

Patient Management for Operation Wards

P.H.R.A Darshana

2018



Patient Management for Operation Wards

**A dissertation submitted for the Degree of Master of
Information Technology**

**P.H.R.A. Darshana
University of Colombo School of Computing
2018**



Declaration

The thesis is my original work and has not been submitted previously for a degree at this or any other university/institute.

To the best of my knowledge it does not contain any material published or written by another person, except as acknowledged in the text.

Student Name: P.H.R.A Darshana

Registration Number: 2015/MIT/010

Index Number: 15550103

Signature:

Date: 02/ 07/ 2018

This is to certify that this thesis is based on the work of Mr. P.H.R.A Darshana under my supervision. The thesis has been prepared according to the format stipulated and is of acceptable standard.

Certified by:

Supervisor Name: Mr. W.V Welgama

Signature:

Date: 02/ 07/ 2018

Abstract

Most of the patients blaming their surgery become delay and they waste their time at the hospital. When discussed with the doctors, they also identified some areas that need improvement. From their feedback, we analyzed some areas that can be improved with software solution. To a doctor, accessing and reviewing the patient's records before the surgery will reduce the time delay for surgery, surgical errors and improve patient safety.

According to current process, doctors have to go through every patient's records manually several times, to gather all necessary information. This is time consuming procedure. Patient surgery registration details also are recorded manually in a book. It is very difficult to manage surgery list efficiently. Due to the shortage of the clerical staff, even emergency surgeries could get delayed too. To increase the efficiency and the productivity of the overall process, a computerized solution is introduced. With this solution, all the patients' records, availability of theaters, coworkers who's going to be there in the theater are single click away from the doctor. Beginning from the registration of a patient at the clinic to the surgery, all the intermediate processes like patient admission, examination, prescribing medicines and theater allocations are accessible to doctors through the system. Post-surgical outcome also could monitor using this system. These information could easily use for future research purposes. Considering all the facilities described above, Patient Management for Operation Wards web application is fully utilized and efficient software for theaters in hospitals to increase the quality of the service provides by them.

PMOW is developed by applying object oriented concepts, C# language using Microsoft Visual Studio Integrated Development Environment (IDE). MVP architecture is used to increase the maintainability and the extendibility of the system. Several testing has been performed to evaluate the accuracy, reliability and performance of the PMOW application.

Medical staff can access surgery related patient information through PMOW web application. This system will help to avoid unnecessary delays in the surgery and provide efficient services to the patients. Since patient information can be accessed easily in orderly manner, health staff can carry out their clinical audits and research activities comprehensively.

Acknowledgement

I express my gratitude to my project supervisor Mr. W.V Welgama, Senior Lecturer, University of Colombo School of Computing for her guidance, support and continuous encouragement throughout the project.

I also grateful to the project coordinator, Dr. F H A M Premachandra and other academic staff members of University of Colombo School of Computing for giving me the opportunity to develop the system making it the most important experience of the MIT degree program.

Finally I thank to my colleagues and friends for their valuable suggestions and comments.

Table of Contents

Declaration.....	i
Abstract.....	ii
Acknowledgement.....	iii
Table of Contents.....	iv
List of Figures.....	vi
List of Abbreviations.....	vii
Chapter 1 Introduction.....	1
1.1 Overview.....	1
1.2 Motivation.....	2
1.3 Aim and Objectives.....	2
1.4 Scope of the Project.....	3
1.5 Summary of Chapters.....	4
Chapter 2 Background.....	5
2.1 Literature Review.....	5
2.1.2. Nine Modules of Hospital Management System.....	6
2.1.3 eHospital Information Systems.....	6
2.1.4 Bayanno Hospital Management System.....	6
2.2 Analysis.....	7
2.2.1 Registrations Use Case.....	7
2.2.2 Patient Surgery Use Case.....	8
2.2.3 Generate Report Use Case.....	9
2.3 Functional Requirement.....	10
2.4 Non Functional Requirements.....	11
Chapter 3 Design.....	13
3.1 Architectural Design.....	13
3.1.1 SharePoint.....	13
3.1.2 ASP.NET MVP.....	13

3.2 ER Diagram	15
3.3 Life Cycle Model	15
Chapter 4 Implementation	18
4.1 Selection of Technology	18
4.2 Resource Requirement	18
4.3 Detailed Implementation.....	19
4.3.1 Solution Structure.....	19
4.3.2 View Implementation.....	20
4.3.3 View-Presenter Implementation.....	21
4.3.4 Model-Database Implementation	22
4.4 User Interfaces Design	23
4.4.1 PMOW – Home Page Design	24
4.4.2 Data Entry Form.....	25
4.4.3 Inquiry Pages.....	26
Chapter 5 Testing and Evaluation	27
5.1 Testing.....	27
5.1.1 Test approaches	27
5.2 Evaluation	29
5.2.1 Questions Design	29
5.2.2 Result analysis.....	30
Chapter 6 Conclusion and Future Work.....	36
6.1 Conclusion	36
6.2 Future Work.....	37
Reference	38
Appendix A - Process Flow Diagram.....	40
Appendix B - Password Encryption Logic	41
Appendix C – Test Cases.....	42

List of Figures

Figure 2.1: Registration Use Case	7
Figure 2.2: Patient Surgery Use Case	8
Figure 2.3: Generate Report Use Case	9
Figure 3.1: Architectural design	14
Figure 3.2: ER Diagram.....	16
Figure 3.3: Waterfall model.....	17
Figure 4.1: Solution Structure	19
Figure 4.2: View Implementation.....	20
Figure 4.3: Interface of the Employee Inquiry	20
Figure 4.4: Interface Implementation of the Employee Inquiry	21
Figure 4.5: View-Presenter Communication	21
Figure 4.6: View-Presenter-Model Implementation.....	22
Figure 4.7: Model-Database Communication	22
Figure 4.8: Model-Database Implementation	23
Figure 4.9: Database Logic with SP	23
Figure 4.10: Home Page	24
Figure 4.11: Patient Registration	25
Figure 4.12: Patient Inquiry	26
Figure 5.1: User Feedback Form	30
Figure 5.2: Easy to Use.....	31
Figure 5.3: Fewest Steps to Do.....	31
Figure 5.4: Errors Clearly Described.....	32
Figure 5.5: Easy to Navigate	32
Figure 5.6: Quick Response.....	33
Figure 5.7: Learn it Quickly	33
Figure 5.8: Easy to Remember	34
Figure 5.9: User satisfaction.....	34
Figure 5.10: Summary of User Feedback.....	35

List of Abbreviations

CSS	Cascading Style Sheet
ERD	Entity Relationship Diagram
ETU	Emergency Treatment Unit
ICU	Intensive Care Unit
JS	Java Script
LAN	Local Area Network
MLT	Medical Laboratory Technician
MVP	Model View Presenter
OOTB	Out Of the Box
OPD	Out Patient Department
POM	Post Operation Management
PMOW	Patient Management for Operation Wards
SP	Stored Procedure
TDD	Test Driven Development
UI	User Interface

Chapter 1 Introduction

1.1 Overview

Every day a large number of patients come to the OPD clinic of the government Hospitals. Various healthcare facilities provided by the hospitals are consultant by doctors, diagnosis and medication, surgeries, medical reports, immunization for children, etc.

Most of the persons have done at least one surgery in their life. If person needs an operation, it has to follow complex process and many of sections have to collaborate with each other. At first, a patient comes to the OPD and then proceeds to the clinic, wards, operation theatres, testing labs, ICU and anesthetists. So a patient has to undergo many processes. Each section do separate job and all government hospitals do this process manually. When one section is going to collaborate with other sections, process becomes more complex. Before surgeon going the do the surgery surgeons have to know about patient history details, medical report details, anesthetists report and surgery theater availability. They have to communicate with other sections of the hospital and it takes time. Therefore, Patient surgeries become delay unnecessary.

Hospital staffs maintain lot of written log books to keep records of the patients' information. For a day, there are many patients come to the hospital thus a lot of information are recorded in the books. Further, log books are destroyed after a certain time period due to difficulty of maintaining inadequate space requirements. There are situations where doctors are difficulty diagnose diseases due to unavailability of the patient history of lab reports since personal record books are misplaced by patient. Sometime, patient information files also lost from hospital staff. Furthermore, when patient coming to the clinic after their surgery, they have to waited in big queue to find their clinic file. Hospital staff also facing a big issue when searching a thousand of patients clinic files.

Most of the time patients come to the clinic and doctors decide which kind of surgery they are needed and patients' details are entered to the surgery waiting list along with the admit date. It may be few days, week or months. On the date of admission, patient may forget the date. In some occasions patient couldn't be admitted on the given date due to unavoidable situations. To manage this kind of situations each day hospital staff has to go through surgery list and

need to verify who is not admitted to the hospital on the particular day. Otherwise, couldn't have time to prepare another patient for the surgery. Difficulty is depending on the surgery type, patient admission date may be changed. It may be two days or one week before the surgery. Therefore, it is difficult to identify those situations in advanced through a manual system.

1.2 Motivation

People health is most important factor of the country. Government invests more money for the healthcare. But utilization of the resources is most important to provide better service to the people. Proposed system helps to provide better service for the patients.

Amount of data is become large, difficult to manage information through a manual system. Best way is stored data in one location and access it very efficient way through a software solution. Patient Management for Operation Wards is an automated system that is used to manage patients in the operation wards. Through a system we can optimize operation theatre waiting list.

There are many reasons for automating the functionality of the patients manages in the operation wards. Main reason is to avoid unnecessary delay in the surgeries and provides quality service for the patients. Also, through an automation system, both patients and hospital staff time could be saved. Daily there are large numbers of patients coming to the OPD for their surgeries needs. They have to wait longtime for their turn to appear. There are many entities like patients, doctors, laboratories and anesthetists who are involved with patient surgery and should follow some approval process to do the surgery. Most of the time patient surgeries become delay due to a communication gap between above parties. Hospital needs a system to access patient information quickly and gets the correct decisions without looking at the so many patient record books.

1.3 Aim and Objectives

The aim of the Patient Management for Operation Wards (PMOW) project is provides efficient way to access patient information related to the surgeries and minimize the unwanted delay in the surgeries. Patient surgery process has involved with other divisions of the hospital

and has to follow approval process for the surgery. Also automate the above approval processes.

Following are the objectives of the project,

- Provide a software solution to Patient Management for Operation Wards
- Minimize the time delayed in patients' operation
- In view of increasing the efficiency of surgery approval process
- Better preparation for the surgery
- To provide patients' history details quickly
- Provide facility to serve more patients
- Less paper work

1.4 Scope of the Project

The PMOW system is focus on patients, who is coming to the surgery wards and also manage their surgery related information to minimize unwanted delays in the surgery. At the very beginning system is start capturing patient information when he coming to the clinic. Appendix A section is show flow of the system. Mainly below functionalities are included in the project.

- Logins create with roles for the user registration
System has stored basic information of users and handle create, update and deactivate System Administrator, Ward doctor, Sergeant, Anaesthetics and Hospital Director users.
- Manage patient registration information
The PMOW system only manages surgery related patients. System maintain insert, update patent information.
- Manage ward Information
Inserts, updates and deactivate about ward information.
- Manage theatre registration
Inserts, updates and deletes about theatre information.

- Manage patient admission details
Inserts, updates and deletes about patient admission details.
- Manage Patients conditions information
Inserts, updates and deletes about patient condition information.
- Manage Patient register of the surgery
Patient registration for the surgery and allocate a theatre.
- Manage Anaesthetic approval process of the surgery
- Manage Hospital directors approval process of the surgery
- Inquiry patient information
Patient information is inquiry using patient Id or NIC.

1.5 Summary of Chapters

Chapter 2 is included a detailed description of functional and non-functional requirements of the system and a literature review. Chapter 3 is described the structure of the overall system, design methodologies which were used, development tools used, use case diagrams, sequence diagrams, class diagrams, entity relationship diagrams and brief introduction of database tables.

Chapter 4 is described design methodologies, implementation environment, development tools used, user interfaces and some reusable codes. Chapter 5 is provided the assessment of effectiveness, efficiency, user friendliness of the system and to decide whether the project meets the user goals and objectives. The last chapter 6 is described the summary of achievements of the project and suggestions for future enhancements.

Chapter 2 Background

People healthcare is measurable criteria of the country's development. Government allocated more money each year to improve health of the people. Do so many researches all over the world to find some solution for the health problems. Every day technology become improves and eases of people day to day life. Information technology act main role to provide quality, efficient, accurate and reliable healthcare service to people.

