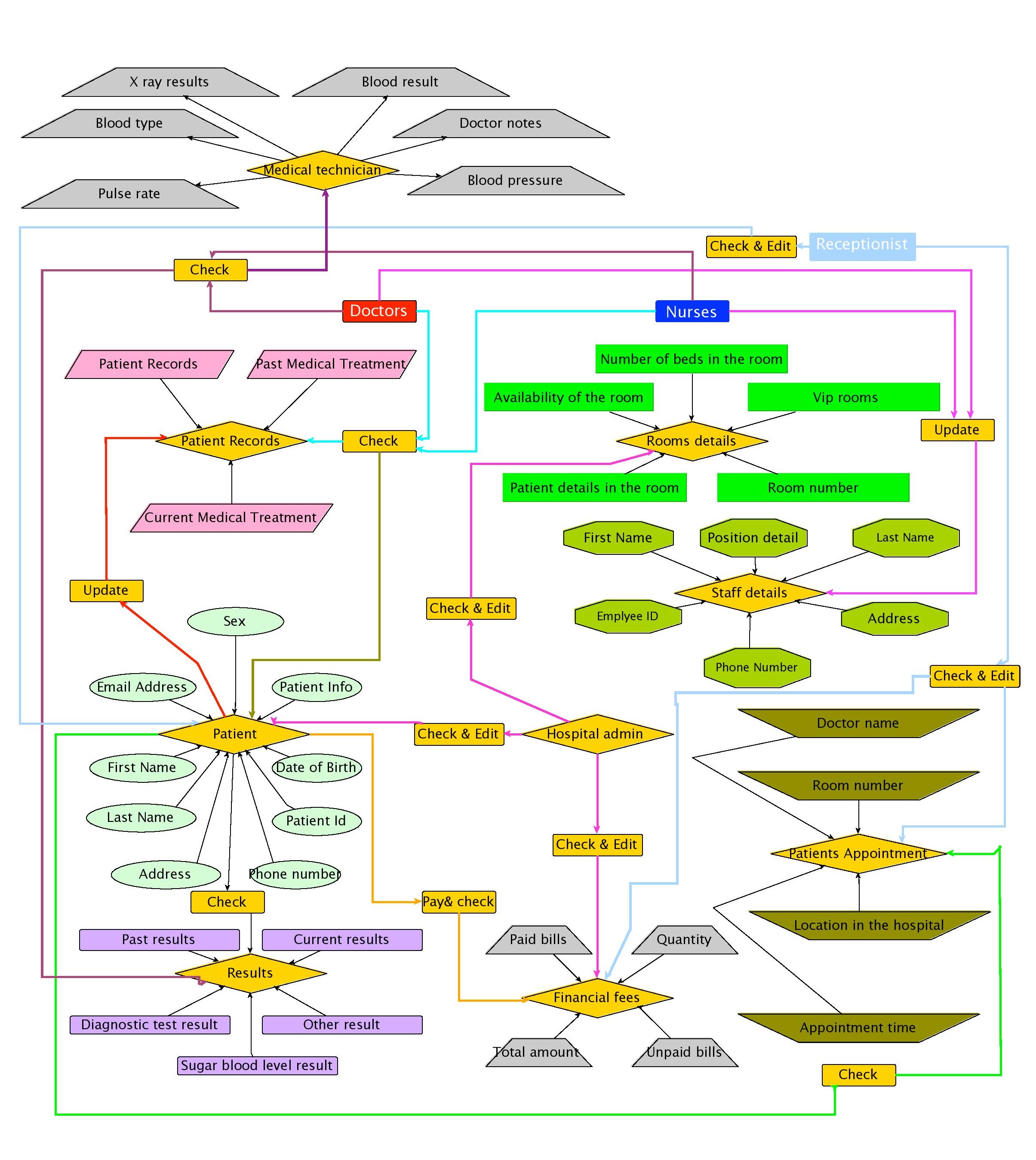
The team will consist of One Member Working on the Business Layer (Middle Ware), Two Members Working on the Presentation Layer and Two Members Working on the Data-Access Layer.

Should provide an overview of the system

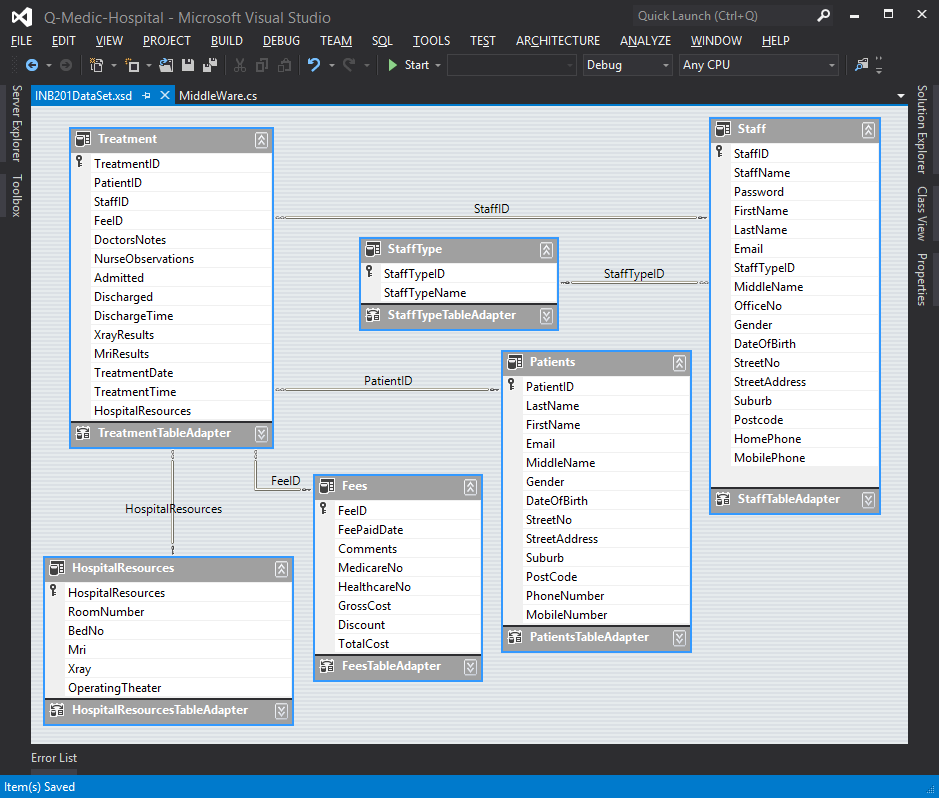
Describe the technologies and tools used to build the system

