# **Hospital Management System**

A project report submitted for the partial fulfillment of the requirement for the degree of Bachelor of Science in Computer science & Information Technology awarded by Tribhuvan University.

**Submitted By**

AAYUSH RANJIT (5-2-410-119-2018)

ANIL KUMAL (5-2-410-122-2018)

SHIRASH SHRESTHA (5-2-410-146-2018)

A Project Report Submitted in partial fulfillment of the requirement of **Bachelor of Science in Computer Science & Information Technology (BSC.CSIT)** 6th Semester of Tribhuvan University, Nepal.



# **ACKNOWLEDGEMENT**

We would like to express our deepest appreciation to all those who provided us the possibility to complete this report. A special gratitude we give to our project Supervisor, **Mr.Sravan Ghimire**, whose contribution in stimulating suggestions and encouragement, helped us to coordinate our project especially in writing this report. Furthermore, we would also like to acknowledge with much appreciation the crucial role of the staff of Prime College, who gave the permission to use all required equipment and the necessary materials to complete the task. We are thankful and fortunate enough to get constant support from our seniors and every teaching staff of BSc.CSIT department which helped us successfully complete our project. We would also like to extend our regards to all the non-teaching staff of B.Sc. CSIT department for their timely support. We appreciate the guidance given by other supervisor as well as the panels especially in our project document that has improved our standard of report. Our thanks and appreciations also go to each and every one of our colleagues for their encouragement and support in developing the project.

At last, we would like to extend heartfelt thanks to our parents because without their help this project would not have been successful. Finally, we would like to thank our dear friends who have been with us all the time. Any omission in this brief acknowledgement doesn’t mean lack if gratitude.

With respect,

Aayush Ranjit (5-2-410-119-2018)

Anil Kumal (5-2-410-122-2018)

Shirash Shrestha (5-2-410-146-2018)

# **ABSTRACT**

Hospital Management System (HMS) is the project that aims to introduce a Doctors online appointment booking service for healthcare services that would ease off the appointment-scheduling journey for users and pave the path of a better doctor-patient experience. The proposed system advances with online facilities that eliminate the chaos of the traditional appointment services. This system offers online appointment booking, to view doctor list, to cancel and update appointments with an administrative portal to manage all the sections. It can be maintained in real time. It provides advanced functionality to streamline the process thus easy access to personal hospital services that help organization to connected with their customers, clients and most importantly patients and can result in significant time and monetary savings. Thus, HMS project is proposed as an important component of scheduling and managing appointments.

**Keywords :**

**Table Of Contents**

[**Hospital Management System** 1](#_Toc118050204)

[**ACKNOWLEDGEMENT** 2](#_Toc118050205)

[**ABSTRACT** 3](#_Toc118050206)

[**LIST OF TABLES** 6](#_Toc118050207)

[**LIST OF FIGURES** 7](#_Toc118050208)

[**LIST OF ABBREVIATIONS** 8](#_Toc118050209)

[**Chapter 1** 9](#_Toc118050210)

[**INTRODUCTION** 9](#_Toc118050211)

[**1.1** **Background** 9](#_Toc118050212)

[**1.2** **Problem Definition** 10](#_Toc118050213)

[**1.3** **Objectives** 10](#_Toc118050214)

[**1.4** **Scope** 10](#_Toc118050215)

[**1.5** **Limitations** 11](#_Toc118050216)

[**1.6** **Process Model** 11](#_Toc118050217)

[**CHAPTER 2** 12](#_Toc118050218)

[**REQUIREMENT ANALYSIS AND FEASIBILITY ANALYSIS** 12](#_Toc118050219)

[**2.1** **Literature Review** 12](#_Toc118050221)

[**2.1.1** **Study of Previous Literature** 12](#_Toc118050222)

[**2.1.2** **Study of Existing Literature** 12](#_Toc118050223)

[**2.2** **Requirement Collection Methods** 13](#_Toc118050224)

[**2.2.1** **Source of data** 13](#_Toc118050225)

[**2.3** **Requirement Specification** 13](#_Toc118050226)

[**2.3.1** **Functional Requirements** 13](#_Toc118050227)

[**2.3.2** **Non-functional Requirements:** 16](#_Toc118050228)

[**2.4** **Feasibility Analysis** 17](#_Toc118050229)

[**2.4.1** **Technical Feasibility** 17](#_Toc118050230)

[**2.4.2** **Operational Feasibility** 18](#_Toc118050231)

[**2.4.3** **Economic Feasibility** 18](#_Toc118050232)

[**2.4.4** **Scheduling Feasibility** 19](#_Toc118050233)