Patient management for operation wards software can provide automates the process of patient surgeries. This is a web application and support for the mobile responsiveness. At first, a patient comes to the OPD and then proceeds to the clinic, wards, operation theatres, testing labs, ICU and anesthetists.

Data capturing will be started when a patient come to the operations clinic. At the very first patient examine by a doctor and record their history in the system. Medical staffs, hospital wards, theatres, operation theatre list and the all approval processers related to the surgery handle by the system. This system is a web base solution and support for the mobile responsive, since authorize users can access from anywhere with any device. Specially, surgeons need to know surgery list before coming to the theater to get ready for the surgeries. Through a system hospital can provide better service to patients and patients can do their surgery minimizing unnecessary delays.

2.1 Literature Review

There are some existing systems are available related to the Hospital Management. Most of the systems try to capture the information and automate the general hospital process like record Doctors, Patient, Drugs information and Administration, Online appointment, Invoice, Medical Service, Doctor Service and Lab Management. Not specially focuses in the manage patients surgery process in the operation wards.

2.1.1 Electra Hospital Management System

Electra Hospital Management system has been developed by ACG Infotech Limited. They have covered all part of the hospital management system including the patient manage for

operation theatres. But they don't have implemented approval process of the operations. That process is different from one country to another country [1].

2.1.2. Nine Modules of Hospital Management System

Hospital Management System developed by SolutionDots. Systems are the healthcare ERP software that allows the hospital management to upgrade their management system and improve their services. There are 9 integrated Modules of Hospital Management System.

Doctors, Patient, Drugs, Administration, Online appointment, Invoice, Medical Service, Doctor Service and Lab Management systems are included. But they do not have included the patient management for the operation theatres [2].

2.1.3 eHospital Information Systems

This system has been developed by Infosystems private limited. They have managed all aspect of the hospital operations. Their system has included OPD and IPD Management, Pharmacy, Laboratory, Radiology, Ward Management, Mobile Application, Online Appointments Scheduling, Secured Messaging, Doctor Portal, Patient and Family Portals, Medical Electronic Billing, Accounting, HR/Payroll. But eHospital system hasn't a way to manage operation theatres with the allocated patients [3].

2.1.4 Bayanno Hospital Management System

Bayanno has provided the basic features of the hospital management system like management of the Patients, Admin, Doctors, Accountants, Pharmacists and Nurse.

But they don't have targeted to manage patients information with the allocated patient for the operations [4].

2.2 Analysis

This section describes business process of the Patient Management for Operation wards system and identified functional and non-functional requirements. Flow charts, module descriptions and use case diagrams can be generated for analyze the requirements. Among these options the use case diagram captures or verifies the functional aspect and very importantly the business process carried out in the system in a very lucid manner.

More details are provided in the flow chart of the section Appendix A. Following depicts a few module of the system using the use case diagrams and use case descriptions.

2.2.1 Registrations Use Case

Administrator is a top level privilege user. He can register every component of the system. Wards and theatres also have registration facility. Those are very rarely adding to the system. Both administrator and doctors can register the patients. But this registration part normally has done by the doctors. Figure 2.1 shows the use case of the user registration module.

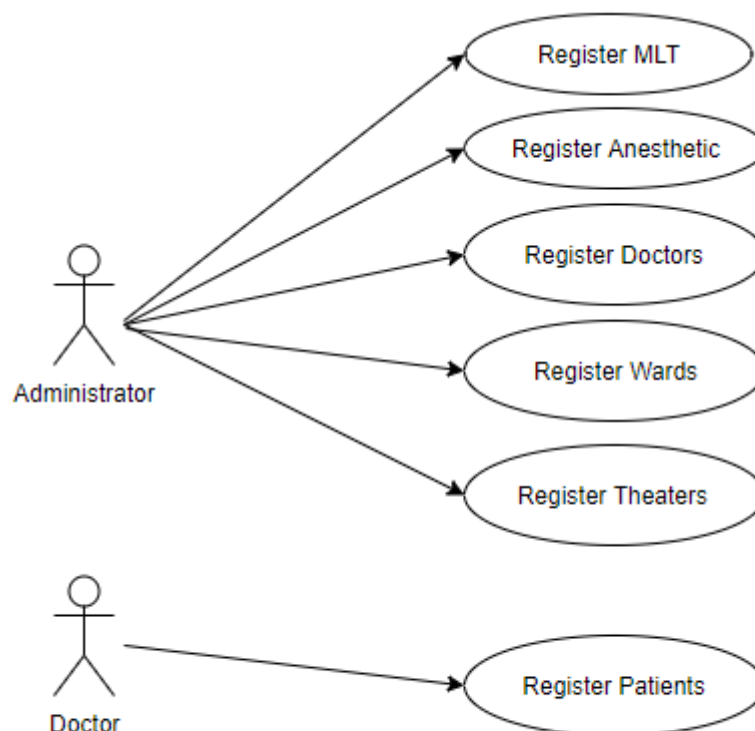


Figure 2.1: Registration Use Case

2.2.2 Patient Surgery Use Case

At the very first doctor diagnosis the patient and identify the illness at the OPD. If patient needs further treatment, admitted to hospital or advice attend to the clinic. Sometimes they have to do the surgery. For that they should admitted to the hospital. All further treatment patients information capture from the system and this is the starting point of the data capturing. Figure 2.2 shows the use case diagram of the patient's surgery.

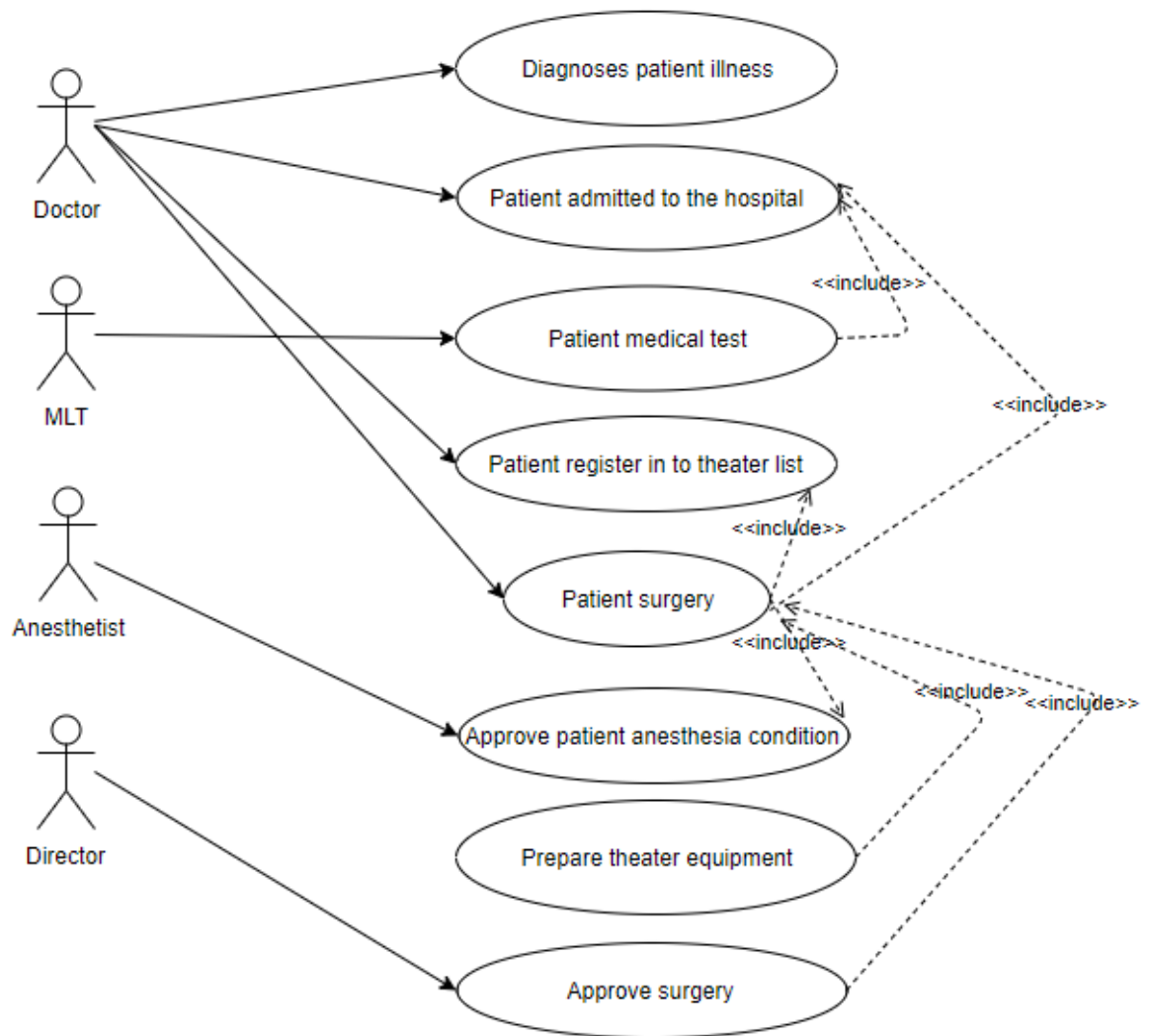


Figure 2.2: Patient Surgery Use Case

All admitted patients no need surgeries. Some people can discharge after the ward treatment. Surgery need patients wanted to register in the theater diary. Only authorized users can access

the theater diary in the system. System automatically sorted the theater list according to the surgery date of the patients.

Before the surgery, patient need to do medical test and that should be satisfied for the surgery and also need to get approval from anesthetist as well as board of directors of the hospital. All medical reports and the patient diagnostic details are stored in the system and easy to take decision about patient's surgery. All approval processes are doing through the system and efficiency of the process will be increased. Surgery doctors can look at the status of the patient through his electronic device from anywhere at any time, whether he ready or not for the surgery and surgeon can prepare for the surgery in advanced. Anesthetist also can looking at the theater list and prepares the equipment for the surgery.

2.2.3 Generate Report Use Case

System has a report generation feature. This system includes very sensitive information and therefore only doctors have privilege to generate reports. There are three kind of reports can be generate. Use case of the report generation is shown in figure 2.3.

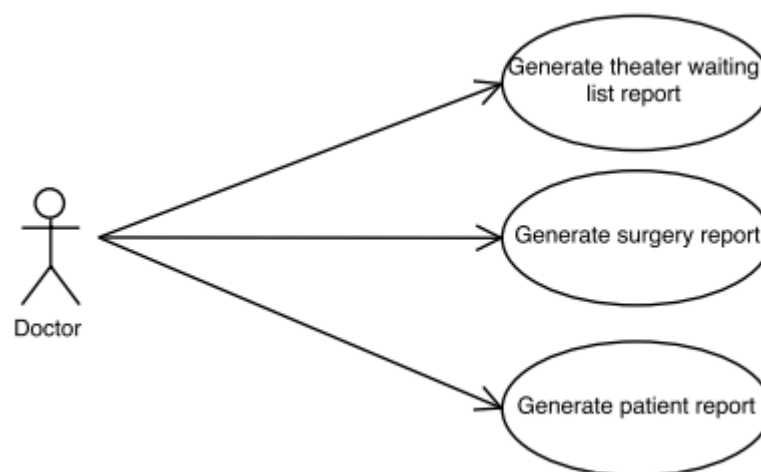


Figure 2.3: Generate Report Use Case

Theater waiting list report consists of who are registered in the waiting list and report generation page has filters to date range, pending surgery list and the completed surgery. Surgery report consist details of all patients who have done their surgery and also include date, surgeon, and status filter of the report generation.

Patient reports consist of all patients' information, which has completed and pending surgery with their treatment history details.

2.3 Functional Requirement

- Management of staff information

Doctors, MLT, Anesthetists and Directors work as staff members. Administrator should be able to insert, update and deactivate above users. Also all users should be able to search and view about staff members.

- Patient Registration

Doctors and administrators should be able to insert, update, deactivate and search patient information. Directors, MLT and anesthetists can search and view about patient information.

- Manage words, operation theaters information

Administrator should be able to insert, update, delete and query about operation wards and theaters. Other users can view and query inserted information.

- Patient health information maintains

Doctors should be able to insert, update and delete about patient diagnostics and medicines details of the patients. Also other users can search about patient health information.