## Part 4.3 Use Cases

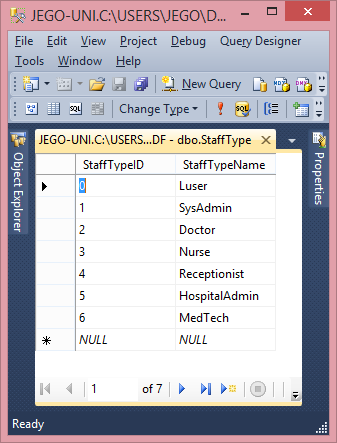
@Author Abdullah



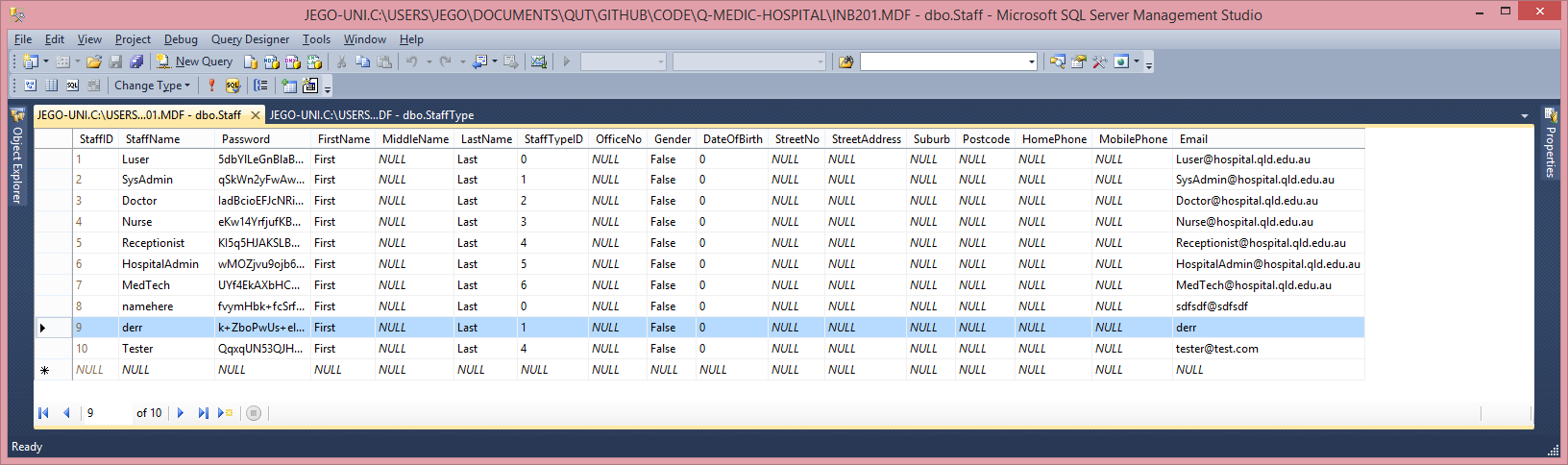
# Part 5 – Database Design



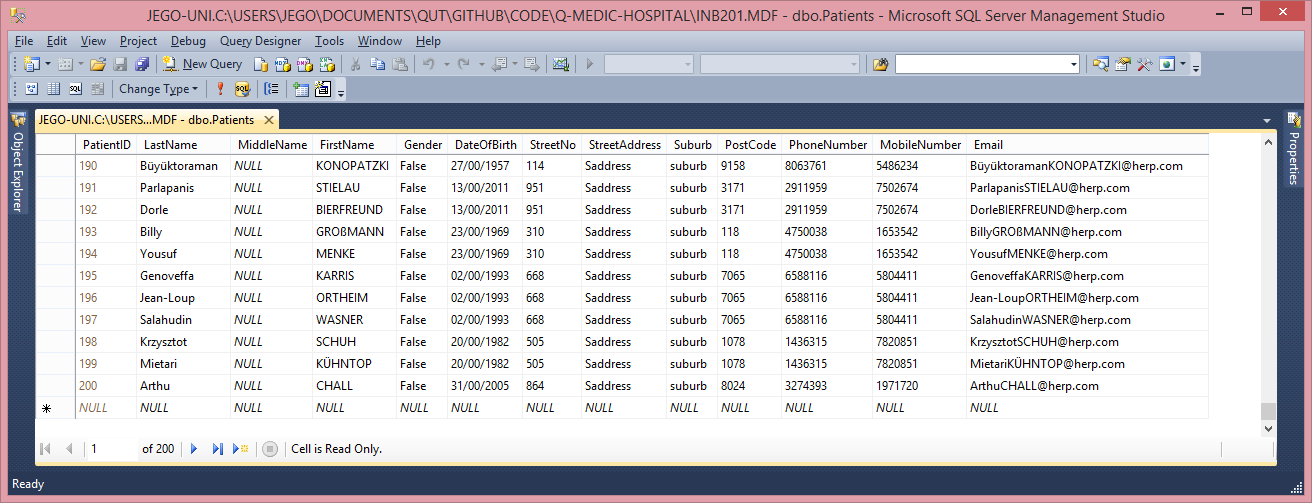
Relationships in the Database



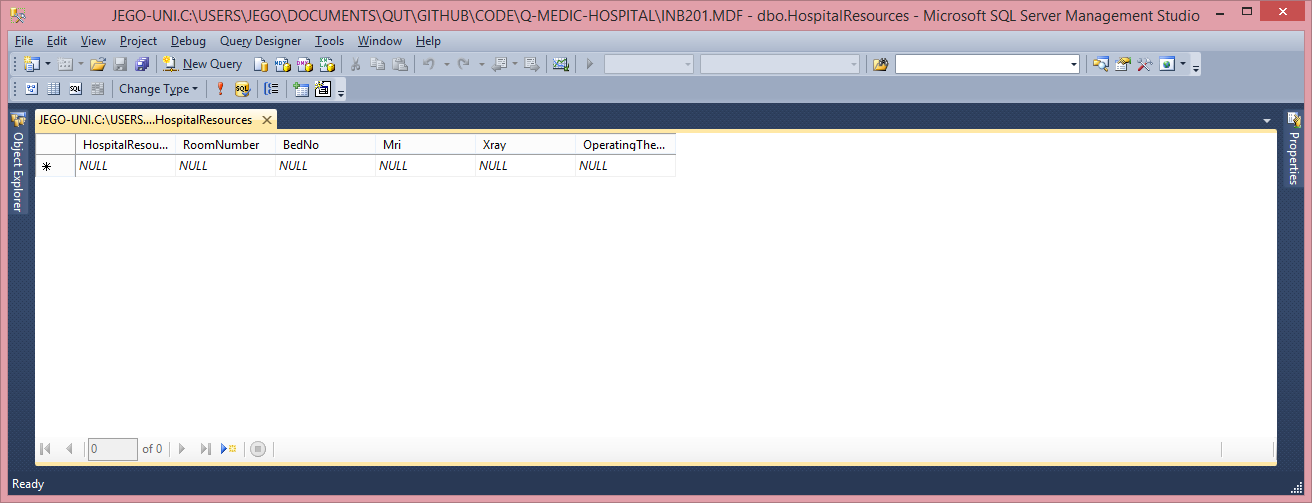
StaffType : used in mapping the user to a particular user type thus enabling separation of duties.



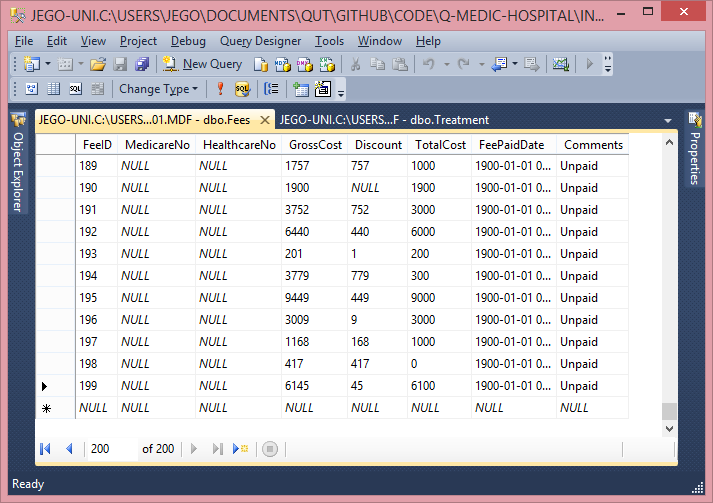
Staff : holds the information about the staff, which staff class they belong to.