[**CHAPTER 3** 24](#_Toc118050234)

[**SYSTEM DESIGN** 24](#_Toc118050235)

[**3.1 System Architecture and Overview** 24](#_Toc118050236)

[**3.2 System Design** 25](#_Toc118050237)

[**3.2.1 UML diagram** 25](#_Toc118050238)

[**CHAPTER 4** 29](#_Toc118050239)

[**System Implementation and Testing** 29](#_Toc118050240)

[**4.1 Implementation Overview** 29](#_Toc118050241)

[**4.1.2 Tools Used** 29](#_Toc118050245)

[**4.1.2.1 Front End Tools** 29](#_Toc118050246)

[**4.1.2.2 Back End Tools** 30](#_Toc118050247)

[**4.2 Modules Description** 30](#_Toc118050248)

[**4.3 Testing** 31](#_Toc118050249)

[**4.4.1 Unit Testing** 31](#_Toc118050250)

[**4.4.2 System Testing** 31](#_Toc118050251)

[**CHAPTER 5** 32](#_Toc118050252)

[**CONCLUSION AND RECOMMENDATION** 32](#_Toc118050253)

[**5.1 Conclusion** 32](#_Toc118050254)

[**5.2 Recommendation** 32](#_Toc118050255)

[**5.3 Future Scope** 33](#_Toc118050256)

[**BIBILOGRAPHY/REFERENCES** 34](#_Toc118050257)

[**APPENDIX (Screenshots)** 35](#_Toc118050258)

# **LIST OF TABLES**

Table 1: Technical feasibility......................................................................................... 9

Table 2: Gantt Chart ......................................................................................................10

# **LIST OF FIGURES**

[Figure 1: Spiral Model 3](#_Toc118027872)

[Figure 2:Use case diagram for user registration 7](#_Toc118027873)

[Figure 3:Use case diagram for registered user login 8](#_Toc118027874)

[Figure 4:Use case diagram of updated information of registered users 9](#_Toc118027875)

[Figure 5: System Architecture 15](#_Toc118027876)

[Figure 6: Database Schema 17](#_Toc118027877)

[Figure 7: Class Diagram 19](#_Toc118027878)

[Figure 8: Sequence Diagram 20](#_Toc118027879)

[Figure 9: Activity Diagram 21](#_Toc118027880)

# **LIST OF ABBREVIATIONS**

CSS Cascading Style sheets

HMS Hospital Management System

HTML Hyper Text Markup Language HTTP

# **Chapter 1**

# **INTRODUCTION**

## **Background**

“The Human Body is The Best Picture of Human Soul” Said by a famous author Tony Robbin. In this busy life people value money and time more than their health. In developed country, people have routine checkup, care about their health more than anything. But in Nepal people find difficult to do regular/routine checkup. And it is a major problem in most of all cities of Nepal. In rural places people don’t even know whom to visit and where. And when they know they have to spend a lot of time waiting for bookings, check-up. Due to this, most of the people go to medical clinic only if there is emergency.

In Kathmandu the scenario is different. Mostly students and workers live here. They are so much busy in their life that they even don’t know how many clinics, medical stores or hospitals are around them. Therefore, this is the situation of health status inside and outside the city, which results in a poor health. Although it may seem a common problem now, but it can be a major problem in future. On the contrary, a viable solution would be to optimize the use of available technology in such a way that the public is able to choose a better health care, faster, easier, and in convenient way.

The project HMS is a community-based healthcare web application for the windows platform that enables user to book an appointment with doctor, have routine check-up and so on. The aim of this application is not just book but also to check the availability of doctors, check medical clinics, health post around them. This application books an appointment with doctor and when the user books an appointment S/he gets a coupon ticket in sequence.

This application also sends a reminder notification/message to user 1hour before the appointment starts. Here this application allows user to search the clinics/medical store/hospital near them using their location. The user also can check the availability of the doctors.

## **Problem Definition**

Human’s three basic need are food, shelter, clothes. And to obtain those needs we valued money more than health. Health is not valued till sickness comes. That’s the mistake of our life. Mostly in under-developed countries health care service are very poor. Likewise, health care services, hygiene, sanitation in Nepal are of inferior quality and fail to reach a large portion of population, particularly in rural areas. The poor have limited access to basic health care due to high costs, low availability, lack of health education and conflicting traditional benefits.

Most of the time we are unknown about the presence of specialized doctors in our nearby clinic, hospitals, medical stores by which we do not get chance to consult with them. So, we are not able to have an appointment with them. In this way we are lacking medical services.