- Operation theater list maintain

If patients need a surgery, they must register in the surgery list. Patient needs admitted to the hospital at correct date. Otherwise, his surgery date assign to the another patient. If patient medical test failed or anesthetics didn't approve the surgery, then another surgery date will assign to the surgery and update the surgery list. Doctors will perform the surgery according to the surgery list. Doctors should be able to patient register to the surgery and assign theater.

- Record patient medical report details

Patient medical test has been done by the MLT and added in to the system. Surgery doctor should be able to review medical report information and decide patient in good condition to the surgery.

- Anesthetist diagnose and approval to the surgery
Before to do the surgery anesthetics need to examine the patient and identify patient has good health condition to do the anesthesia and which kind of anesthesia need to the patient. Anesthetist should be able to add above diagnostic information and his approval status in to the system.
- Director approval to the surgery
Surgery final decision has been done by the hospital director. He should be able to view patient surgery information through the system and give the approval.
- Theater Reservation
There are many surgeries happen in the hospital at the day. Surgery doctors should be able to view theater availability and allocate theaters in advance.
- Post operation management
After the surgery, operation status and treatments need to mention by the surgeons. Word doctors should be able to record that information.

2.4 Non Functional Requirements

- System should be very user friendly, as the target audience may not have equivalent computer literacy.
- System should be able to provide accurate information about patients, surgeries and reports.
- Efficient retrieval of patient's medical history. The physician who consulted the patient in prior occasion in the patient's medical conditions and drugs prescribed should be readily available.
- Application should be mobile responsiveness and users should be able to access the system through their mobile devices.
- System will be contained patients highly sensitive data. System should only allow accessing authorized users. The main purpose of having different user levels is to provide more secure and flexible environment to all patients in the system.

- Current system has identified three types of reports. But at the future there are many report need to be develop. Hence, database design should capture data with appropriate relationship.

Chapter 3 Design

This section describes architectural design, design and high level component design of the Patient Management for Operation Wards system.

3.1 Architectural Design

There are many software development techniques coming to the industry every day. Software architecture design is very important part of the development. Before taking the decision, it is necessary to refer existing technology capabilities with the requirements. SharePoint and the ASP.Net MVP are popular for develop this kind of requirement. Need to analyze advantages and drawbacks to select suitable technology.

3.1.1 SharePoint

These days SharePoint is become very popular in the industry. There are two SharePoint versions available. One is SharePoint On-Premises and other one is SharePoint online.

On-Premises design organization level (LAN) service providing and need to server for hosting. Otherwise can be host in the azure cloud. SharePoint online hosted in the Microsoft servers and they maintain the infrastructure. SharePoint has list and library concept for the data storage. It has nice feature like security trimming, versioning, alert and OOTB workflows. According to the propose system, database has some level of complex queries. SharePoint list and libraries couldn't cater this requirement. System data growth happens in every day. Then list and libraries throttling issued can be occurred in the data retrieval. Considering licensing cost is high. Therefore SharePoint is not selected as suitable technology for this system [5].

3.1.2 ASP.NET MVP

A MVP design pattern is an evolved version of MVC. MVP is very popular in the software industry. It has been used Model View Presenter architecture and purpose is modularizing of the components. Also MVP design pattern will be help to [6],

- Maximize the amount of that can be tested with automaton.
- Separate business logic from user interface (UI) logic to make code easier to understand and maintain.

The MVP is Model, View, and Presenter. This pattern is how the interaction between these layers can be done. Figure 3.1 shows the structure of the system with MVP design pattern.

- View: View can be aspx page in your web applications or any user controls/Interface for the end user.
- Model: Contains all the business logic and database communication.
- Presenter: Works as the intermediate agent for Model and View. It binds the view with the model. Presenter methods don't return any values. In other way all methods in the presenter layer has void in the method signature. See the diagram below.

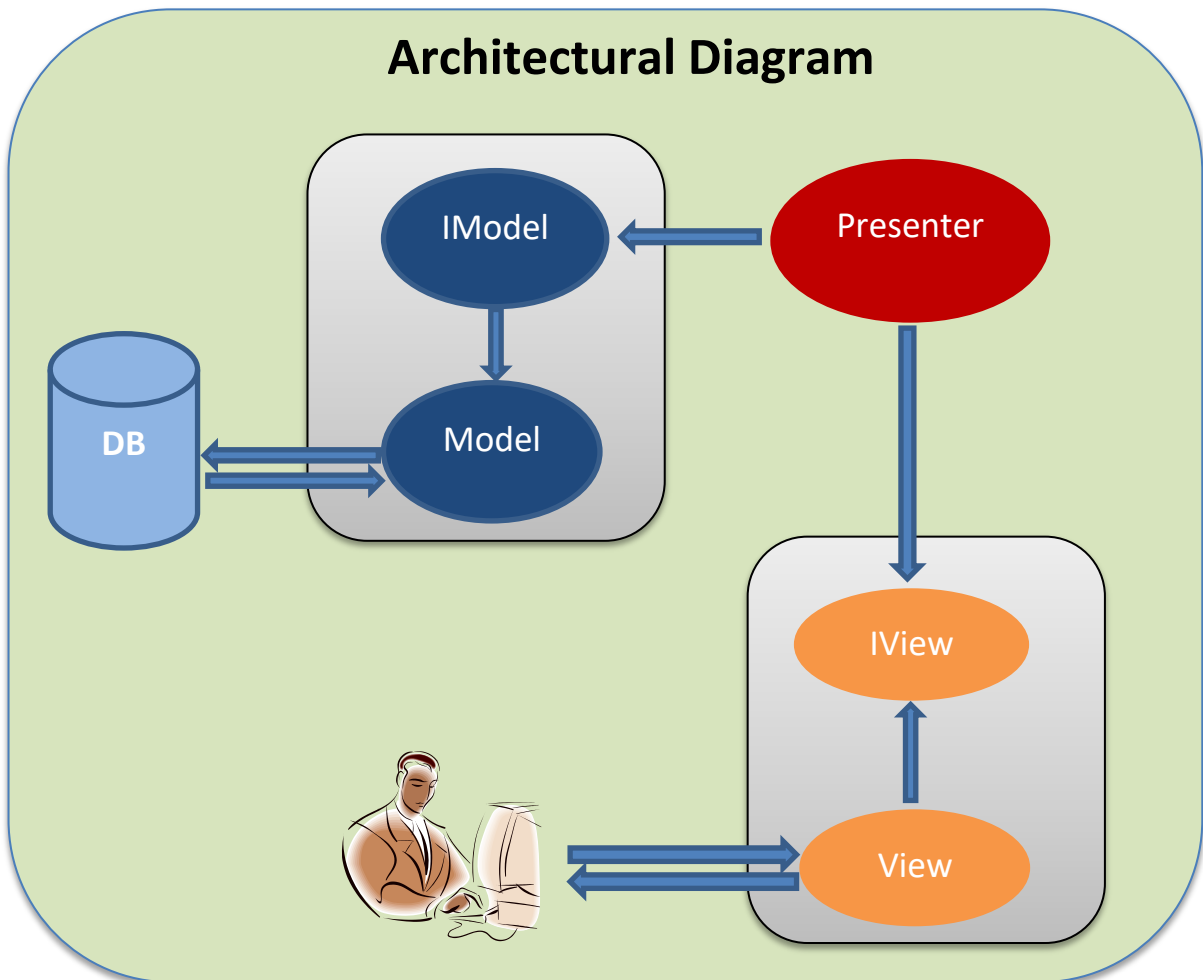


Figure 3.1: Architectural design

For the separate layer communication, it is needed to use some method to throw data. As an example, when we want to save an employee data, from view to model has to pass that information. Normal methodology is used employee type object to pass that information. But later if want to add some attributes to throw to model, have to change entity. In this project solution doesn't create entity class for the data communicate through separate layers. Instead of that Microsoft dynamic class has been used. .Net framework version 4.0 or later has been introduced dynamic object. Dynamic object behave at runtime. Using dynamic object, anytime we can create attributes and communicate throw each layers. Comparing with above factors ASP.net MVP design is better than the SharePoint solution.

3.2 ER Diagram

ERD is the only tool available for both relational data modeling and object oriented data modeling. Figure 3.2 illustrates the entity diagram with their attributes and the relationship between entities of the system. It also shows the primary keys, foreign keys and cardinality ratio of the entities.

3.3 Life Cycle Model

Proposed system requirement is very clear. Process of the patient surgery is fixed. Very less probability to requirement becomes change. Then easy to finalized requirements and design. Therefore Waterfall software development lifecycle model has been used in this system. Figure 3.3 shows the process of the waterfall model [7].