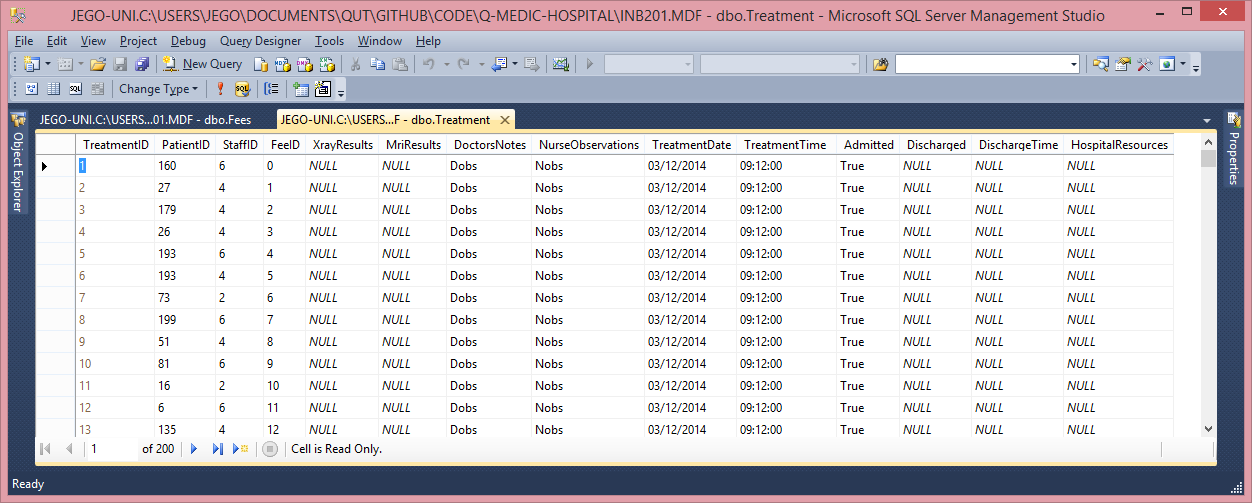
Patients : used for managing the patient data.



HospitalResources : used for managing the resources of the hospital. Examples are hospital bed allocation and operating theatre times. NOTE: not yet implemented or tested due to forms being incomplete.



Fees : this is the “money” table where all the financials are stored.



Treatment : the most critical of the table this is the table that joins all the others together. It manages which treatment it is what patient is having the treatment which staff member is overseeing the treatment which fee is associated with the treatment the doctors and nurse notes, and which Hospital resources are used via the HospitalResource table.

# Part 6 – Code Quality Assurance

The Coding standard was the one INB270 used in its course. The Majority of the coding and code reviews where done by Joseph Salmond.

# Part 7 – Acceptance Test Plans

## Part 7.1 User Interface Acceptance Tests

User interface was poorly designed and implemented, the interface is incomplete and sometimes counter intuitive. Thus the interface would fail to be easy to use by the user base. The testing was done “by eye” and troubleshooting various issues.

## Part 7.2 Functional Requirements Acceptance Test

The project did not arrive at completion due to the fact that the interaction with the database (SQL) was not implemented. The functions that where implemented was user registration and login as well as exporting to pdf’s all three complete by Joseph. And able to view some patient details in the patient details form completed by Kaan.

The tests for the User registration and login where that the use can only login with their username and password. The pdf’s where tested by exporting many different datasets and viewing them within adobe reader and Foxit reader.

# Part 8 – Operations documentation

Within the install directory, you may run the program from the primary executable (Q-Medic GUI.exe) or install it as an application (Q-Medic GUI.application). Also there is a sql script to recreate/create the database if required (Hospital DataBase Creation.sql).

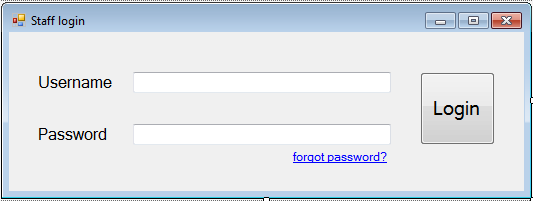
# Part 9 – User Guides

@Author Kaan

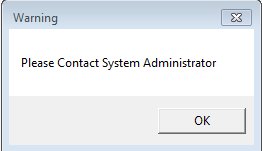
First time usage of the interface begins with the Login screen. The hospital staff are first granted access by the hospital administrator directly and system administrator indirectly. With first usage of the program, system administrator is the only user that has an access to the system. If you’re another user you may start from step 3.

## Part 9.1 Login screen

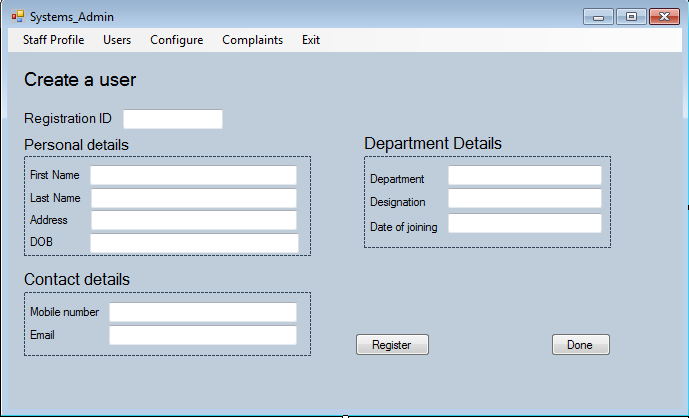
.as shown below, provides the users with the username and password inputs. The passwords are protected by salted SHA256 hashes to provide an extra layer of security to user experience.



The users may click on Forgot password link, which then returns an information alert to contact the system administrator.



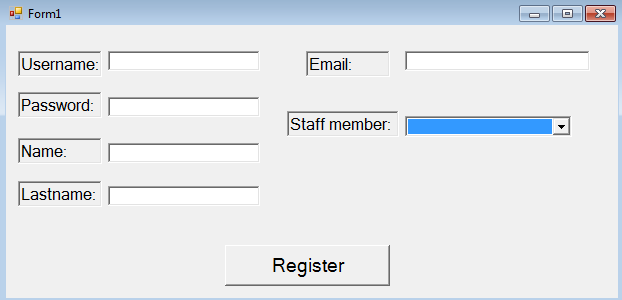
## Part 9.2 System Administrator

For unregistered staff members, after the login screen, the system administrator can create an account by first filling in the form returned after login and then clicking “Register” before “Done”.

After the required privileges are added by the system administrator, the users may now navigate the system by logging in. The MiddleWare makes sure the relevant forms to the titles of the logged in users are returned, such as Doctor form is returned to the Doctors who are logged in. This way the staff members are synchronized with the main database.

On top panel, system administrator can click and see the details of the Staff Profile, according to their title, with different details. Next to this we have Users, which should include a list of registered users with their titles. Configure next to Users is available in order to modify the staff members’ details and “Complaints” to get access to various different complaints uploaded for various departments of either “Doctors”, “Nurses” or “Receptionists”.