Going through all of these problems the project HMS is an application that is introduced to give an idea to connect with doctors, book an appointment or consult with them, check if the doctors are available or not which helps them to find the best health service facility.

## **Objectives**

This project aims to meet the following objectives:

1. To make easy to book an appointment with related doctors.
2. To create a connection between doctors and patients.
3. To check the availability of the related doctors.
4. To provide a better health care services to user/patient.

## **Scope**

This application registers doctors using their citizenship and doctor license, which means any doctor of Nepal, with a citizenship and doctor license is able to use the application. Whereas for patient, by filling up the form they are easily able to use the application. This application is not primarily monetary purpose however maintenance and support cost are balanced by few percentages of booking and some ad-works. Application has provided a messaging facility with doctors which create a friendly bond between them, where they can consult privately.

However, this doesn’t mean that the system cannot be used to book appointment for someone else who want to check-up or consult with doctors.

## **Limitations**

The Limitations of this project can be listed out as follows:

1. Internet connection is needed.
2. Not so suitable for emergency cases.
3. Doctors may not be available at required time.
4. Different specialist may not be available/ may not use the app where patients may find hard to get particular service.

## **Process Model**

Spiral model is one of the most important Software Development Life Cycle models, which provides support for Risk Handling. In its diagrammatic representation, it looks like a spiral with many loops. The exact number of loops of the spiral is unknown and can vary from project to project. Each loop of the spiral is called a Phase of the software development process.



Figure 1: Spiral Model

# **CHAPTER 2**

# **REQUIREMENT ANALYSIS AND FEASIBILITY ANALYSIS**



## **Literature Review**

### **Study of Previous Literature**

Appointment scheduling has become a complex task especially for healthcare professionals in hospitals and clinics. Few reasons that could cause these complications range from a heavy flow of patients traffic to a physician that practices in a number of clinic and moves from one medical facility to others.

An ineffective appointment management could cause overlapping appointments, rise in number of no-shows, patient dissatisfaction in general and revenue loss for healthcare institutions. Traditionally, medical appointments have been made with schedulers over the telephone or in person. These methods are based on verbal communications with real people and allow for maximum flexibility in complicated situations. However, because these traditional methods require the intervention of schedulers, the ability to get a timely appointment is not only limited by the availability of appointment slots, but also by the schedulers and phone lines. Patients’ satisfaction with appointment booking is influenced by their ability to book at the right time with the right health service providers.

### **Study of Existing Literature**

Nowadays the appointment system has been a bit modernized. Many medical institutions use computerized appointment scheduling and also followed by phone-based appointment system. It has also helped people in many ways as compared to traditional scheduling system.

The Internet has recently emerged as another means to make appointments. Different countries have already been using online doctor appointment system. Web-based appointment scheduling is an extremely important feature, and most patients would use the service. In comparison, to paper-based appointment scheduling, web-based appointment scheduling is faster that allows multiple user access at any given time. In recent days, many medical institutions use a combination of phone-based scheduling and computerized appointment scheduling. So, this combo along with out-sourcing services make a better efficient system by allowing patients to access their health information and communicate with their health care providers at any time.

## **Requirement Collection Methods**

### **Source of data**

* **Content Analysis**

When analyzing the most successful projects we found that they were based on the real-time necessity of the population. The current situation analysis showed that the medical field was in need of advanced digital applications and reliable information for the convenient use of a large population.

A cognitive perspective of the common population showed that they considered factors like time taken for booking, payment and being updated about the situation for convenience and satisfaction. These were some important factors that affected their intentions which were taken into consideration while designing the project.

## **Requirement Specification**

### **Functional Requirements**

**• User Authentication**

The system shall authenticate the user and then display panel based on the particular identified user.

**• Keyword matches for efficient search**

The system shall provide an efficient search feature based on keywords.

**• Listing of doctors according to their specialty**

The system shall rank all the doctors based on the specialty.

**Use Case Diagram**

#### Register User

The system shall authenticate the user after registering to the system with required email address and password.

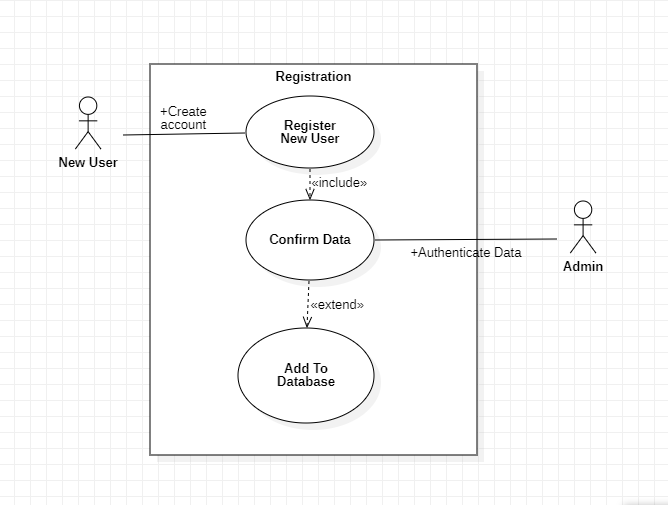


Figure 2:Use case diagram for user registration

Use case scenario for registering user

#### Login

After successful registration, user shall be able to login the system.