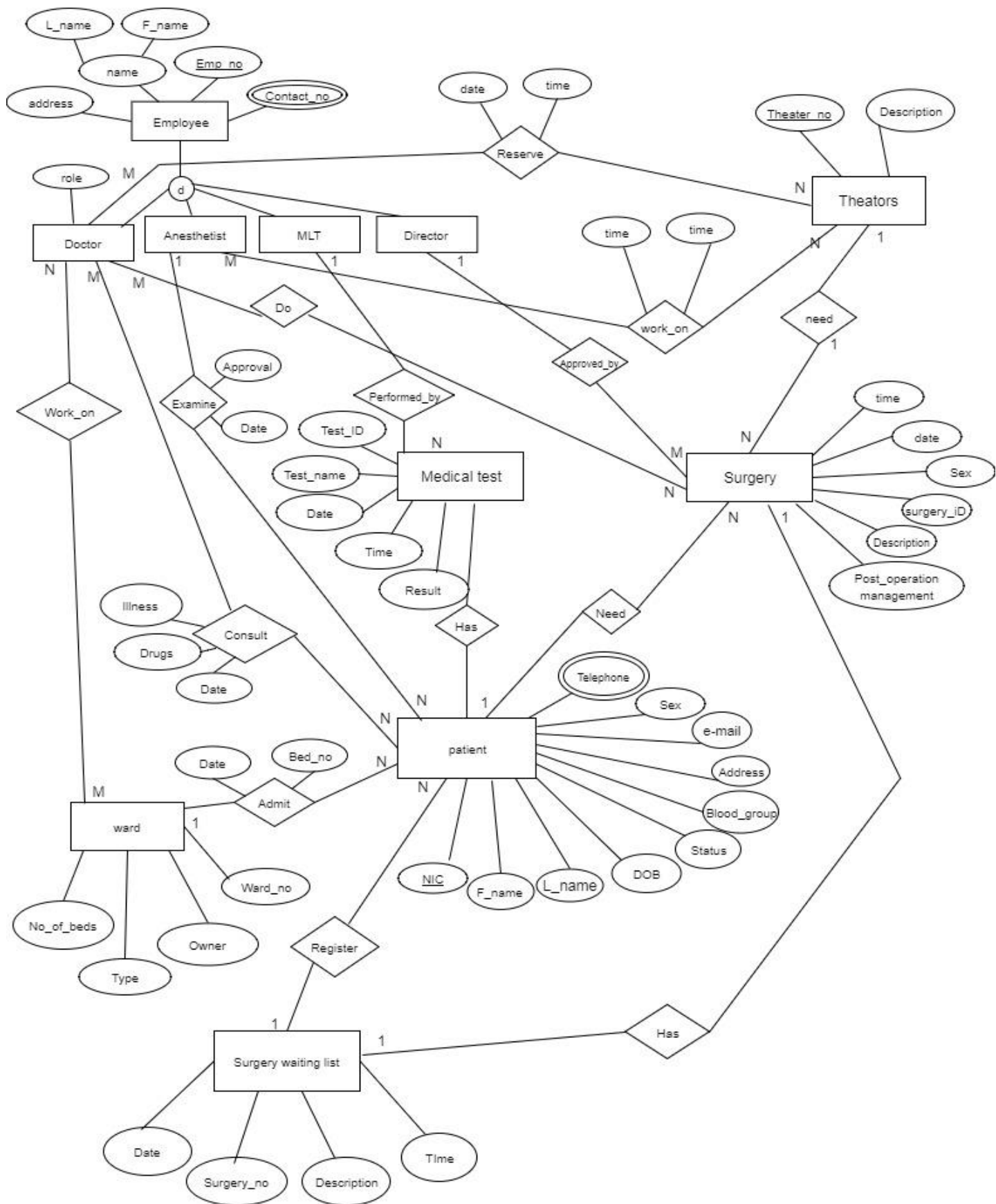


Figure 3.2: ER Diagram

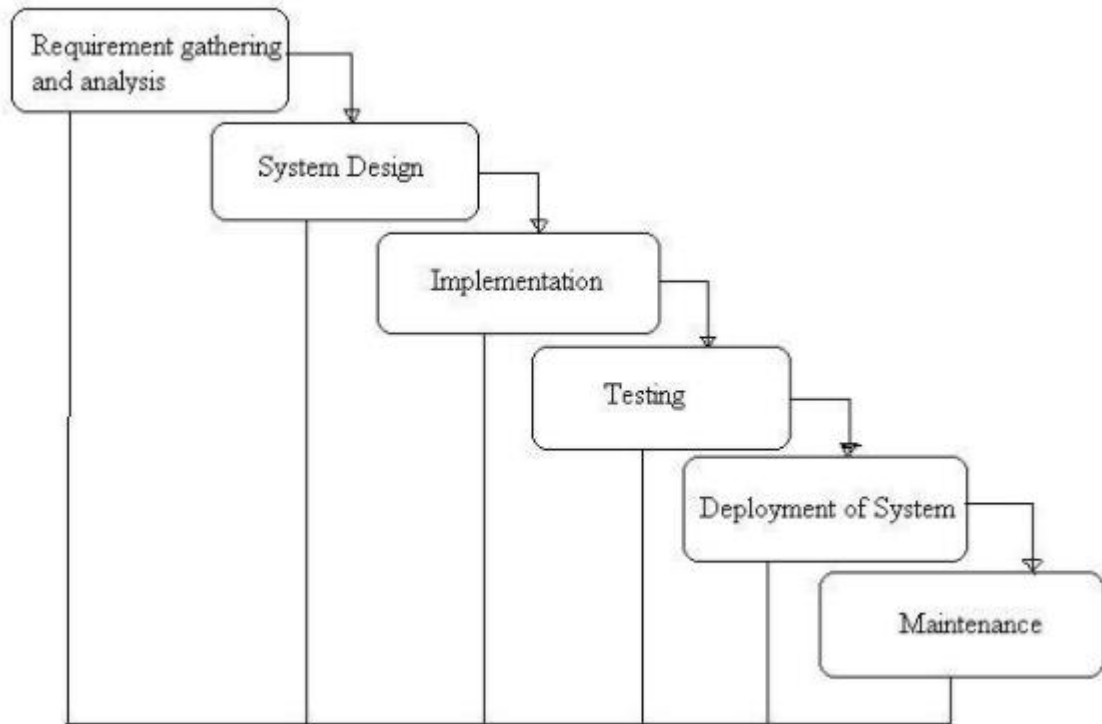


Figure 3.3: Waterfall model

Chapter 4 Implementation

Implementation chapter is focus on the implementation of the project. The implementation of the Patient Management for Operation Wards web application is carried MVP architecture where each module is implemented separately and later integrates each other.

The scope of this chapter is to discuss the how each feature was developed, Use of features, possible screen shorts, and code segment as well.

4.1 Selection of Technology

After designed entire system had to decide what kind of technology use for develop the system. Entire system based on .Net framework 4.6 and SQL Server 2012 .In here web application develop by C# .Net language with inherit ASP.Net .Visual Studio 2015 is the development tool .This PMOW will able to host in local machines by using IIS server. SQL Server 2012 is provided useful data storage facility and data manipulation methods for application development .The main point is .Net Framework providing very developer friendly environment for SQL Server 2012 because both are include same Microsoft family.

jQuery has been used to client side scripting and some user friendly validations [8]. Also used bootstrap for create attractive interfaces and mobile responsive [9].

4.2 Resource Requirement

Hardware Requirement

- Personal Computer / Laptop (with 2GHz or higher processing speed)
- 4 GB or higher RAM
- Router to connect the internet

Software Requirement

- Windows 8.1/ Windows10
- Microsoft Office for documentation purpose
- Microsoft Project
- Rational Rose for documenting the design
- Microsoft Visual Studio 2015
- SQL Server Management Studio 2012

- Photoshop
- Web browser (Google chrome, Internet explorer, Mozilla)

Frameworks

- Bootstrap
- jquery.js
- datepicker.js

4.3 Detailed Implementation

MVP design pattern is not just created object and throw data to other layer. It has standards way to communicate through layers. In this section described how to MVP design pattern apply to PMOW web application and how to communicate with each layer.

4.3.1 Solution Structure

According to the proposed MVP design pattern of PMOW solution structure has been created. Each layer separated with the folders. As an example all the views, presenters and models has separate folders and that very helpful to maintain the solution. All the CSS files are in separate folder. Common styles are write in main.css file and page specific CSS files are maintain as separate files which is named followed by the module name. Figure 4.1 shows the folder structure of the visual studio solution.

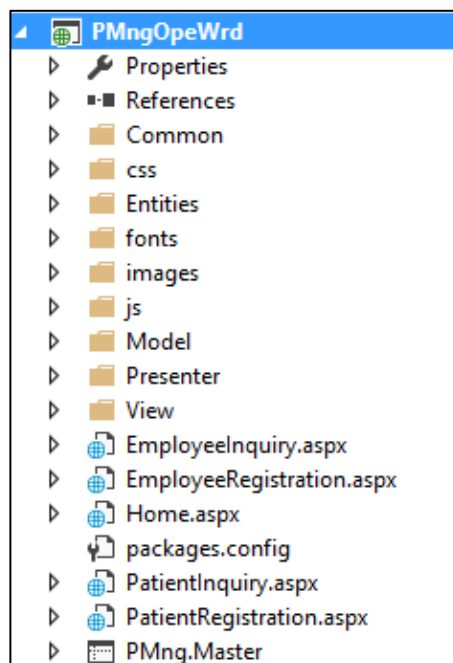


Figure 4.1: Solution Structure

4.3.2 View Implementation

View responsibility is to handle all UI logics. Figure 4.2 shows how view implements with the user control and get data from the user inputs.

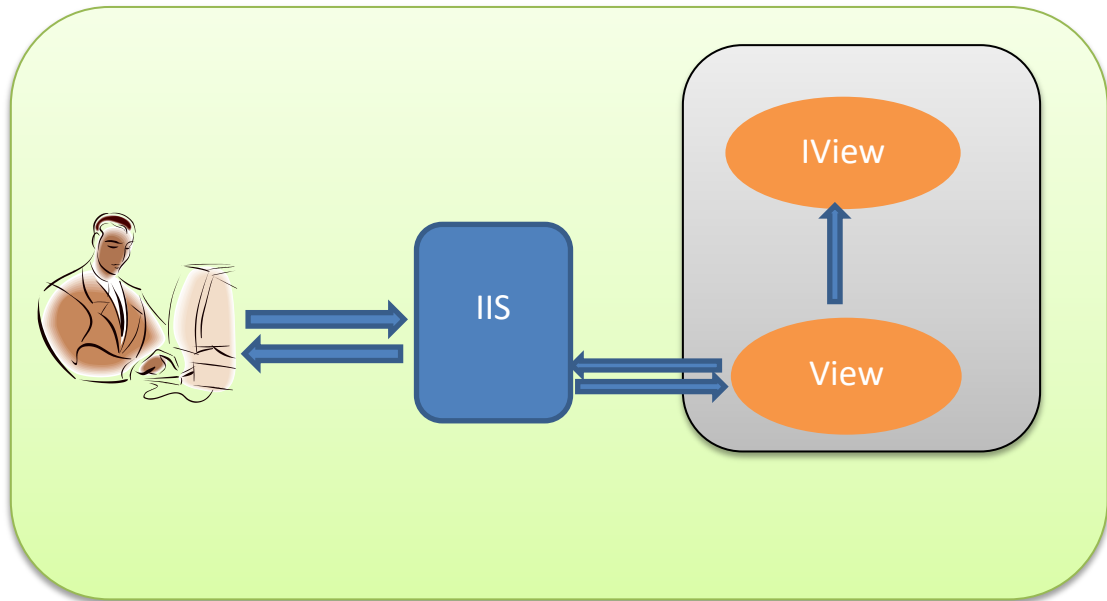


Figure 4.2: View Implementation

All UI controls are implemented in the aspx pages. Code behind file consider as a view. Each input form control which is need to communicate with the Presenter, need an interface. Interface defined in the IView interface with the getters and setters. Interface inherited to the view and interface implementation need to do in the view. Getters and setters are used to value set and get from the form controls. Best practice is developed only getters and setters in the view.

Figure 4.3 and 4.4 show Employee Inquiry interface and the implementation.

```
3 references
public interface IEmployeeInquiryView
{
    1 reference
    string employeeId { get; set; }
    3 references
    DataTable employeeData { set; }
    2 references
    string searchColumn { get; }
    3 references
    string searchValue { get; set; }
}
```

Figure 4.3: Interface of the Employee Inquiry

```

1 reference
public string employeeId
{
    get
    {
        return hdnEmployeeId.Value == "" ? string.Empty : hdnEmployeeId.Value;
    }
    set
    {
        hdnEmployeeId.Value = value;
    }
}

3 references
public string searchValue
{
    get
    {
        return txtSearchId.Text;
    }
    set
    {
        txtSearchId.Text = value;
    }
}
}

```

Figure 4.4: Interface Implementation of the Employee Inquiry

4.3.3 View-Presenter Implementation

There are no return types in presentation layer methods. View and Presenter communicate through the interface properties. When initialize the constructor of the presenter, create an object of the view interface and model. Through that object, presenter can get and set the values to the view. Also presenter can access the method of the model. In MVP design, presenter works as a bridge between view and the model. It does not handle any business logic. Figure 4.5 shows the how view-presenter communicates with each other.

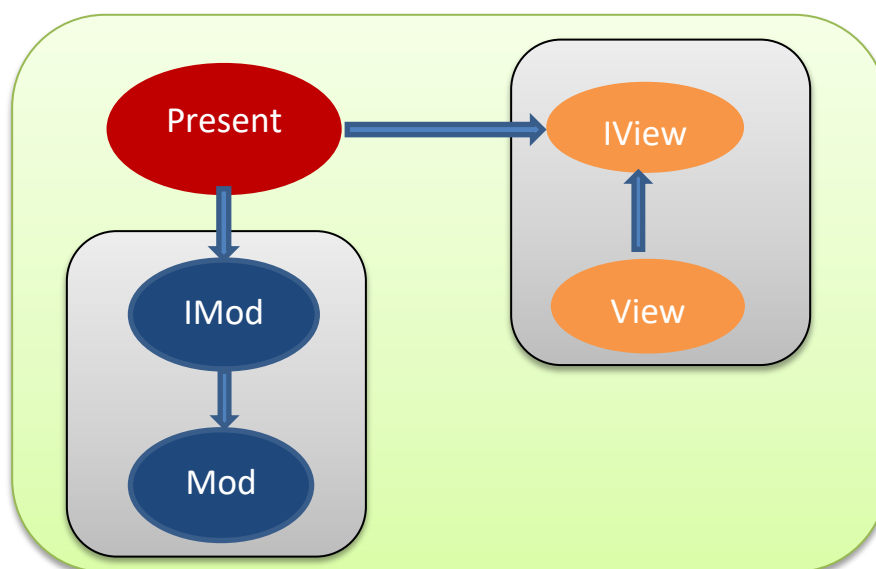


Figure 4.5: View-Presenter Communication

Figure 4.6 shows the code implementation of the View-Presenter-Model.

```
3 references
public class PatientRegistrationPresenter
{
    IPatientRegistrationView patientView;
    PatientRegistrationModel patientRegistration;

    /// <summary> constructor of the patient presenter
    1 reference
    public PatientRegistrationPresenter(IPatientRegistrationView view)
    {
        patientView = view;
        patientRegistration = new PatientRegistrationModel();
    }

    /// <summary> get patient information by given id
    1 reference
    public void GetPatientById()
    {
        DataTable patientData = patientRegistration.GetPatientById(patientView.patientId);
        patientView.firstName = patientData.Rows[0]["FirstName"].ToString();
        patientView.lastName = patientData.Rows[0]["LastName"].ToString();
        patientView.NIC = patientData.Rows[0]["NIC"].ToString();
        patientView.address = patientData.Rows[0]["Address"].ToString();
        patientView.mobilePhone = patientData.Rows[0]["MobilePhone"].ToString();
        patientView.landPhone = patientData.Rows[0]["LandPhone"].ToString();
        patientView.email = patientData.Rows[0]["Email"].ToString();
        patientView.bloodGroup = patientData.Rows[0]["BloodGroup"].ToString();
        patientView.gender = patientData.Rows[0]["Gender"].ToString();
        patientView.maritalStatus = patientData.Rows[0]["MaritalStatus"].ToString();
        patientView.emergencyContact = patientData.Rows[0]["EmergencyContact"].ToString();
        patientView.dateOfBirth = patientData.Rows[0]["DateOfBirth"].ToString();
        patientView.gardianName = patientData.Rows[0]["GardianName"].ToString();
        patientView.gardianAddress = patientData.Rows[0]["GardianAddress"].ToString();
        patientView.isNewPatient = "false";
    }
}
```

Figure 4.6: View-Presenter-Model Implementation

4.3.4 Model-Database Implementation

PMOW haven't much complex business logic. Therefore, rather than using separate classes to handle business logic, used model class for that purpose. Then model has responsibility to handle all business logic and communicate with the database layer. Figure 4.7 shows how the model and the database are structured in the solution.

