UNIVERSITY PARTNER

UNIVERSITY OF WOLVERHAMPTON

HERALD COLLEGE KATHMANDU

Project and Professionalism (6CS020)

A1: Project Proposal

CHATBOT IN HEALTHCARE

Student Id : NP03A180039
Student Name : Shradha Limbu
Supervisor : Mr. Sachin Kafle
Cohort/Batch : 4

Submitted on : 16-10-2020

ACKNOWLEDGEMENT

I would like to express my gratitude towards our supervisor Mr. Sachin Kafle for his guidance throughout this proposal writing. Since this is is the beginning of the Final Year Project, I am looking forward to his guidance throughout the end of the project.

ABSTRACT

The increase in healthcare and hospital medical services has been shifting from a treatment focus on prevention and health management. The medical industry has been creating additional services for healthcare that includes research-based on Artificial Intelligence healthcare equipment. Among various advanced equipment introduced in the health sector, chatbots are one of those revolutions. A chatbot is a software that can communicate with human beings using various Artificial Intelligence and natural language processing algorithm.

The project Hospital Management System is a web-based application that will provide user management, appointment booking, transaction management, and chatbot services to the user. The name of the application is Sorabot. Sorabot is a system that will implement deep learning generative chatbot to interact with the user through text. The chatbot will provide a symptom checker and emergency first aid service to the user. This chatbot will interact with the user through the message by asking a series of questions about the symptoms and help the patient to find their disease. Sorabot will use one of the most advanced deep learning algorithms for chatbot called Long Short Term Memory. Sorabot will be build based on the Scrum framework that will track the overall progress of the project. Work Breakdown Structure and Gantt chart will be used to identify work packages and a time duration of six months for overall project completion.

TABLE OF CONTENTS

TABLE OF FIGURES

TABLE OF ABBREVIATION

Abbreviation	Explanation
AI	Artificial Intelligence
UML	Unified Modeling Language
API	Application User Interface
URL	Uniform Resource Locator
LSTM	Long Short Term Memory
GRU	Gated Recurrent Unit

SECTION 2: STATEMENT OF PROJECT DETAILS

1. Project Title

The project title is chatbot in healthcare with deep learning neural networks. The name of the system is Sorabot.

2. Academic Question

The academic question related to the project are:

i. What is the business problem the proposed project is trying to solve?
ii. What will be the learning outcome that will be achieved from this project on project management?
iii. Will this project be able to meet the advanced technology in the field of healthcare.
iv. What are the social and ethical issues to take into consideration?

3. Aims

The main aims of the hospital management system are:

i. The project aims to provide and achieve high-quality healthcare services.
ii. The integration of chatbot in the system will provide 24 hours online service to patients.
iii. The chatbot will provide emergency first aid and symptom checking features to patients.
iv. Patients can also book an appointment with appropriate physicians in the system.
v. All the data of the patients will be securely stored in the database.

4. Objectives

To achieve the aims of this project, some of the important objectives to carry out during the project phase are:

i. The most important foundation of the AI project is dataset. Since it is related to health, some of the health-related data can be sensitive. Data collection, data cleaning, and data pre-processing will be used to produce a clean dataset for increasing the performance accuracy of the chatbot.
ii. To complete the project at a specified time, the Scrum methodology will be followed to produce a continuous artifact.
iii. The accuracy of chatbot in predicting symptoms will be achieved by training and testing datasets with deep learning algorithms model, LSTM model.
iv. In the hospital management, to store the patient's data securely, token-based authentication of Django will be used.

5. Smart Objectives

i. Start the project with features of web application such as user authentication, appointment booking which I am familiar with in order to slowly gain the insight and focus into the project.
ii. Complete features of web application which covers almost 50% of the total project by the end of fifth semester.
iii. Everyday motivating and challenging myself to focus on project
iv. Planning, implementation and testing the project according to the system diagrams and requirement specification document to avoid any errors.

v. Work side-by-side with project as well as documentation with time specified in gantt chart in order to keep project in track.