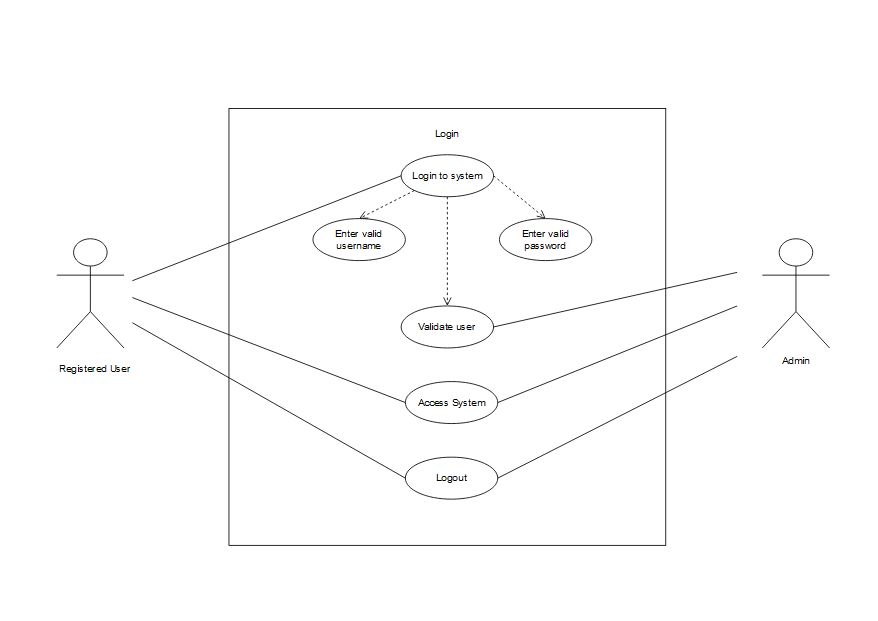


Figure 3:Use case diagram for registered user login

#### Update data

Registered user can update their information.

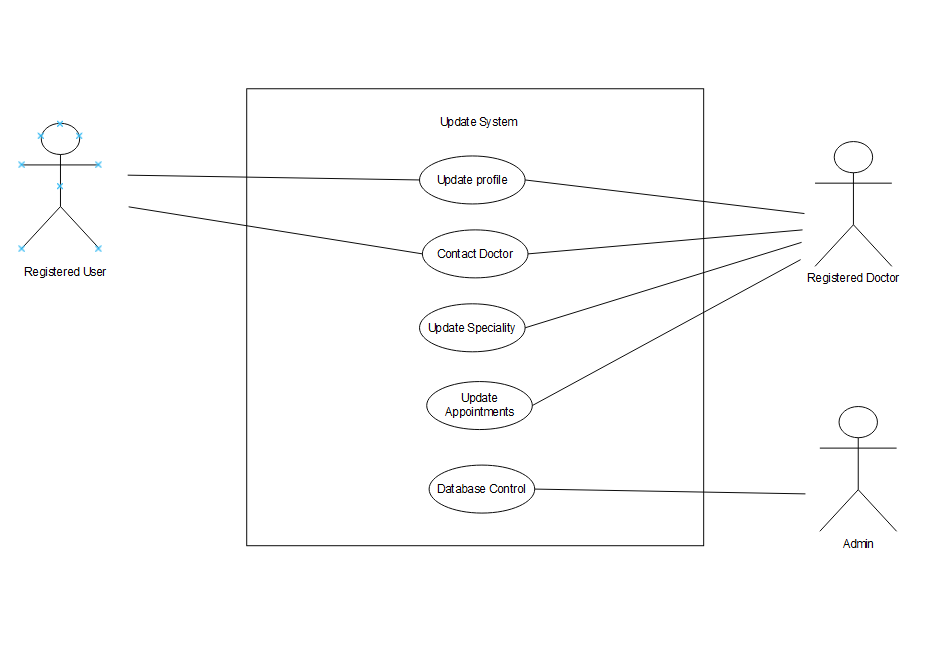


Figure 4:Use case diagram of updated information of registered users

### **Non-functional Requirements:**

**• Reliability:**

All the data including the user’s and doctor’s information are stored in the database and are under the authentication of the admin. So, all the transactions of

data are safe and reliable.