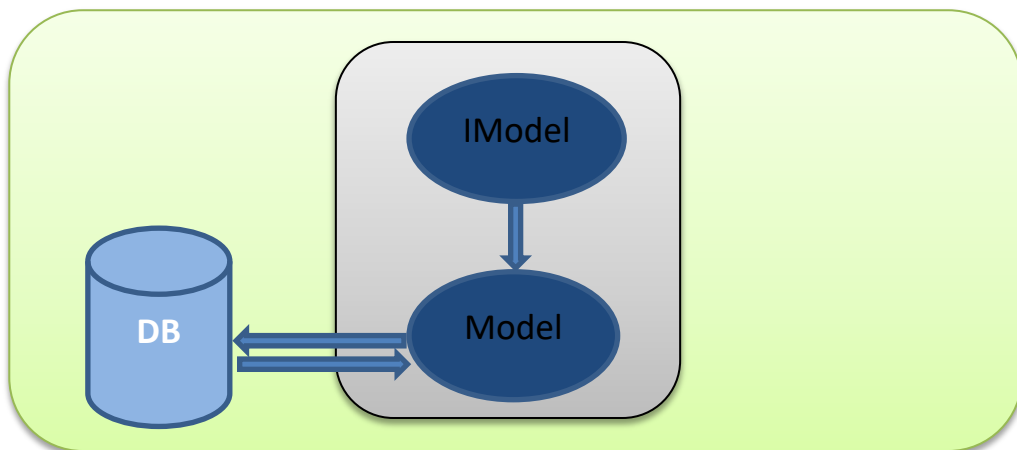


Figure 4.7: Model-Database Communication

Model interface has all the method definitions of the model. Model has been inherited that interface class and do the implementation of the model. At the beginning of the class needs to need to make a database connection. When we implementing the method have to make sure if database connection is closed, then open the connection. Now model can communicated with the database. Figure 4.8 shows implementation of the model-database layers.

```

//database connection string
SqlConnection sqlCon = new SqlConnection(@"Data Source=RDARSHANA;Initial Catalog=PntMngOpeWrd;
MultipleActiveResultSets=true;Integrated Security=true;");

2 references
public DataTable GetAllEmployeeData()
{
    if (sqlCon.State == ConnectionState.Closed)
    {
        sqlCon.Open();
    }

    SqlDataAdapter sqlDa = new SqlDataAdapter("GetAllEmployeeData", sqlCon);
    sqlDa.SelectCommand.CommandType = CommandType.StoredProcedure;
    DataTable dataTable = new DataTable();
    sqlDa.Fill(dataTable);
    sqlCon.Close();
    return dataTable;
}

```

Figure 4.8: Model-Database Implementation

Model and database communicate using stored procedures. That is very helpful to maintain our code. Also stored procedures have been used to handle some level of business logic. Because of that if need change the business logic of the application, no need to do the solution code change. Only have to do the stored procedure change. Figure 4.9 shows the sample SP.

```

CREATE PROCEDURE [dbo].[GetNextPatientId]
AS
BEGIN
    DECLARE @RowCount INTEGER;

    SET @RowCount = (SELECT COUNT(*) FROM dbo.Patient WHERE YEAR(CreationDate) = YEAR( getdate()));
    SELECT CONCAT('PAT', @RowCount +1 , '-', YEAR(getdate()));
END

```

Figure 4.9: Database Logic with SP

4.4 User Interfaces Design

The user interface for the software shall be compatible to any browser such as Internet Explorer, Mozilla, Safari and Chrome access to the system. In this section describe some of

main user interfaces. PMOW is used different age group users and some of them haven't good computer skills. Since develop the interface, mainly focusing on simplicity of the system.

4.4.1 PMOW – Home Page Design

Homepage content includes the description of the hospital like their services. After the user logged in, he will be redirect to the home page as startup page. Figure 4.10 shows the home page. Through a main navigation user can navigate to the pages [10]. Also bottom of the page has included footer of the page. It has included “Contact us” and the “About us” details. Both main navigation and footer come from master page and these are common for all pages. Main navigation has security trimming. Authorized users can only navigated to the particular page.

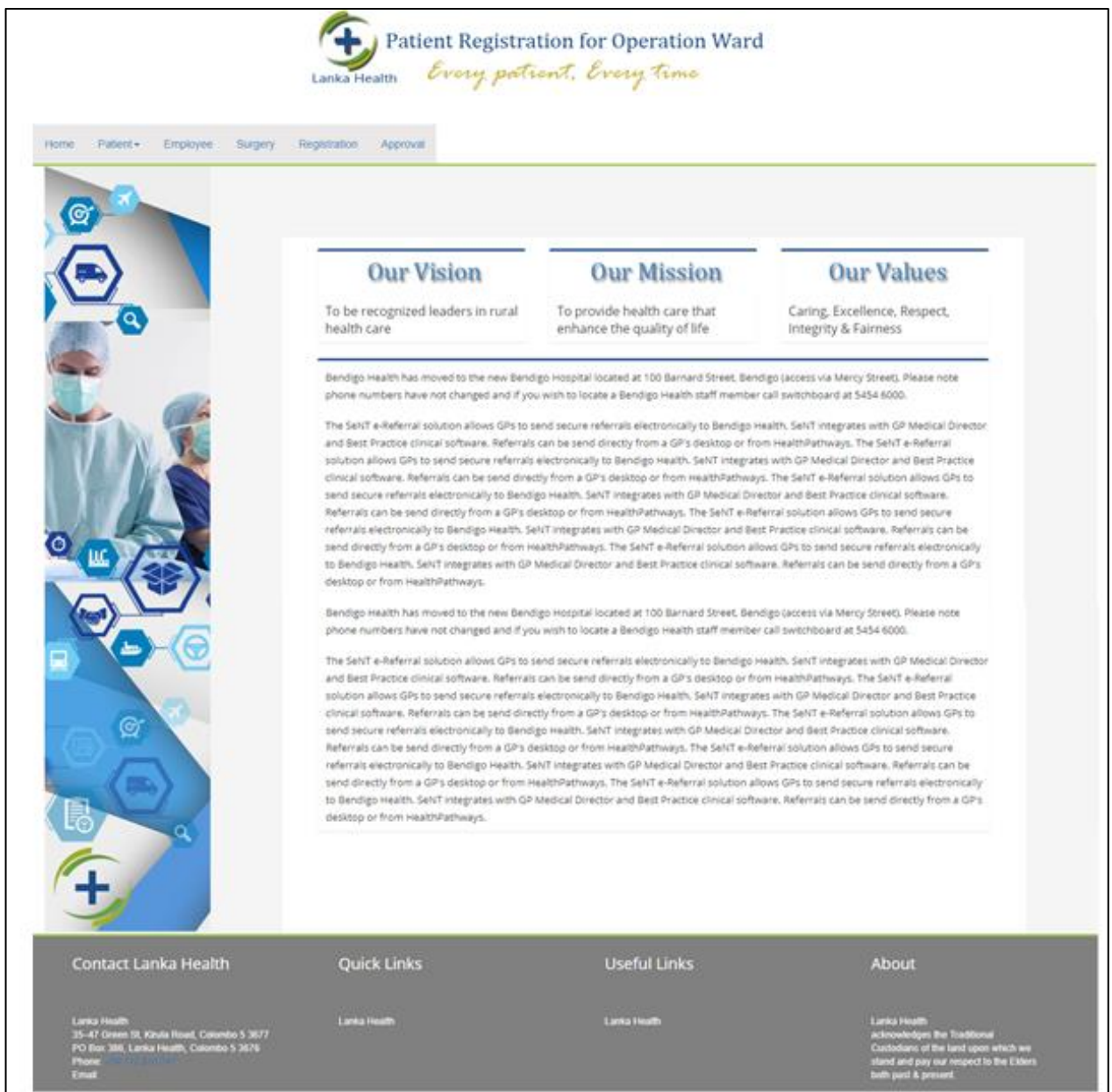
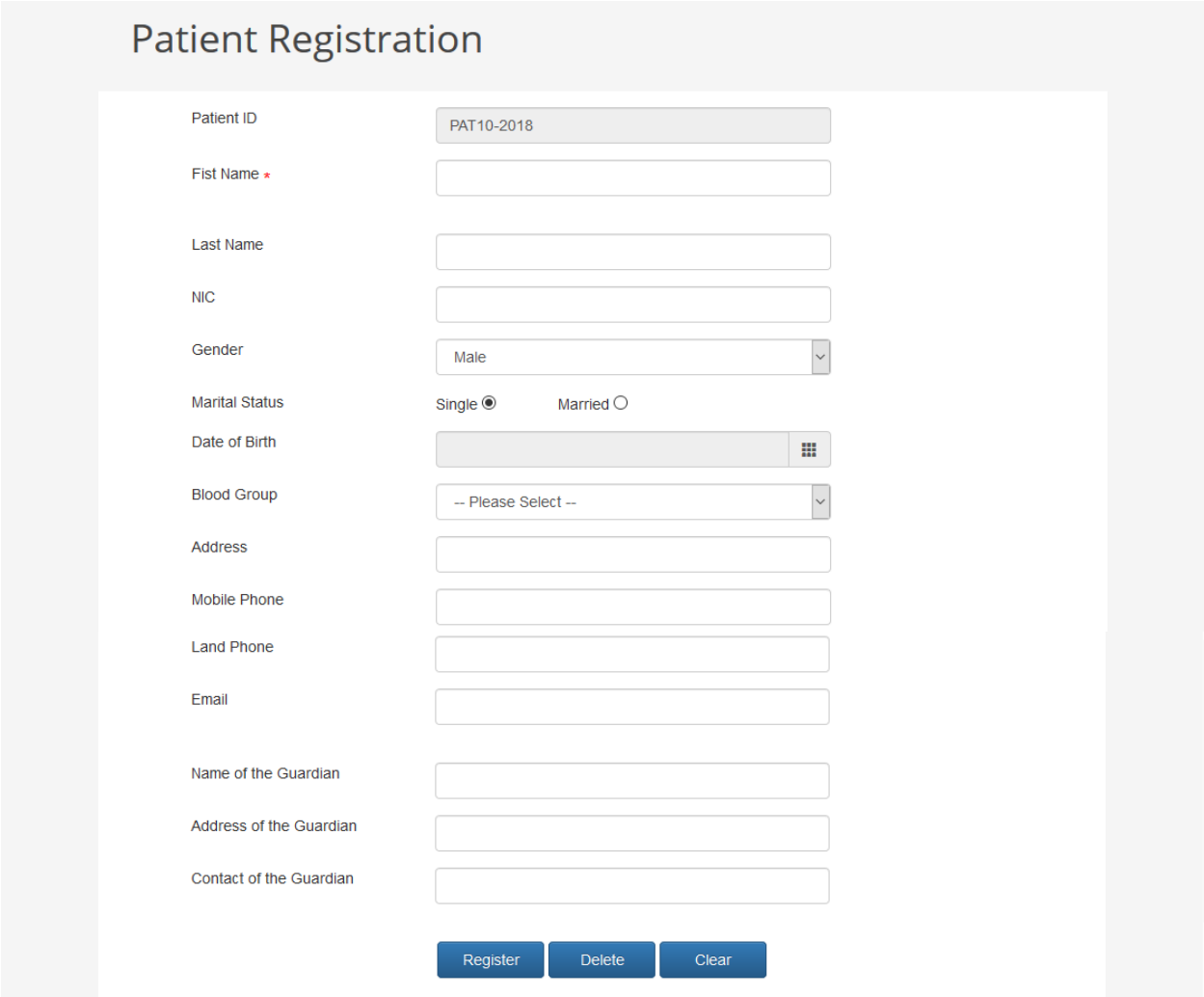


Figure 4.10: Home Page

4.4.2 Data Entry Form

Figure 4.11 shows the how data entry forms basically structured. This interface related to the Patient Registration it used to register patient at the first time coming to the clinic. Bootstrap classes have been used to structure the page. ASP controls are used as a form controls and bootstrap control classes used to style controls. ASP validation controls are used to form validation. Button text, notification functionality and field visibility handles by the server side code.