6. Artifact

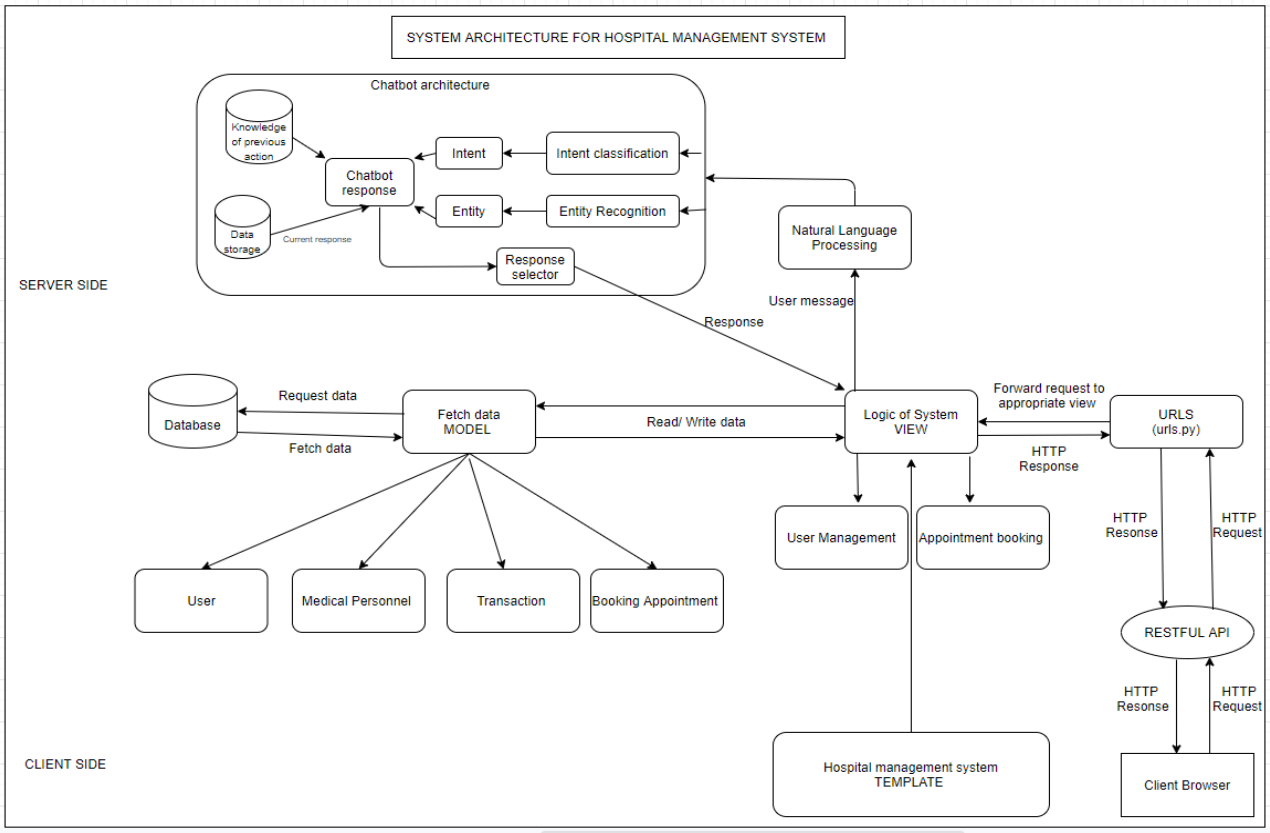


Figure 1: System Architecture of Hospital Management System.