• **Time-consuming:**

It is very important that people get immediate access to doctors and services. HMS provides a wide variety of qualified and experienced doctors for the patient's convenience and scheduled time.

• **Usability:**

With the provided user-interface, it is very simple to use and all one needs to do is click and fill the form to book the preferred doctor or contact them directly as well.

**• Concurrency:**

Since HMS is a real-time application, it has the potential to handle multiple computations and should execute simultaneously for each of the users. And the users can easily get access to doctor’s information and vice versa.

## **Feasibility Analysis**

A feasibility study is the study of the project at its initial stage and it shows if the proposed project can be practically implemented in real-time or not. It is judged on the basis of finance, manpower, management, time and whether it is worth the investment. All these factors are analyzed and studied to check if the system can be built efficiently with the available resources.

### **Technical Feasibility**

The technical feasibility study focuses on gaining an understanding of the present technical resources, required technical expertise, their applicability to the expected needs of the proposed system and the tools and software required for the overall implementation of the project. It requires a web server and a database management system in which it can operate and it should be easily affordable. The tools and software required for this project are easily available on the internet since most of the software and tools are open source. However, the implementation of the algorithm and the calculations may seem complex.

Table 1: Technical feasibility

|  |  |
| --- | --- |
| Software Requirements | Hardware Requirements |
| Desktop View | Desktop View |
| Windows, IOS or any other operating system  which supports internet. | Computer, laptop, smartphones with internet access |

### **Operational Feasibility**

Operational feasibility quantifies whether the proposed project satisfies the requirement or can solve the defined problem that has been proposed during the analysis. This application is a web-based system and users with knowledge regarding basic web-based applications can easily use through the platform with ease.

### **Economic Feasibility**

Economic feasibility quantifies the project’s costs and revenue in an effort to determine whether it is possible to complete or not. A good web hosting service is necessary for a fully working application. As the proposed project requires no extra equipment to be bought and the required needs are already present, so the system is economically feasible. Maintaining an extensive database, which would be required as the number of users of the app starts increasing. Additionally, it would require marketing and advertising costs through the internet which is comparatively cost-effective.

### **Scheduling Feasibility**

Table 2: Gantt chart

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Activities | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 |
| Requirement  Gathering and Analysis |  |  |  |  |  |  |  |  |  |  |  |  |
| Design |  |  |  |  |  |  |  |  |  |  |  |  |
| Coding |  |  |  |  |  |  |  |  |  |  |  |  |
| Testing |  |  |  |  |  |  |  |  |  |  |  |  |
| Implementation |  |  |  |  |  |  |  |  |  |  |  |  |
| Documentation |  |  |  |  |  |  |  |  |  |  |  |  |

The above diagram represents the time taken for the completion of the project and the time consumed by each activity in the following days respectively. It shows that the gathering and analysis took about 3 days and similarly the time taken by design, coding, testing, implementation, and documentation is shown in the above Gantt chart. The documentation part has consumed more time as compared to others.

# **CHAPTER 3**

# **SYSTEM DESIGN**

## **3.1 System Architecture and Overview**

A system architecture or systems architecture is the conceptual model that defines the structure, behavior, and more views of a system. An architecture description is a formal description and representation of a system, organized in a way that supports reasoning about the structures and behaviors of the system. A system architecture can consist of system components and the sub-systems developed, that will work together to implement the overall system.