The screenshot displays a web form titled "Patient Registration". The form contains the following fields and controls:

- Patient ID:** A text input field containing "PAT10-2018".
- Fist Name *:** A text input field.
- Last Name:** A text input field.
- NIC:** A text input field.
- Gender:** A dropdown menu with "Male" selected.
- Marital Status:** Radio buttons for "Single" (selected) and "Married".
- Date of Birth:** A date picker control.
- Blood Group:** A dropdown menu with "-- Please Select --" selected.
- Address:** A text input field.
- Mobile Phone:** A text input field.
- Land Phone:** A text input field.
- Email:** A text input field.
- Name of the Guardian:** A text input field.
- Address of the Guardian:** A text input field.
- Contact of the Guardian:** A text input field.

At the bottom of the form, there are three buttons: "Register", "Delete", and "Clear".

Figure 4.11: Patient Registration

4.4.3 Inquiry Pages

Figure 4.12 shows the how data entry page basically structured. This interface related to the Patient inquiry and used to view and search patient information and only authorized users can navigate to this inquiry page. ASP.Net grid view is used to display search results and bootstrap classes is used to style grid view. Default page is appeared with the all patient records. When user clicks on the search button, then identify column name and field value from the interface and passed to the database layer. Dynamic SQL stored procedure is used to execute and return matching value from the database. Pagination is available in the grid view and it's appeared when number of records exceeds the row limit [11].

Patient Id	First Name	Last Name	NIC	Mobile Phone	Email	Gender	Marital Status	Emergency Contact	Date Of Birth	
PAT1-2017	Rukshan	Darshana	873383070V			male	Single			View
PAT1-2018	Nilanka	Harshana	865784251V			male	Single			View
PAT2-2017	Lakshika	Prasadini	885793070V			female	Married			View
PAT2-2018	Viraj	Nimarshana				male	Single			View
PAT3-2018	Rajith	Disanayake	835578465V			male	Single			View
PAT4-2018	Yasitha	Nawagamuwa	892547242V			male	Single			View
PAT5-2018	Laksitha	Okadapola				male	Married			View
PAT6-2018	Rukshan	acns				male	Single			View
PAT7-2018	Kalana	Pathishantha	875894944V			male	Single		14/11/2017	View
PAT8-2018	Damitha	Shamantha				male	Single			View

Figure 4.12: Patient Inquiry

Chapter 5 Testing and Evaluation

5.1 Testing

In general, Testing is a process carried out by individuals or groups for end to end domains where requirements exist. It provides an objective, independent analysis of the software to allow understanding the risk of software implementation as well.

In the Patient Management for Operation Ward web application has been used manual testing to verify the functional accuracy. Below table have been included functional test cases.

5.1.1 Test approaches

This section describes a test approach that was planning to use Patient Management for Operation Wards web application. Blow testing mythologies has been used to test the web application.

- Unit Testing
- Integration Testing
- Smoke Testing
- Regression Testing
- User Acceptance Testing

5.1.1.1 Unit Testing

Unit testing was done at the lowest level. It tests the basic units of software, which is the smallest testable piece of software, and it is often called “unit “, module or “component” interchangeably. Unit testing refers to tests that verify the functionality of a specific section of code, at the beginning of function levels [12].

5.1.1.2 Integration Testing

Integration testing is the phase in software testing in which individual software modules are combined and tested as a group. It occurs after unit testing. Integration testing takes as its input modules that have been unit tested, groups them in larger aggregates, applies tests

defined in an integration test plan to those aggregates, and delivers as its output the integrated system ready for system testing [13].

5.1.1.3 Smoke Testing

Smoke Testing, also known as “Build Verification Testing”, is a type of software testing that comprises of a non-exhaustive set of tests that aim at ensuring that the most important functions work. The results of this testing is used to decide if a build is stable enough to proceed with further testing [14].

5.1.1.4 Regression Testing

Regression testing is a type of software testing that ensures that previously developed and tested software still performs the same way after it is changed or interfaced with other software. Changes may include software enhancements, patches, configuration changes, etc. During regression testing, new software bugs or regressions may be uncovered. Sometimes a software change-impact analysis is performed to determine which areas could be affected by the proposed changes [15].

5.1.1.5 User Acceptance Testing

User acceptance is a type of testing performed by the Client to certify the system with respect to the requirements that was agreed upon. This testing happens in the final phase of testing before moving the software application to Market or Production environment [16].

5.1.2 Test Cases

A test case is a set of conditions or variables under which a tester will determine whether a system under test satisfies requirements or works correctly. The process of developing test cases can also help find problems in the requirements or design of an application. The PMOW system test cases are designed based on the module functionality and test cases are covered all the functional requirement of the system. Appendix C has been describes functional testing of the each module in the Patient Management for Operation Wards web application.

5.2 Evaluation

Evaluation is an important part of any project or program, which aims to bring out the value of what has been done, and help capture the learning that has taken place during the activity. The solution, Patient Management for Operation Wards web application was evaluated in the perspective people who are involving with patient surgery process.

There are five user types are involved with the system and they have different roles. Therefore during the process of evaluations decided to select 5 users in each role. At the beginning give brief introduction of the system for all users. Then assign task to each individuals to carryout cover all process of the patient surgery.

- System Administrators
- Doctors
- Hospital Directors
- MLT
- Anesthetist

5.2.1 Questions Design

After the each user task completed, provide questionnaires to get user feedback in different areas and analyze all response. Figure 5.1 shows the all user evaluation form of the PMOW system.

Patient Management for Operation Wards Web Application

Evaluation Form

This is a Patient management for operation wards web application evaluation form. This will be help to identify user feedback of the system. Please use below rating levels to complete feedback table.

Ratings

Poor - 1, Fair - 2, Average - 3, Good - 4, Excellent - 5

Description	Ratings (Fill with the number)
It is easy to use.	
It requires the fewest steps possible to accomplish what I want to do with it.	
It is errors clearly described.	
It is easy to Navigate.	
It is response in satisfy time.	
I learned to use it quickly.	
I easily remember how to use it.	
I am satisfied with it.	

Comments:

Name (Optional): _____

Contact No (Optional): _____

Figure 5.1: User Feedback Form

5.2.2 Result analysis

Each user role represented by five uses and analyzed all responses user role wise. Finally build a pie chart summarizing all responses. It will easy to understand user feeling of the users.

- It is easy to use.

Through this measurement has been measured how easy to carry out their tasks without trouble. According to figure 5.2, PMOW system is easy to use.

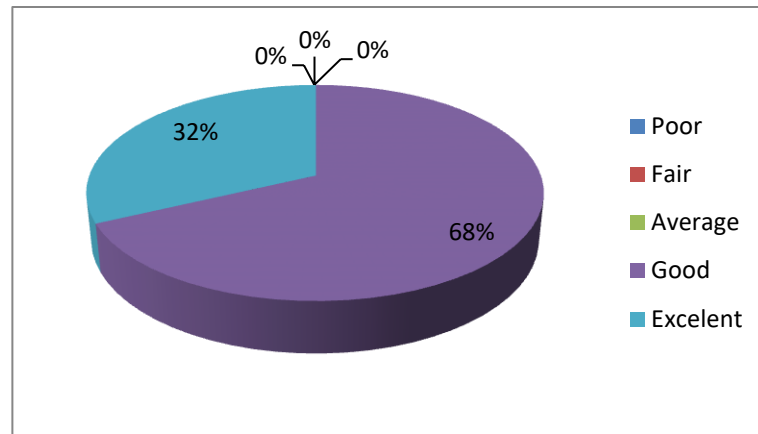


Figure 5.2: Easy to Use

- It requires the fewest steps possible to accomplish what I want to do with it.

This measures how user can finish their task within few steps. As an example how easy to refer patient history details through a system before doctor going to examine patient. Figure 5.3 show users feedback.

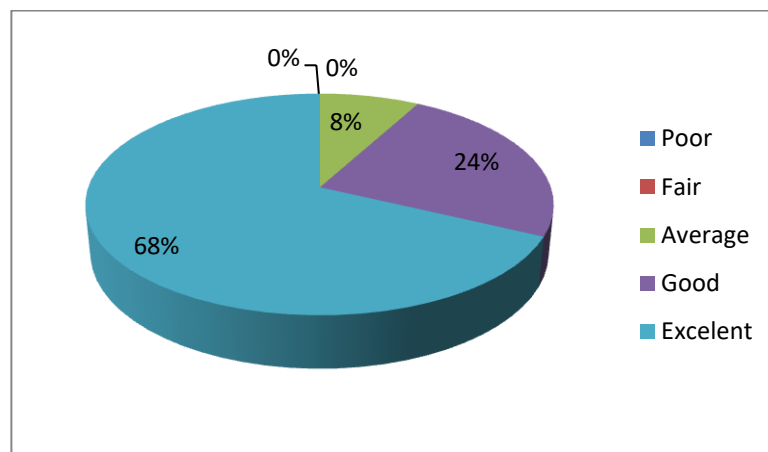


Figure 5.3: Fewest Steps to Do

- It is errors clearly described.

This measure when error occur while user carry out their task, how user can easily identify what is the error through a system message. Figure 5.4 shows user feedback.

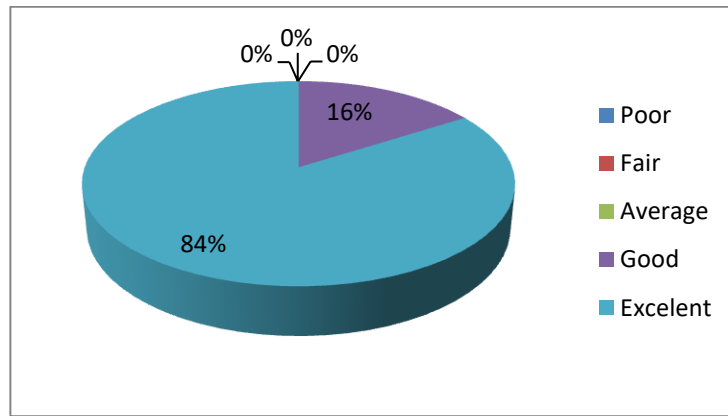


Figure 5.4: Errors Clearly Described

- It is easy to navigate

This measure how easy to user can navigate through pages to do their task. Figure 5.5 shows users gave positive feedbacks.

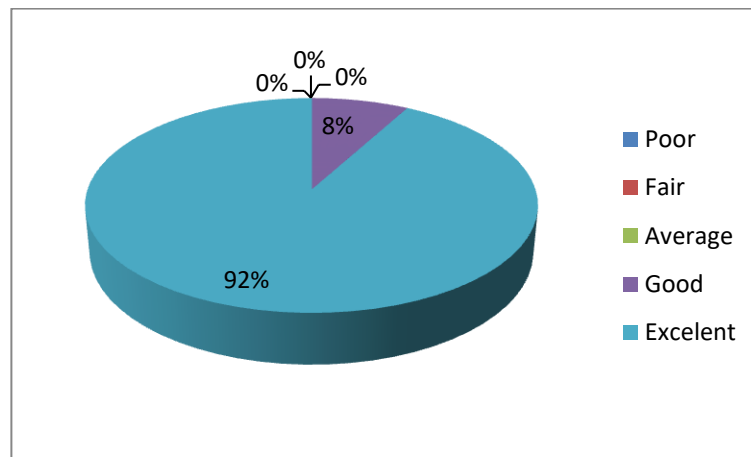


Figure 5.5: Easy to Navigate

- It is response in satisfy time.

This measure after user does something in the system how fast system reacts to the user action. Figure 5.6 shows efficiency of the system.