If the system administrator wanted to add a new staff member, input the details and clicked on Register, they may now proceed to “Done” to finish their work and they may login again. But the process of registration opens a new form for the system administrator to fill in as seen below;



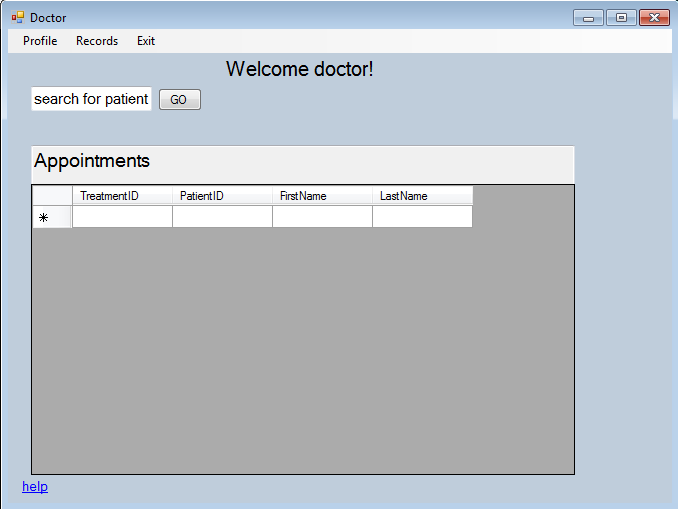
When the system administrator selects the staff title, email, name, lastname, username and password (which would be hashed in the database) then they click “**Register**” and “**Done**” when they’re completely finished. If they may wish to see the complaints from the users, staff members or anybody in general, they may click the “**Complaints**” from the top panel which then takes the system administrators to the list of current available complaints to attend to.

## Part 9.3 Staff members

The registered staff members may login to the system with the access granted by their system administrator. This step is further divided to provide further details of different staff members and their forms.

### Part 9.3.1 Doctors

The doctors have access to an initial “Appointments” table which includes the treatments and their IDs, patients and their IDs and the patient details. If the doctor is looking for a particular patient, they may input the name, lastname, initials, part of the firstname or lastname in order to find the patient/treatment file they are looking for. After they input their search element, clicking on “**Go**” will be sufficient for the database to return them the result(s) they wish to see.



The doctors also have the option to view certain profiles. They have access to the patients’ profiles and their records with further details. On the top left corner, if the “**Profile**” is clicked, a form is returned as below:

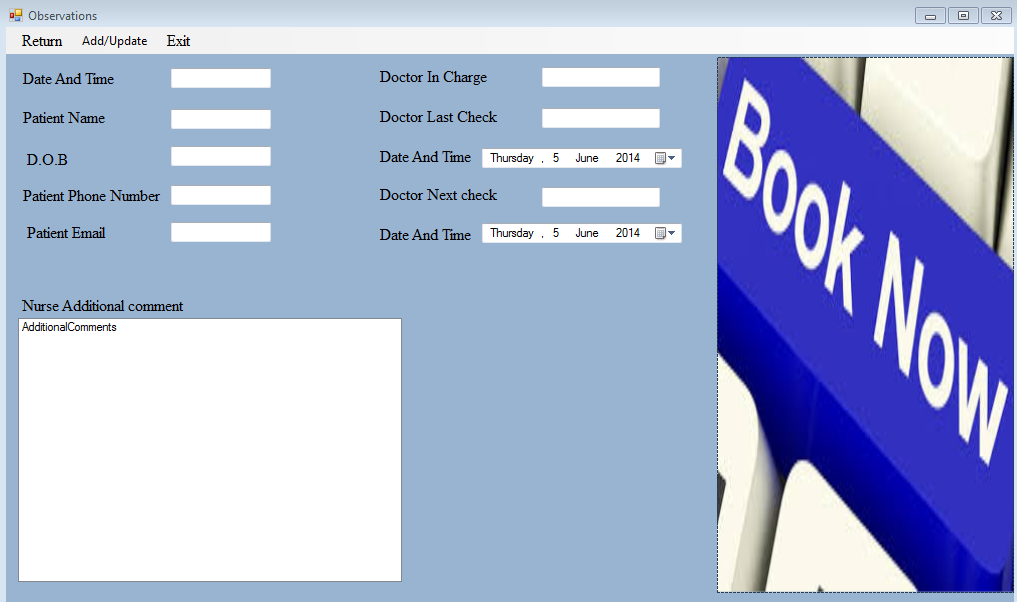


In this section the doctor then must click on the top panel “**UPDATE**” button to update the patient details they searched for in the previous form. If they wish to modify the details of the patient and add their assigned “**Room number**”s, preferred doctors, nurses or past medical diagnosis, current medical diagnosis, they first modify the input and then click “**Fill**” in order to update their database file instantly. The doctors may then finish and leave by clicking “**Exit**” from top panel.

### Part 9.3.2 Nurses

For nurses who are logged in, the working form is displayed as seen in the example below. The nurses are able to type in the hospital details for the patients that must be transferred to another hospital together with the name of the nurse who is responsible from the transfer. Main nurse function is being able to select a date from the calendar table and returning the scheduled observations for that date.

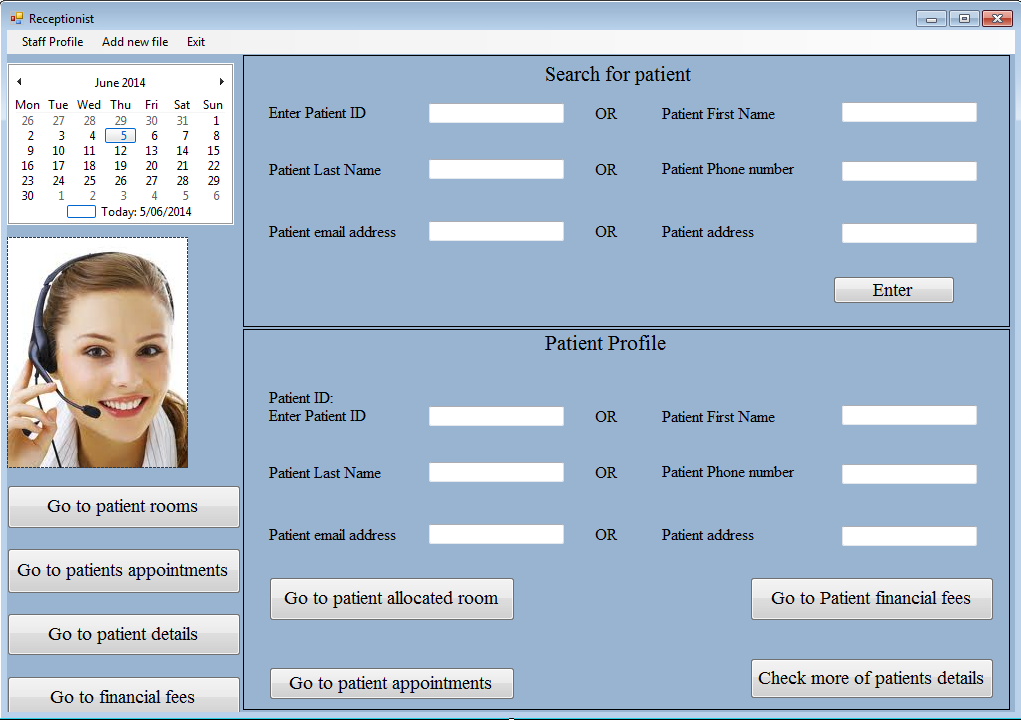


Further in the observations screen (seen below), the nurses may include the details of their current observation for certain doctors to see as well, depending on which doctor would be in charge of the patient. They can modify the doctors names for certain checks, previous or next, including the date and time for the doctor so that they are alerted before that time through the system. They may also add additional comments and they are saved into the database. If the nurse wants to update an existing observation, they simply click “**Add/Update**” after filling in the new details, or adding the doctor name for the next check, or possibly the previous one. 