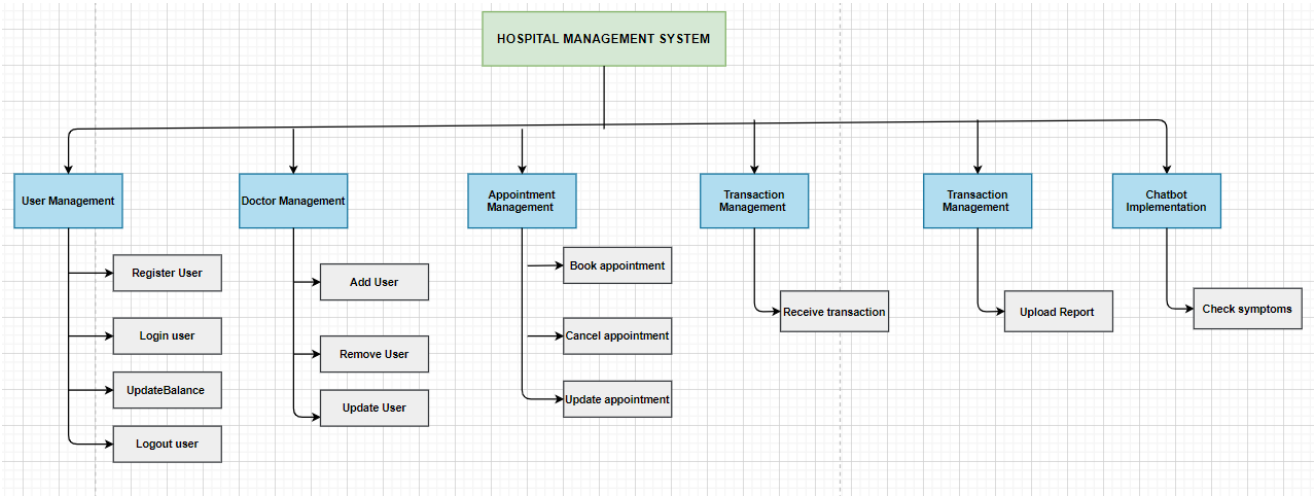


Figure 2: Functional Decomposition Diagram of Hospital Management System.

SECTION 3: PROJECT PROPOSAL (MAIN SECTION OF THE REPORT)

1. Introduction

Chatbots are the complex intelligence system that is built to engage and interact automatically with users through text or voice interface using various artificial intelligence, natural language processing, big data analytics, and machine learning algorithms. (Projects, nevon;, 2019). Deep learning is the major driving force behind the advancement of the chatbot. Deep learning helps to exploit the powerful neural networks to solve complex computational problems (Deng & Liu, 2018).

The hospital management system is a web-based application project with Artificial Intelligence in the form of chatbot. The hospital management system will provide a few of the most important services that every user seeks in healthcare. The main subsystem of Hospital management consists of user management, doctor management, appointment booking, transaction management, report management, and chatbot. The chatbot will provide symptom checking service to the user.

The deep learning neural network called the Long-Short Term Memory model is integrated into the chatbot. The chatbot uses Natural Language Processing (NLP) for intent identification to understand the intent of user input, and entity recognition to extract the key information from the input (Sreelakshmi, et al., 2018). The extracted information is then mapped with a deep learning LSTM model to perform complex computation problems like analyzing the input to produce a reasonable response. This proposed chatbot is a supervised learning chatbot which works in labeled. In supervised learning techniques, the model learns from a large number of labeled datasets to make a prediction based on pattern matching (Bargagli-Stoffi, et al., 2020).

2. Background Research

2.1. Problem Domain

Poverty is one of the largest barriers to quality healthcare access in Nepal. Nepal has 0.3% of doctors and nurses per 1000 patients which is 2.3% less compared to medical personal recommended by WHO (World Health Organization). This issue has been leading to unsafe and poor quality health services in Nepal. Due to a lack of guidance and consultation, patients often seek advice and treatments from inappropriate medical personnel and take medical precautions from local pharmacies (Garha, 2017) (Sennaar, 2019). The interference in proper and earlier diagnosis of diseases will cause a significant chance of infection and errors in healthcare leading towards poor health, disease becoming more severe, and sometimes causing death (Sennaar, 2019) .

2.2.The project as a Solution

Hospital Management System will provide all the major service that the user seek from the hospital. The user will be able to seek medical service from a chatbot and find an appropriate physician from the system. Chatbots in healthcare is a great advanced solution to provide proper health access to patients. Sorabot will be one of that deep learning chatbot that will help the patient discover their disease by understanding their symptoms and help them to discover their disease (neveonProjects;, 2020). Sorabot will

also provide frequently asked questions related to emergency first aid treatment. Sorabot will use a model trained with deep learning algorithms to interact with the patients to get an accurate analysis of their symptoms. This chatbot will also provide answers to all the important inquiries from the patients thus guiding them to the appropriate service (WelcomeAI;, 2020).

Sorabot will engage and improve the overall patient's experience with or without the need for customer support teams or physicians by providing healthcare support 24 hours online (neveonProjects;, 2020). The chatbot can also decrease medical costs and improve health facilities.

3. Description to Academic Question

Below is the justification for the academic question :

i. [i] The project, Hospital management system with the chatbot is proposed to provide people with better healthcare services. The reason to integrate a chatbot in the system is to provide symptom checking service to users so that they can seek medical advice faster and easily.

ii. [ii] The modern project management approach, Scrum will be followed from planning to development to project closure. So, the practical achievement of project management through a modern approach will be achieved through this project.

iii. [iii] The project will be able to meet the advanced technology in the field of healthcare with the practical implementation of deep learning neural networks that will reason and respond in similar to human beings to provide better medical service to the user.

iv. [iv] In software projects there always comes the question of how to handle user privacy and provide data security. There are many social and ethical issues we need to handle before and after the completion of the project. Regarding this topic, firstly user themselves plays an important role in helping to safeguard their information. This project will integrate Django token authentication to ensure who has access to what data, and how they can access it.