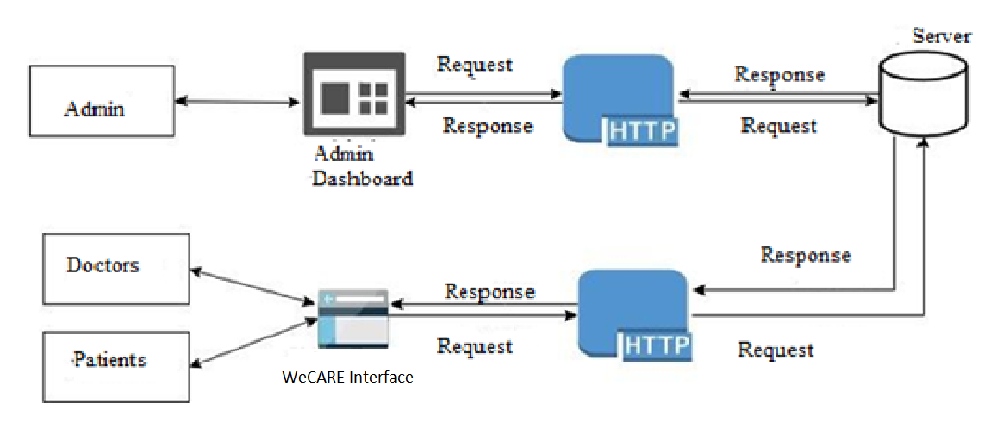


Figure 5: System Architecture

For our proposed project, we developed the required system that works with the help of the internet. Since our project works with the help of the internet, it runs on any devices that have access to the internet. To use this system, we just need a user that has an understanding of the system and the internet. The above architecture shows how different users exchange information or message through the interaction of Web services. This handles real-time data so it provides real-time syncing to the clients subscribed to the server at any given instance.

## **3.2 System Design**

System Design includes the means and methodologies to improve the management and control of the software development process. It is the process of defining the elements of a system such as the architecture, modules and components, the different interfaces of those components and the data that goes through that system. It is meant to satisfy specific needs and requirements of a business or organization through the engineering of a coherent and well-running system.

### **3.2.1 UML diagram**

A UML diagram is a diagram based on the UML (Unified Modeling Language) with the purpose of visually representing a system along with its main actors, roles, actions, artifacts or classes, in order to better understand, alter, maintain, or document information about the system.

The UML diagrams below includes different types of diagrams incorporating the structure and behavior of the system. The diagrams designed to develop the system are shown below.

#### 3.2.1.1 Class Diagram

A class diagram in the Unified Modeling Language (UML) is a type of static structure diagram that describes the structure of a system by showing the system's classes, their attributes, operations (or methods), and the relationships among objects.

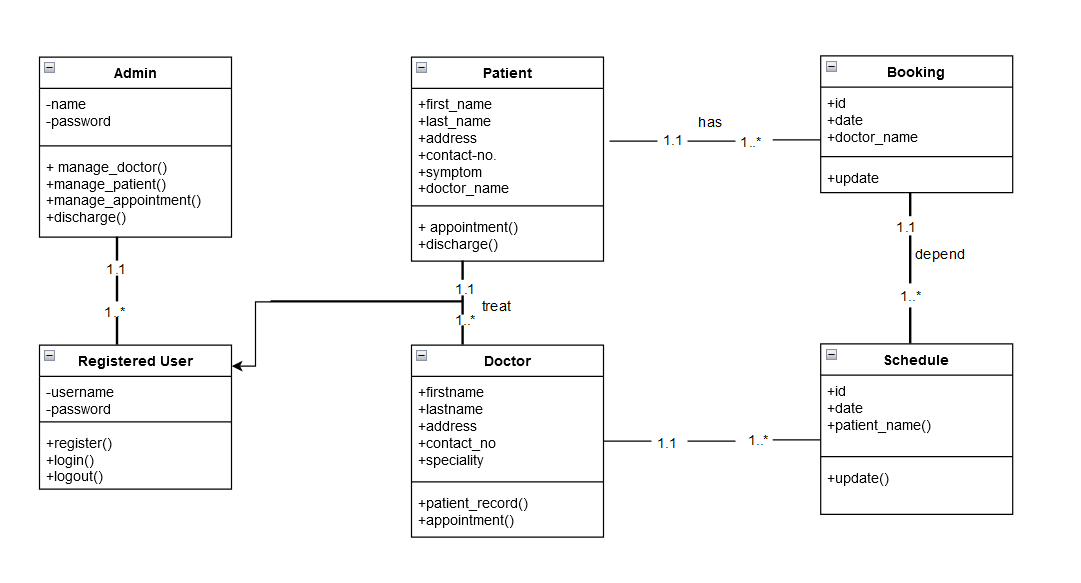


Figure 7: Class Diagram

#### 3.2.1.2 Sequence diagram

Sequence Diagrams are interaction diagrams that detail how operations are carried out. They capture the interaction between objects in the context of a collaboration. Sequence Diagrams are time focus and they show the order of the interaction visually by using the vertical axis of the diagram to represent time what messages are sent and when.