They may now click “**Exit**” after they have finished.

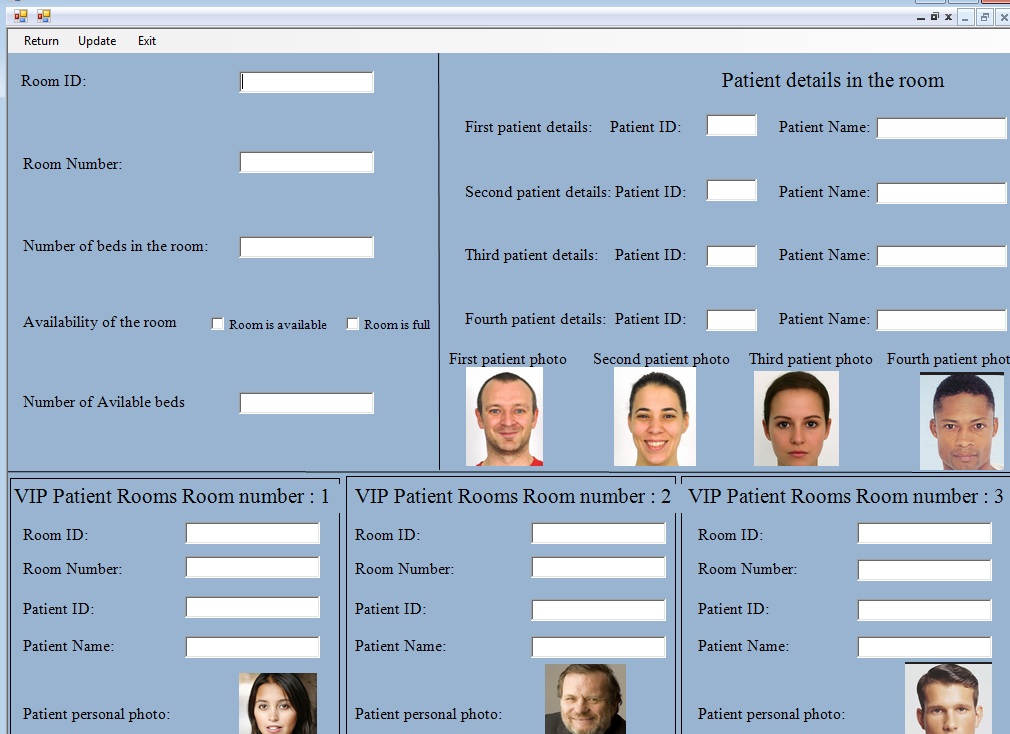
### Part 9.3.3 Receptionists

Below we can see the receptionist interface. Receptionists have access to patient files, patient profiles, patient rooms, patient appointments, patient details and the financial fees. Their main function is the arrangements of financial fees before they proceed to diagnosis/treatment phase after doctors/nurses observations.



The receptionist may type in either the patient ID, first name, last name, phone number, part of the address or email address in order to search for a particular patient and they may click “**Enter**” to return the patients which match exactly with the search input. If they were looking for the patient profiles related to the financial fees or the room they were allocated in, they would be required to type into the lower half screen form of the interface and then click the element they need to see, such as appointment, allocated room or fees.

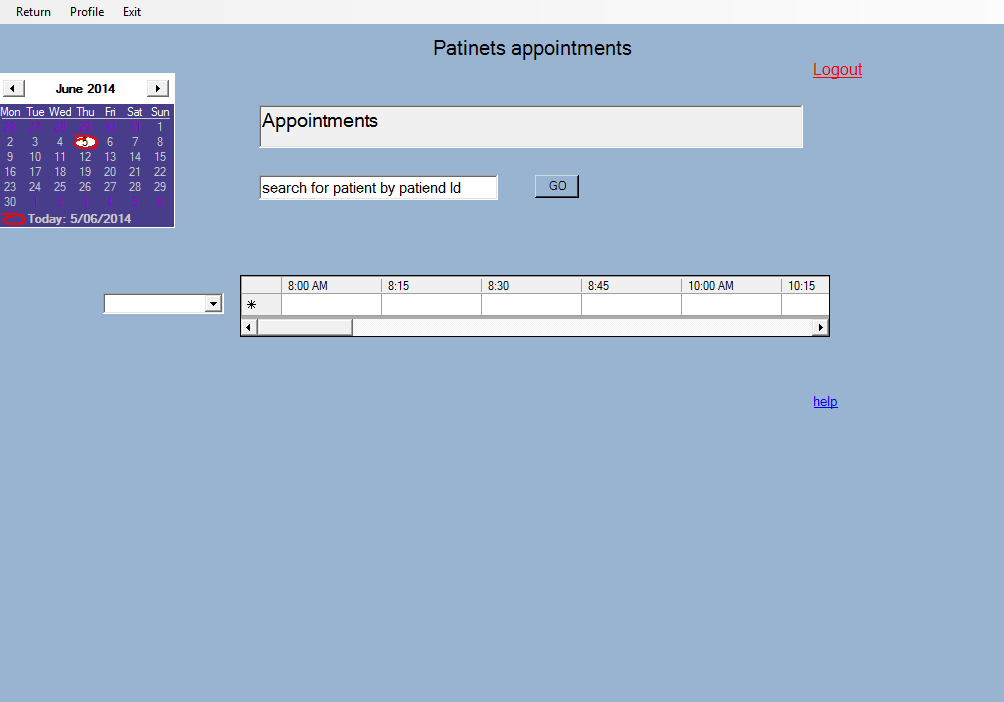
Inside the patient rooms section, the receptionists have access to the form as seen below.



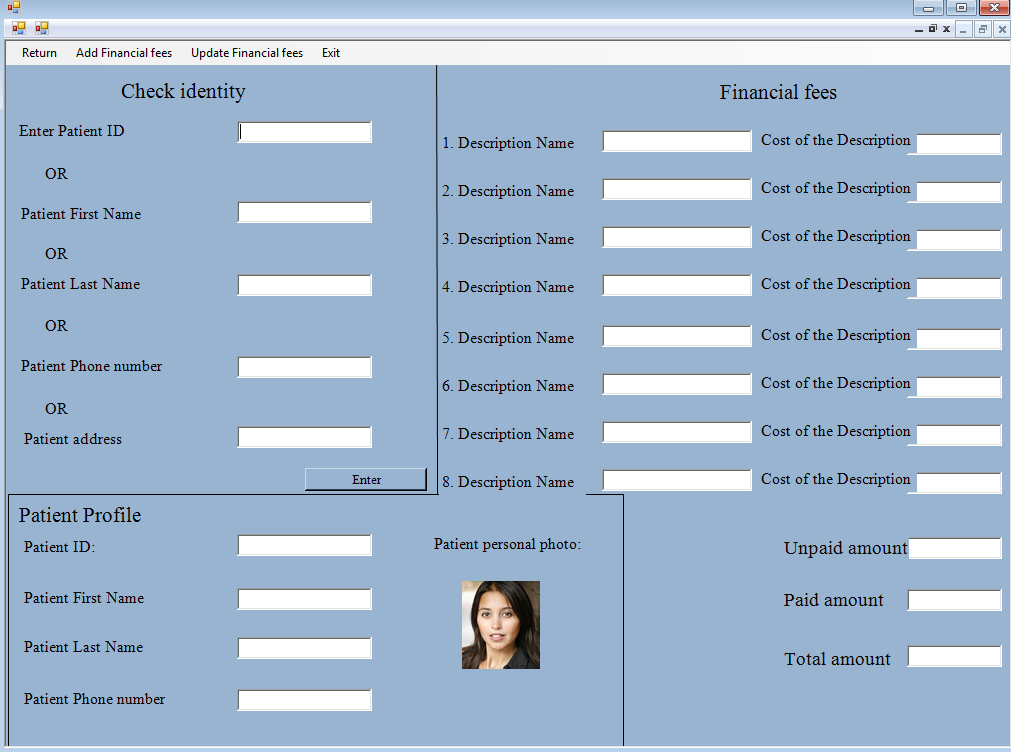
This part of their interface, if they have previously input a search filter in their main form, returns the allocated room of the patient of particular treatment. They may also check a particular room by typing into the necessary field **only** in the main page of receptionist interface. This page includes various information about the room, it’s ID, number, availability of the room and the beds, allocated current patients in room and possible VIP options that may be present in certain rooms. They may also use the “**Update**” from the top panel in order to update the details of a certain room or the possible changes with certain rooms or VIP rooms and the way the patients are allocated in.