4. Initial Research

4.1.Research on Similar System

4.1.1. Florence

Florence is an artificial intelligence and machine learning chatbot that is built to assist the patient with a better understanding of their health through Facebook messenger. The important services of Florence include general health trackings such as body weight, mood fluctuations, and mensuration. The important feature of Florence is to provide friendly medication reminders to the patient (Sennaar, 2019). Firstly, the patient has to provide a time table for their medication. Florence keeps the record for timely medication and later remind the patient of their time for medication time. This feature of Florence has come to be very efficient and handy for older patients and to those who are busy with their life (Futurist , The Medicial ;, 16).

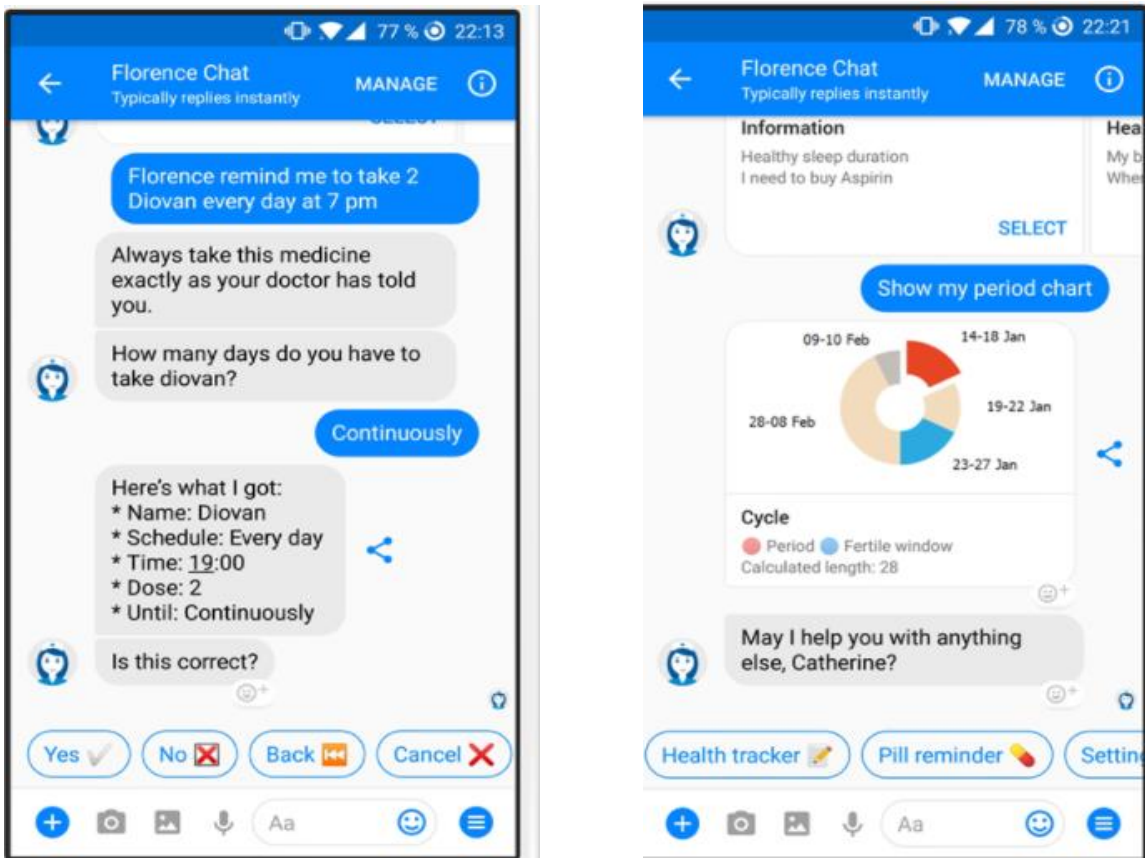


Figure 3: Florence medication reminder and health tracker *(Hawig, 2018)*.

Florence uses Natural language processing, and intent classification, and entity recognition to classify and analyze the text (S, et al., 2018).