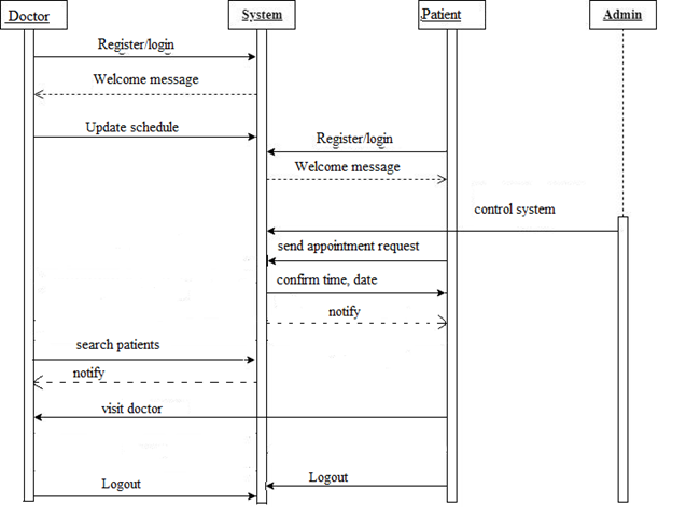


Figure 8: Sequence Diagram

#### 3.2.1.3 Activity Diagram

Activity diagram is another important behavioral diagram in UML diagram to describe dynamic aspects of the system. Activity diagram is essentially an advanced version of flow chart that modeling the flow from one activity to another activity.

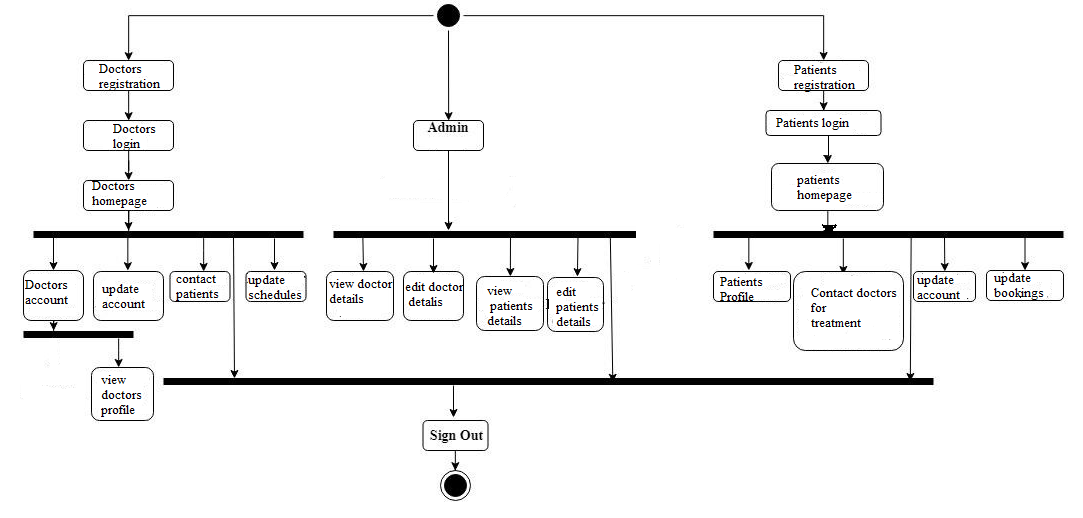
****

Figure 9: Activity Diagram

# **CHAPTER 4**

# **System Implementation and Testing**

## **4.1 Implementation Overview**

**4.1.1 Process Model used**

Spiral Model is a combination of a waterfall model and iterative model. Each phase in spiral model begins with a design goal and ends with the client reviewing the progress as this project is not dealing with the clients and hence beforehand planning can be made about how to carry out each phase of development. The development team starts with a small set of requirements and goes through each development phase for those set of requirements. The software engineering team adds functionality for the additional requirement in every new spiral until the application is ready.

We are using the Spiral Model because as additional functionality or changes further in the project can be integrated and as we have to meet a specific deadline every week.



### **4.1.2 Tools Used**

### **4.1.2.1 Front End Tools**

**HTML5**: HTML or Hyper-Text Markup Language can be referred to as the Worldwide Web’s primary language. Most of the web pages hosted on the internet are written in some variation of HTML. HTML has seen many updates over time, and currently, the newest HTML version is HTML5. HTML5 provides full support for JavaScript to run in the background this is possible courtesy to the JS web worker API of HTML5. There is no use to employ any JS-based or Flash work-around because there are elements inherently present in HTML5 that provide all the functionalities.

**CSS3**: CSS3 is the advance level of CSS2.0. In CSS3 we have new properties - border radius, box shadow, text shadow, multiple background images and much more. CS3 supported by all new browsers. The CSS3 version supports many more browsers than CSS2.

**BOOTSTRAP**: Bootstrap is compatible with all modern browsers (Chrome, Firefox, Internet Explorer, Safari, and Opera). Bootstrap offers a lot of helper classes that make development of a responsive website easy and fast. Bootstrap’s components are well- adopted to the ecosystem of popular JS MVC Frameworks like Angular. Bootstrap provides several ways to include it into your project: Bootstrap is designed to work with latest desktop and mobile browsers.