When the user is done, they may now click **Return** to go back to the main interface.

The next functionality receptionists have is the patients appointments. If they have previously included the input in main form, they may now return the related appointments to the patient they were looking for or all the appointments that are relevant to the room number they have input, or all the appointments for the timeframe they have input. They may return with the “**Return**” button when they are finished.



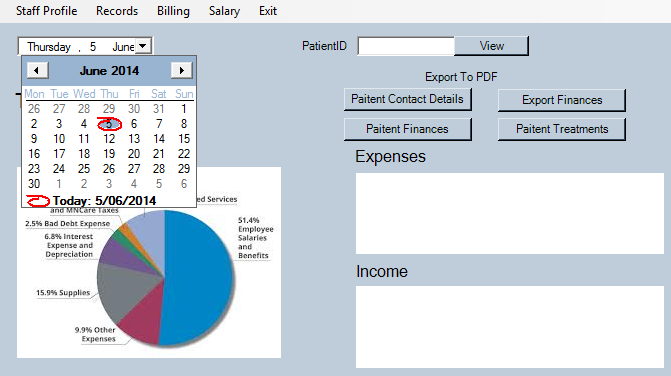
The next functionality included with the Receptionists’ interface is a common functionality of the doctors, which enable them to see the patients details, that can be seen in 3-1) section. All the further details of certain patients are shown in this section.

Lastly, they have access to the financial fees of a certain treatment that is associated with the patient that is searched for in their main interface. This interface has a small summary screen for the patients profile, and a list of the financial fees they have been charged for until now and from now on.

The way this section of their interface works is, they first need to type in at least one input for the patient they are looking for and then click “**Update Financial Fees**” from the top panel to return the financial fees results for the patient, if there’s any. If the receptionist wants to add new financial fees for the patient, they simply need to Update financial fees first, then input the fees they may wish to charge, then click on “**Add financial fees**” in order to complete the invoice for saving into the database. When they are done, they may now wish to return back to the main interface or logout by “**Return**” or “**Logout**”.

### Part 9.3.4 Hospital Admin

The hospital admin has a unique, innovative interface with a chart that includes general expenses and how it’s divided between which parts of the hospital’s income and expanses, so that they can follow the monthly revenues of the hospital and possibly provide extra beds, patient rooms, building extensions or even extra services.



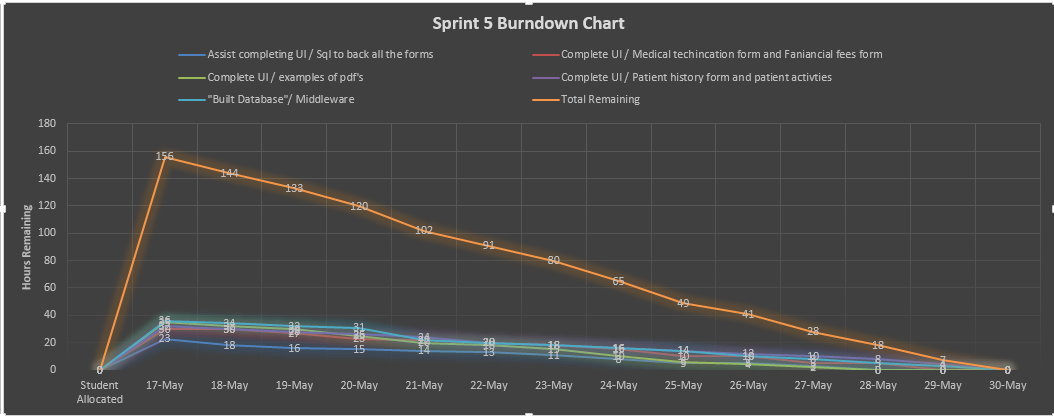
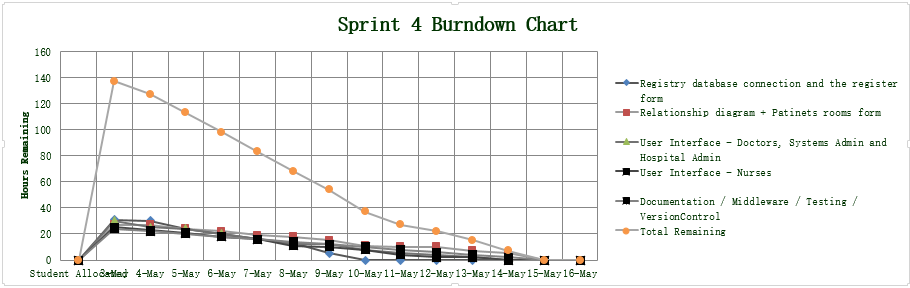
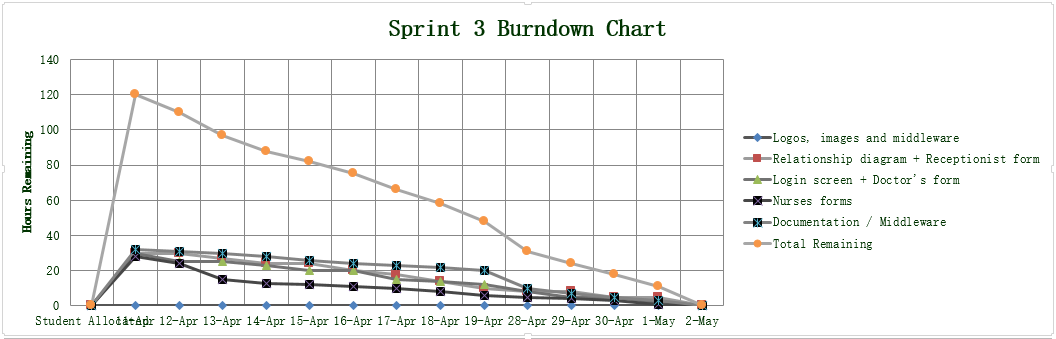
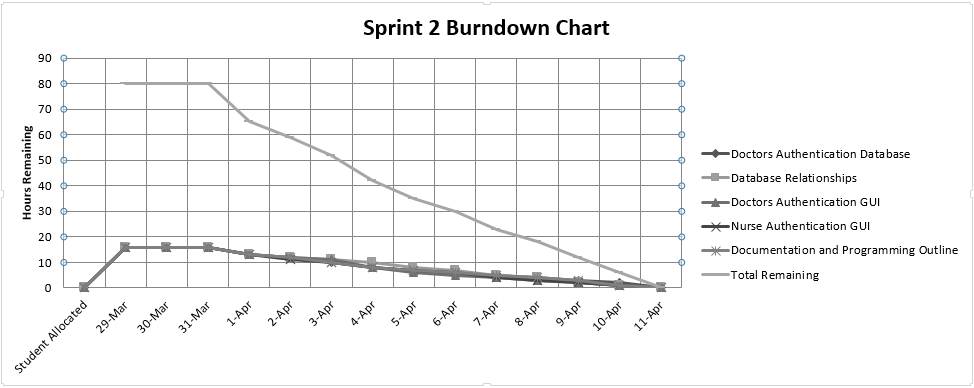
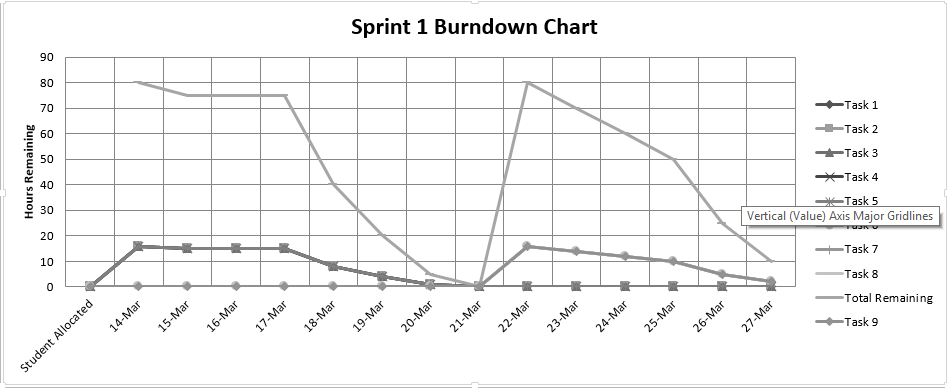
Seen above in the example, the hospital admin can type in the “**PatientID**” into the box and click on “**View**” in order to return the patient details interface which was covered in 3-1) and 3-3) sections (Doctors and Receptionists respectively) of the user guide. But as an extra functionality, hospital admins may wish to see the details of any patients together with the details of the staff members so that they can pass it on to the system administrator to add, modify or remove access from the system depending on how the hospital is doing.