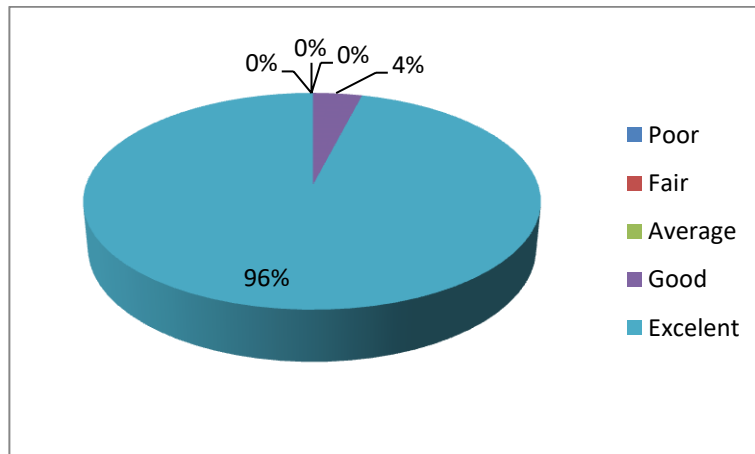


Figure 5.6: Quick Response

- I learned to use it quickly.

This is measure, assign task to user and how user can continue it without problem.

Figure 5.7 shows about user feedback.

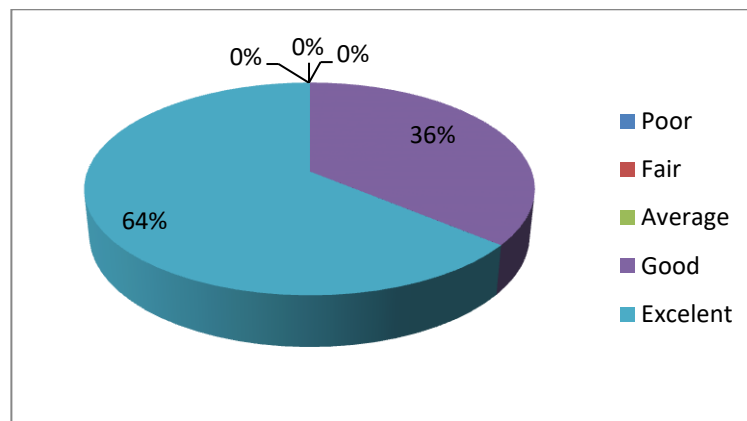


Figure 5.7: Learn it Quickly

- I easily remember how to use it.

This is measure assign task to user after gave some introduction about the system and measure how they remember steps to carry on their task. Figure 5.8 shows about user feedback.

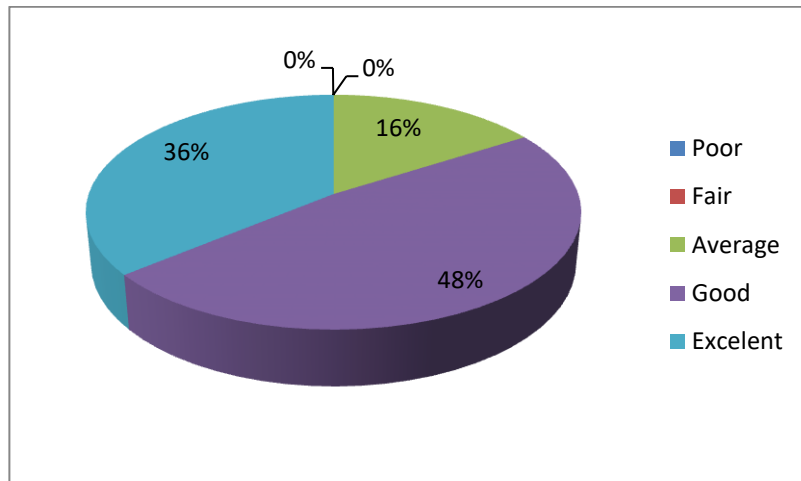


Figure 5.8: Easy to Remember

- I am satisfied with it.

This is measure overall satisfaction of the system. Also users give their suggestions to add some other features to the system. According to the figure 5.9 users have satisfy about the system.

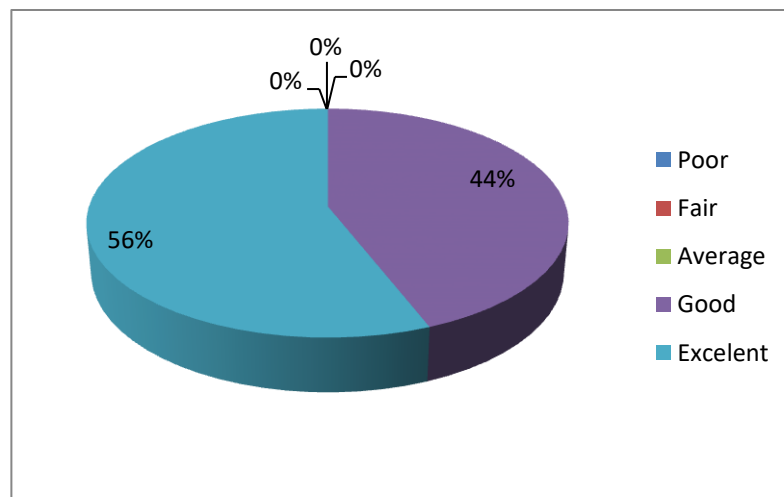


Figure 5.9: User satisfaction

After getting feedback of each user evaluation criteria, generate a summary graph of the user feedback. Figure 5.10 shows summary of the User feedback

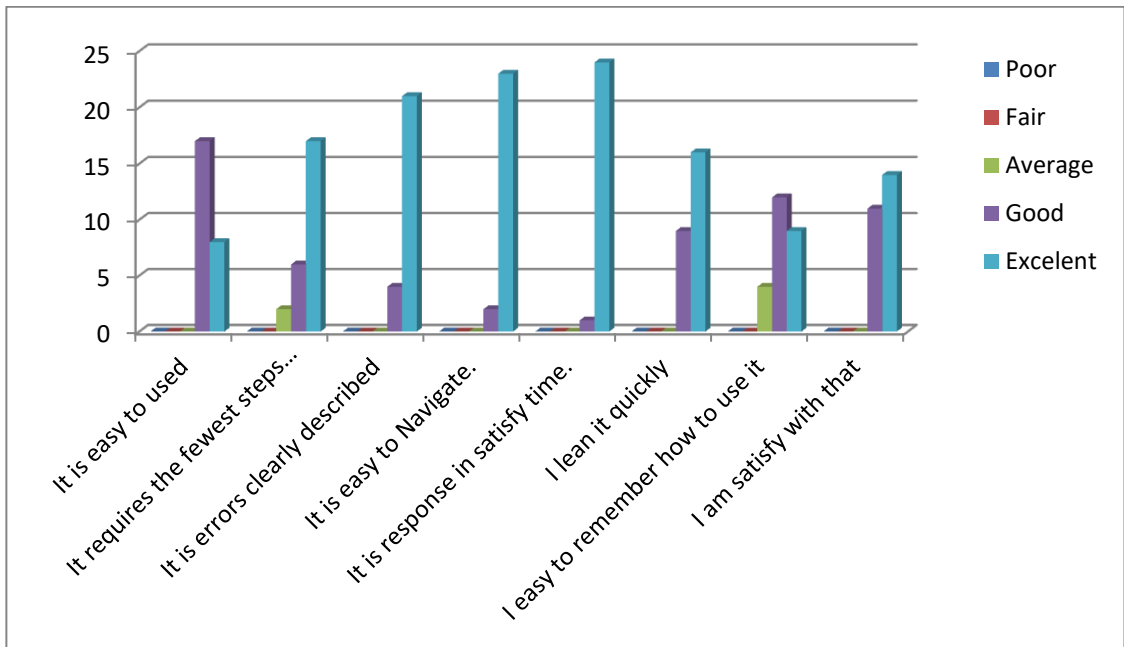


Figure 5.10: Summary of User Feedback

Chapter 6 Conclusion and Future Work

6.1 Conclusion

Goal of this project is to implement a system which will serve hospital staff to provide efficiency service to the patients in the surgery process. Through this system, hospital staff can access all the information needed for the surgery at one location. When patient need to do the surgery, he need to follow some process and many parties participate for that until end of the surgery. In PMOW system identified each user roles and provided separate login for them. According to their user role they have responsibilities and provided separate interface to efficiently provide their services. When they need to take decision, they can easily refer patient information without waiting for others.

As an example sergeant need to take decision before the surgery weather this patient has good enough health condition to the surgery. Then sergeant can directly check patient history details from the system and take necessary actions in advance. Early decision will be help to get sergeant valuable service to another patient.

Before start surgery has to follow approval process like word doctors, anesthetist and hospital directors. Through this system sort listed their pending approval and that will help to reduce unnecessary delay of the surgery. And also the PMOW is a web application and authorized users can access the system at anywhere. Since staff members no need to come to the hospital to look at their pending tasks and also sergeant can look at surgery list in advance to prepare for the surgery.

Some of users are not much familiar with computer base system with their age range and they have work more time with paper works. But this system hasn't complex logics. With some training they can adopt to the new system. Young ages easily adopt with this system.

In the requirement gathering phase has faced some difficulty to match their requirement to software solution. Few times discuss with them to clear their requirement. Sometime has to provide them to chart like process flow of the system. In the development phase time to time met them with developed system to make sure both we are in same picture. Hope this system will be helpful to provide better service to the patients.

6.2 Future Work

According to the current implementation PMOW scope is only for one hospital and targeted to reduce delay of the patient's surgery in that hospital. If this system is connected with the other hospital, it will be help to manage Hospital's resources and save patient life.

The PMOW is hosted in the cloud. But if internet connection is broken, couldn't find the patient information to continue surgery process. For avoid that kind of situation, through a system need to provide offline database for hospital network. Also this information is very critical. If virus has been attack, system becomes unavailable. Due to avoid such a situation system should provide daily data printing capability.

Now a day's most of the people have smart phones. They are not like to use desktop computer when they are travelling. Current system couldn't provide fully mobile responsive capability for all pages. System should be improved with any mobile responsive feature.

Current system is only supported for the patient surgery process of the hospital. System should be update with capturing all of the process of the hospital like medicine, clinic process, pharmacy and the staff management. Then hospital manage with fully computerize system and top management people easy to manage hospital and take decisions.

Reference

- [1] Hospital Management Information System Software Company India, Electra HIS, 2015" [Online]. [Online]. http://www.acgil.com/products/hospital_management_system.htm
- [2] Modules of Hospital Management System, 2017 [online]. Available:. [Online]. <https://solutiondots.com/blog/9-modules-hospital-management-system-developed-solutiondots-systems/>
- [3] eHospital - Best Hospital Management System Software (EHR Software) in India and USA, 2017 [online]. Available:. [Online]. <https://www.adroitinfosystems.com/products/ehospital-systems>
- [4] Bayanno Hospital Management System by Creativeitem | CodeCanyon, 2017 [online]. Available. [Online]. <https://codecanyon.net/item/bayanno-hospital-management-system/5814621>
- [5] SharePoint Capabilities, 2017 [online]. Available:. [Online]. https://www.tutorialspoint.com/sharepoint/sharepoint_capabilities.htm
- [6] 2017 [online]. Available: The Model-View-Presenter (MVP) pattern. [Online]. <https://msdn.microsoft.com/en-us/library/ff649571.aspx>
- [7] SDLC Quick Guide, 2017 [online]. Available. [Online]. https://www.tutorialspoint.com/sdlc/sdlc_quick_guide.htm
- [8] JQuery datapicker, 2017 [online]. Available. [Online]. <https://jqueryui.com/datepicker/>
- [9] Bootstrap v4 library, 2017 [online]. Available. [Online]. <https://v4-alpha.getbootstrap.com/getting-started/introduction/>
- [10] Master Pages and Site Navigatio, 2017 [online]. Available:. [Online]. <https://docs.microsoft.com/en-us/aspnet/web-forms/overview/data-access/introduction/master-pages-and-site-navigation-cs>
- [11] Easy Bootstrap Pagination for ASP.NET Gridview, 2017 [online]. Available:. [Online]. <https://www.codeproject.com/Tips/1085031/Easy-Bootstrap-Pagination-for-ASP-NET->

[Gridview](#)

[12] Unit Testing, 2017 [online]. Available. [Online].

https://www.tutorialspoint.com/software_testing_dictionary/unit_testing.htm

[13] Integration testing – Wikipedia, 2017 [online]. Available:. [Online].

https://en.wikipedia.org/wiki/Integration_testing

[14] Smoke Testing - Software Testing Fundamentals, 2017 [online]. Available. [Online].