### **4.1.2.2 Back End Tools**

**Django**: Django is a high-level Python web framework that encourages rapid development and clean, pragmatic design. Built by experienced developers, it takes care of much of the hassle of web development, so you can focus on writing your app without needing to reinvent the wheel. It’s free and open source.

**SQLite**: SQLite is a C-language library that implements a small, fast, self-contained, high-reliability, full-featured, SQL database engine. SQLite is the most used database engine in the world. SQLite is built into all mobile phones and most computers and comes bundled inside countless other applications that people use every day.

## **4.2 Modules Description**

A modularization consists of well-defined manageable units with well-defined interfaces among the units. Desirable property of modular system include:

a) Each module is a well-defined sub-system.

b) Single, well – defined purpose of each module.

c) Modules can be separately compiled and stored in a library. D) Modules can use other modules.

e) Modules should be easier to use than to build.

f) Modules should be simpler from outside then from inside.

The application comprises the following major modules:

**Doctor Module:**

This module provides functionalities for doctors.

• Applicants can post their professional details.

• The applicants can browse and edit their user profiles.

• Doctors have the right to accept or reject any patients.

**Patient Module:**

This module provides functionalities related to patients.

* Patients can browse and edit their profiles.
* Patients can search for doctors.
* Patients can establish contact with doctors

**Administrator Module:**

This module provides administrator related functionalities.

* Administrator manages entire application.
* Administrator can view and edit the profiles of doctors and patients.

## **4.3 Testing**

Testing is the process of evaluation a software item to detect differences between given input and expected output. Testing is a process that should be done during the development process.

### **4.4.1 Unit Testing**

The Unit testing part of a testing methodology is the testing of individual software modules or components that make up an application or system.

### **4.4.2 System Testing**

The system testing part of a testing methodology involves testing the entire system for errors and bugs. This test is carried out by interfacing the hardware and software components of the entire system, and then testing it as a whole.

# **CHAPTER 5**

# **CONCLUSION AND RECOMMENDATION**

## **5.1 Conclusion**

The HMS has been designed aiming to introduce the online platform for medical services in Nepal. The project has proposed an online doctor booking system built on the web service architecture. The system is designed to achieve maximum user satisfaction. This system achieves functional capabilities with the latest technology methodologies used in the market today. One of the prime reasons for the proposed of this project is to provide a platform for users to help book appointment of doctors and save time of the patients. It holds an edge over traditional booking methods as people get a confirmed appointment and the payment can be done online without the need to stand in long queues at the hospital. This project also helps to create a connection between doctors and patients, and help to check the availability of the related doctors and also to provide better health care services to patients.

## **5.2 Recommendation**

Compared with traditional appointment methods, Web-based appointment scheduling is more patient-centered and has many advantages due to improved access. One of the main reasons people in our country hesitate to have regular/routine checkup is high costs, low availability, lack of health education and conflicting traditional benefits. Hence, HMS is a project based off of solving and creating a new digital era in medical field. People acknowledge services and facilities that are provided digitally and tend to continue using them. HMS not only encourages digitalization of medical field but also spreads awareness and easy access of health-related services which has become more than just necessary considering that Kathmandu is listed as one of the most populated cities in the world. It is highly recommended as the platform helps to book appointment of doctors and save time of the people in this busy era. As this website functions within the Internet, anyone with the knowledge of Internet can use the services.

## **5.3 Future Scope**

The proposed system limits the user as the system is a simple online doctor booking system. User are allowed to make appointments and make online deposit and the admin is only capable of accessing patients records and registering new doctors. There is a huge scope of enhancement and adding functionalities to this website. In term of such functionalities, the system lacks behind from hospital management system. It does not generate reports, it only generates appointment status of the patients.

# **BIBILOGRAPHY/REFERENCES**

*Django documentation | Django documentation | Django*. (2019). Djangoproject.com. <https://docs.djangoproject.com/en/3.0/>

Holzner, S. (2009). *HTML black book*. Dreamtech Press.

*Hospital Management System: All-in-One Guide*. (n.d.). Www.uptech.team. <https://www.uptech.team/blog/develop-hospital-management-system>

*Software Analysis & Design Tools - Tutorialspoint*. (n.d.). Www.tutorialspoint.com. <https://www.tutorialspoint.com/software_engineering/software_analysis_design_tools.htm>

*SQLite Tutorial - An Easy Way to Master SQLite Fast*. (n.d.). SQLite Tutorial. <https://www.sqlitetutorial.net/>

# **APPENDIX (Screenshots)**