The hospital admin can input the patientID not only for seeing the patient details, but they may also export;

* Finances
* Patient Treatments
* Patient Finances
* Patient Contact Details

# Part 10 – Sprint backlogs, burn down charts.

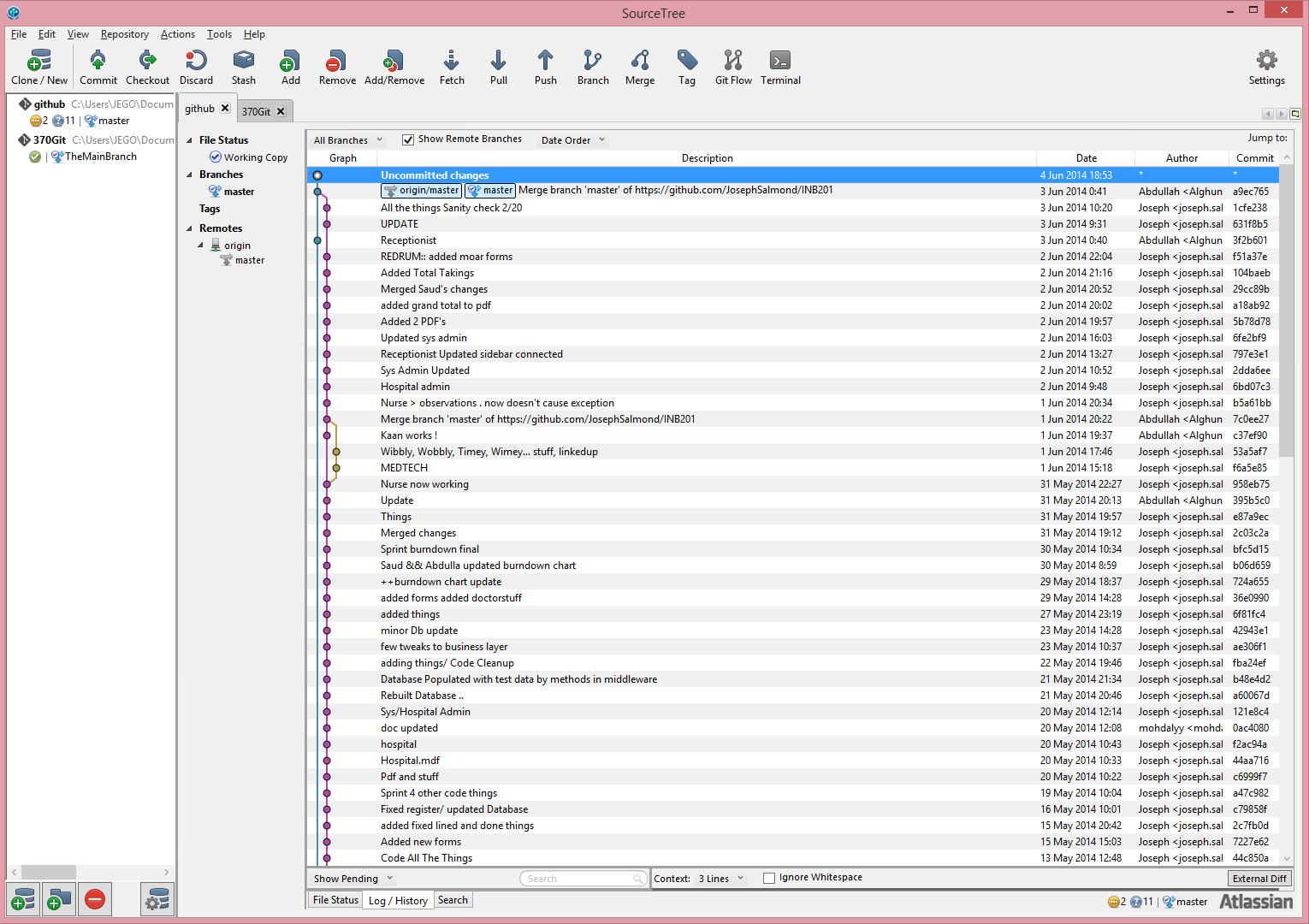
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Sprints | UserStory: Tasks Assigned | | | | |
| Member: | Kaan Osmanagaoglu | Saud Jehani | Abdullah Abdulaziz | Syed Ali | Joseph Salmond |
| 1 | Database:  Schema Tables | User Interface:  Design Draft/ reports/forms | Database: Relationships | User Interface:  Design Draft/forms/reports | Documentation: |
| 2 | Doctors Authentication/Database | Nurse Authentication/Gui | Database Relationships | Nurse Authentication/GUI | Documentation/ Programming |
| 3 | Logos, images and middleware | Nurse forms | Relationship diagram / Receptionist form | Login screen / Doctors Form | Documentation / Middleware |
| 4 | Registry database connection and the register form | Relationship diagram + Patients rooms form | User Interface - Doctors, Systems Admin and Hospital Admin | User Interface - Nurses | Documentation / Middleware / Testing / Version Control |
| 5 | Assist completing UI / Sql to back all the forms | Complete UI / Patient history form and patient activties | Complete UI / Medical techincation form and Faniancial fees form | Complete UI / examples of pdf's | "Built Database"/ Middleware |