<http://softwaretestingfundamentals.com/smoke-testing/>

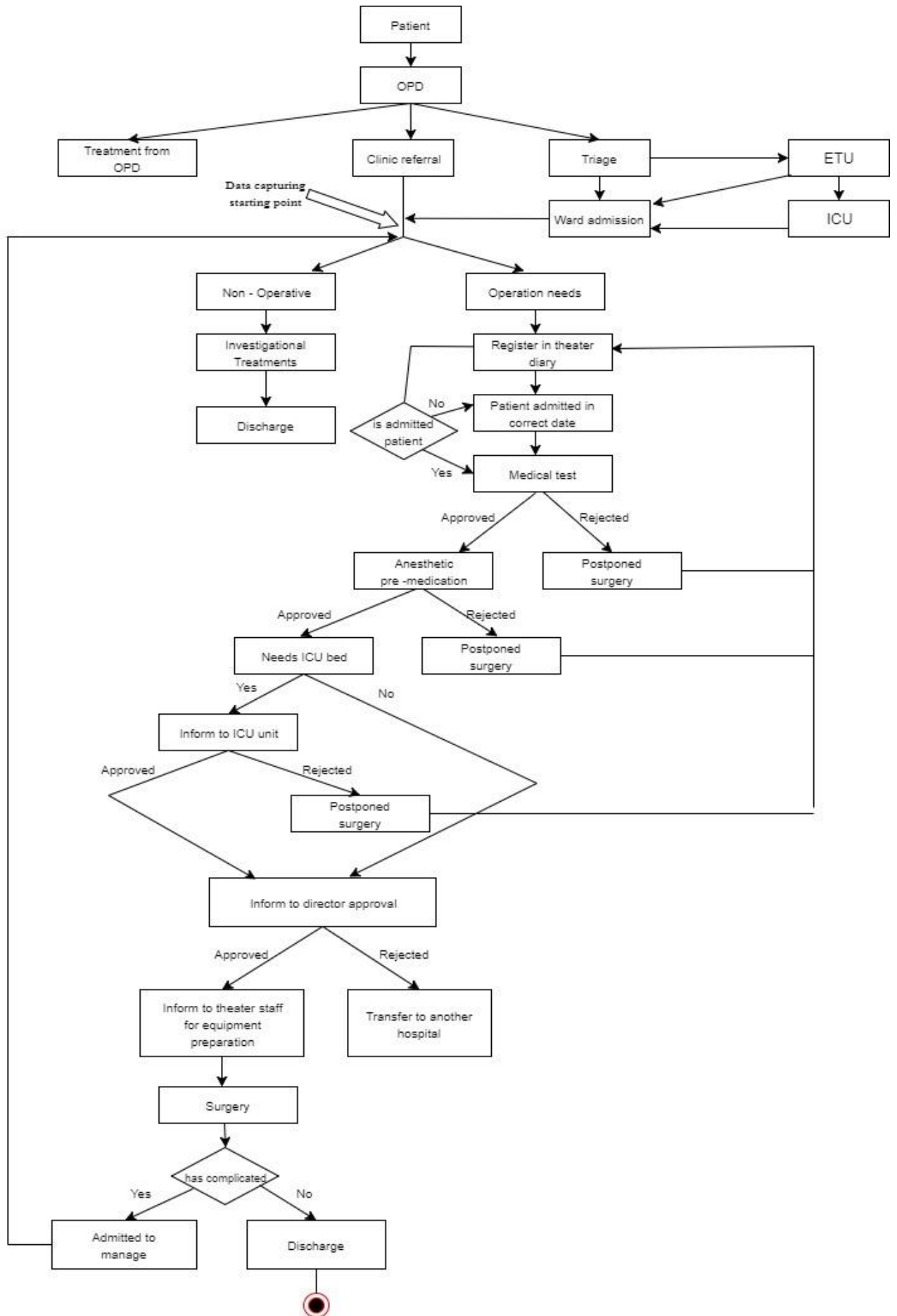
[15] Regression testing – Wikipedia, 2017 [online]. Available:. [Online].

https://en.wikipedia.org/wiki/Regression_testing

[16] What is User Acceptance Testing (UAT), 2017 [online]. Available:. [Online].

<https://www.guru99.com/user-acceptance-testing.html>

Appendix A - Process Flow Diagram



Appendix B - Password Encryption Logic

```
1 reference
public static string HashSHA1(string value)
{
    var sha1 = System.Security.Cryptography.SHA1.Create();
    var inputBytes = Encoding.ASCII.GetBytes(value);
    var hash = sha1.ComputeHash(inputBytes);

    var sb = new StringBuilder();
    for (var i = 0; i < hash.Length; i++)
    {
        sb.Append(hash[i].ToString("X2"));
    }
    return sb.ToString();
}
```

Appendix C – Test Cases

Test Report Document					
1. Introduction					
This section includes test cases carried out for the functionality testing of Patient Management for Operation Wards web application.					
2. Revision History					
Ver. No	Date	Created/Revised by	Sections Affected	Remarks	
0.1	10/01/2018	P.H.R.A Darshana	All	Internal Version	
3. Test Case Details					
SL.No	Function Name	Test Case	Expected Result	Pass / Fail	Actual Result / Remarks
Common					
01	User Login	Enter the user name, password and click the login button.	Should navigate to the home page.	Pass	Navigate to the home page.
02		Entering the valid user name and invalid password.	Should display the error message.	Pass	Display the error message.
03		Entering the invalid user name and the password.	Should display the error message.	Pass	Display the error message.

04		Entering the valid user name and invalid password.	Should display the error message.	Pass	Display the error message.
05		Displaying the login user name.	After login to the system, login user name should be display on top of the each and every form.	Pass	Login user name display on top of the each and every form.
06		Logout from the system.	When clicking the “Logout” link, should exit from process and navigate to the Login Page.	Pass	Logout form the system and redirect to the login page.
07	Patient Registration	Navigate to the patient registration page.	Patient Id should be display in the “patient id” field	Pass	Patient Id display in the “patient id” field.
08		Enter patient information without first name and click on the registration button.	System should display “First name is required” error message and stay on same page.	Pass	Display “First name is required” error message and stay on same page.

09		User tries to select future dates in the date of birth field.	System shouldn't display future dates in the Date of Birth calendar.	Pass	Doesn't display future dates in the Date of Birth calendar.
10		Click on the registration button with patient information.	Should be able to register new patient.	Pass	Register new patient successfully.
11	Patient Inquiry	Navigate to Patient inquiry page.	Should display all patient information in the page grid.	Pass	Display all patient information in the page grid.
12		Enter filter criteria and click on the filter button.	Should display patient information, related to the filter criteria	Pass	Display patient information, related to the filter criteria.
13		Without entering filter criteria, click on the filter button.	All patents information should be display in the grid.	Pass	All patents information in the grid.
14		Click on the view button in the patient information grid.	Particular patient information should be display in the patient update form.	Pass	Particular patient information display in the patient update form.
15	Patient Admit	User click on the "Admit" sub link under patient link in the main menu.	Patients admit form should be loaded successfully.	Pass	Patients admit form loaded successfully.
16		Find patient trough patient finder with valid patient ID.	Matching patient details should be display in the page	Pass	Matching patient details display in the page.
17		Find admitted patient using a search option.	Relevant patient information should be display in the form and	Pass	Relevant patient information display in the form and "Admit" button

			“Admit” button text should be change to the “Update”		text change to the “Update”
18		Find new patient using a search option and then select a ward.	Available number of beds should be display in the system.	Pass	Available number of beds display in the system.
19		Follow step 15. Fill ward details and click on the “Admit” button.	System should display successful message of the patient admitted.	Pass	System display successful message of the patient admitted.
20	Patient examine details	Find patient trough patient finder with valid patient ID.	Patient details with the treatment history details should be loaded successfully.	Pass	Patient details with the treatment history details loaded successfully.
21		Follow step 19. Filled illness and drug details and click on the “Add” button.	Patient treatment details should be added to the system and the grid view should be updated.	Pass	Patient treatment details added to the system and the grid view updated.
22		Follow step 19. Filled illness and drug details and click on the “Clear” button.	All editable fields should be clear.	Pass	All editable fields clear.
23		Follow step 19. Click on the “View” link in the grid view.	Selected patient history details should be display in the form.	Pass	Selected patient history details display in the form.
24		Follow step 19. Don’t fill drugs or illness details and try to click on the “Add” button.	System shouldn’t enable add button.	Pass	System disabled add button.
25	Patient Discharge	Find patient trough patient	Matching patient ID	Pass	Matching patient ID related details

		finder with valid patient ID.	related details should be filled in the form.		filled in the form.
26		Follow step 25. Click on the "Discharge" button.	Patient discharged and "Patient has been discharged" message should be display in the system.	Pass	Patient discharged and "Patient has been discharged" display in the system.
27		Follow step 25. Click on the "Cancel" button.	User should be navigated to the "Home" page.	Pass	User navigated to the "Home" page.
28	Employee Registration	Change employee type in the form.	According to the employee type, "Employee Id" should be changed.	Pass	According to the employee type, "Employee Id" changed.
29		Filled relevant form information in the system and click on the registration button.	Employee successfully registered in the system and successful message should be display.	Pass	Employee successfully registered in the system and successful message display.
30		Filled relevant form information without required field. Click on the register button.	System should be prevent registration and display "This field is required" message in front of the fields.	Pass	System prevent registration and display "This field is required" message in front of the fields.
31		Enter patient registration details with previously entered NIC.	System prevents employee registration and should be display error message.	Pass	System prevents employee registration and display error message.
32	Employee Inquiry	Select employee type is Doctor and enters ID related to the Director and	System should display message called "There is no matching records".	Pass	System display "There is no matching records" message.

		click on the “Search” button.			
33		Enter employee id partially and click on the search button.	Matching employee id related records should be displayed in the page grid.	Pass	Matching employee id related records displayed in the page grid.
34		Follow step 33. Click on the view link in the grid.	Employee details should be loaded in the edit form.	Pass	Employee details loaded in the edit form.
35		Follow step 34. Change patient information and click on the “Update” button.	Employee details should be updated successfully and display verification message.	Pass	Employee details updated successfully and display verification message.
36		Follow step 34. Click on the “Clear” button.	Clear all employee details and form should be ready for new patient registration.	Pass	Clear all employee details and form should be ready for new patient registration.
37	Surgery Registration	Enter patient id and click on the search button.	Patient basic information should be filled on the form.	Pass	Patient basic information filled on the form.
38		Follow step 37. Fill surgery date, time and theater details. (Previously reserved time entered). Then click on “Check availability” button.	Previous theater reservation details should be display in the grid view. Registration button should be disabled.	Pass	Previous theater reservation details display in the grid view. Registration button should be disabled.
39		Follow step 37. Fill surgery date, time and theater details. (Entered time doesn’t match with previous	Previous theater reservation details should be empty in the grid view and registration button should	Pass	Previous theater reservation details empty in the grid view and registration button enabled.

		reserved time). Then click on “Check Availability” button.	be enabled.		
40		Follow stem 39. Click on register button.	Patient should be register in the theater and success message display in the system.	Pass	Patient register in the theater and success message display in the system.
41		Follow step 30. Anesthetics navigated to the page through anesthetics approval link.	Only anesthetics approval form area should be enabled and other form details should be disabled.	Pass	Only anesthetics approval form area enabled and other form details disabled.
42		Follow step 30. Director navigated to the page through director approval link.	Only director approval form area should be enabled and other form details should be disabled.	Pass	Only director approval form area enabled and other form details disabled.
43		Follow step 30. Ward doctor navigated to the page through doctor approval link.	Only doctor approval form area should be enabled and other form details should be disabled.	Pass	Only doctor approval form area enabled and other form details disabled.
44		Follow step 41, 42, 43 and user navigate to the page.	All form fields should be disabled.	Pass	All form fields disabled.
45	Theater registration	Click on the “Theater” sub menu item under “Registration” main menu.	User should be able to navigate theater registration page and theater id should be display.	Pass	User navigate theater registration page and theater id display.

46		Follow step 45.Fill other theater details and click on the register button.	Newly added theater should be display in the theater grid view.	Pass	Newly added theater display in the theater grid view.
47		User click on the view link in the theater grid view row.	Selected theater details should be display in the form.	Pass	Selected theater details display in the form.
48		Follow stem 47. Modify description field and click on the update button.	Theater description should be updated and display in the gird view.	Pass	Theater description updated and display in the gird view.