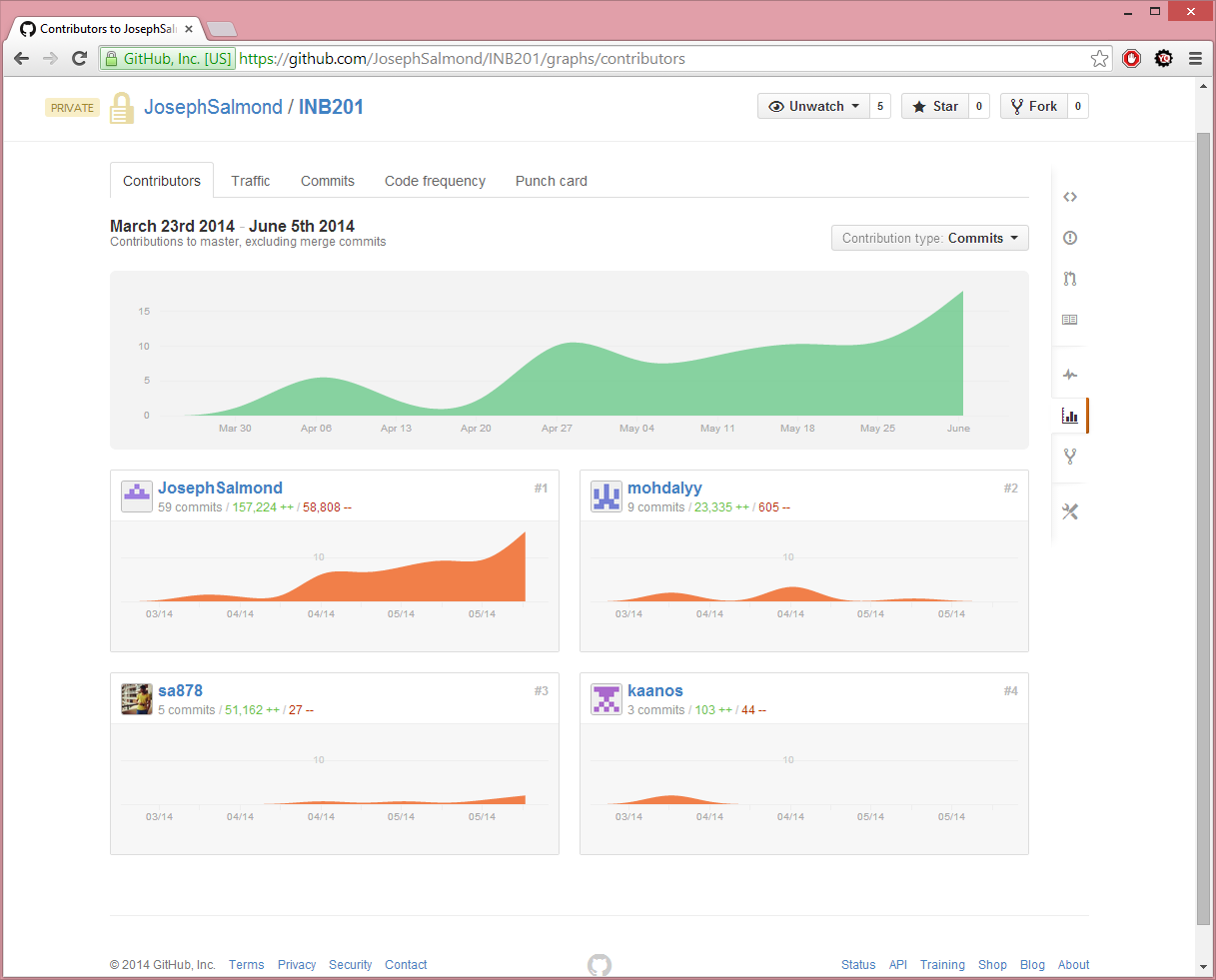
# Part 11 – Misc.

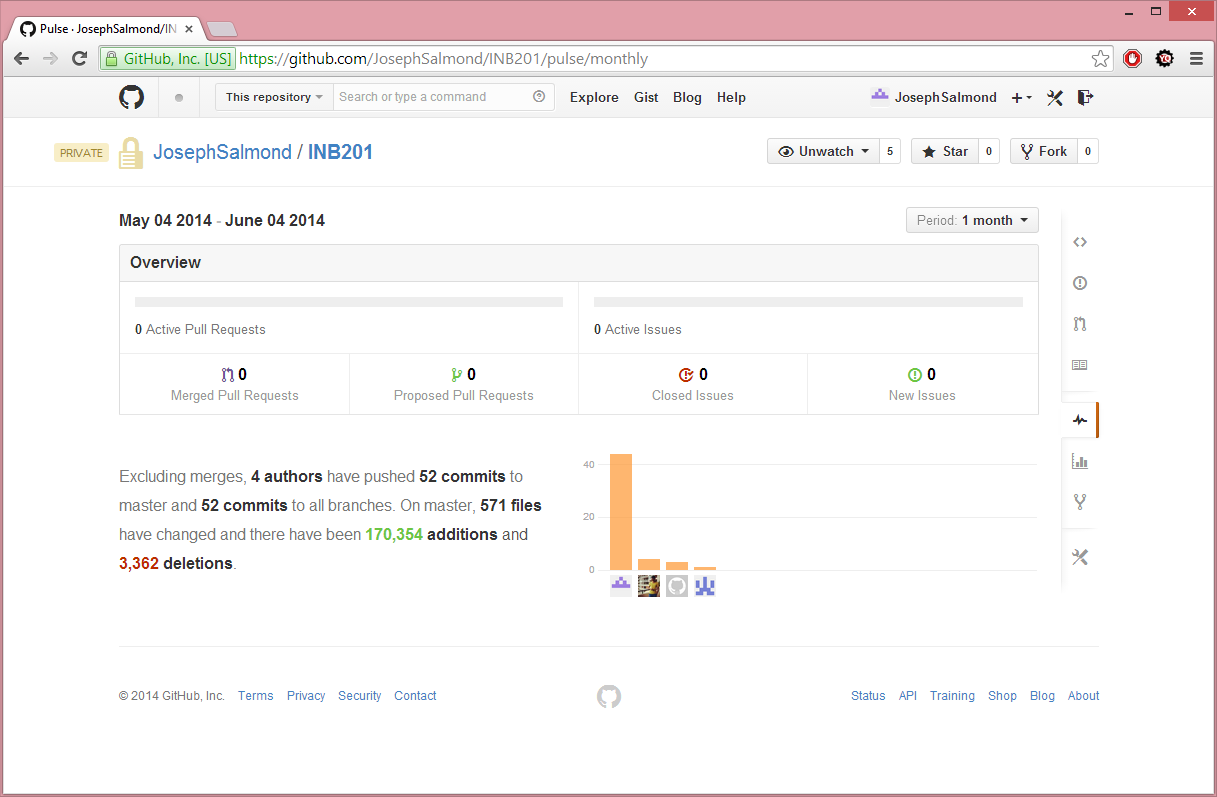
Things that don’t fit into other categories/haven’t been sorted into proper categories.

## Part 11.1: Source tree, Primary Git Management Software.



## Part 11.2 Github Contribution Chart





## Part 11.3 Git Log

Observe file within Documentation/SourceLogs.txt