4.1.2. Babylon Health

Babylon health chatbot is an AI online consultation and a symptom checker system founded in 2013 under the support of the United Kingdom National Health Service by a team of doctors and scientists (Marr, 2019). This chatbot provides service to the user by first asking them to choose the relatable symptoms or simply search the symptom of their illness that they are experiencing (Ayanouz, et al., 2020). From the selected symptom, the chatbot will ask a few more questions that can be related to current health conditions, daily lifestyle, and family history to the user to gather detailed information about the symptoms (Babylon;, 2020). The collected information is analyzed in the vast dataset of similar symptoms to find the possible disease or the cause of the disease (Ayanouz, et al., 2020). The patient can also take a video or audio appointment with a real doctor. During the appointment, the doctor analyses the symptoms of the patient and write the appropriate diagnoses and prescription or in critical condition refer the patient to a nearby specialist (Ayanouz, et al., 2020).

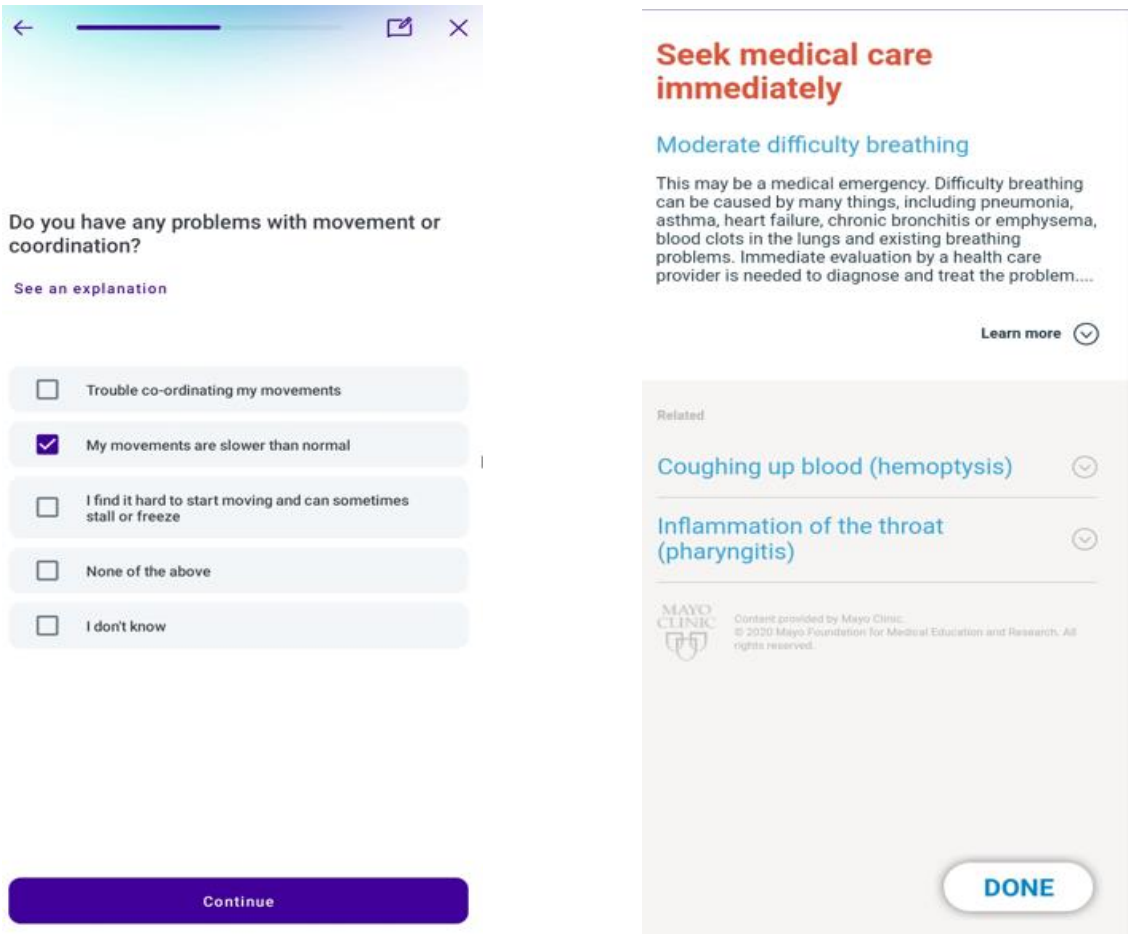


Figure 4: Babylon chatbot interaction with the user.

Babylon health uses natural language processing, and intent classification, and entity recognition to analyze the user input (S, et al., 2018). This AI chatbot also has built-in speech recognition to analyze the speech. The medical data and information of the user are securely encrypted and safely stored so that the information about the user stays secure and private (Babylon;, 2020).

4.1.3. Amazon Alexa

In 2015, Amazon developed a new conversational agent called Amazon Alexa. Alexa receives the user input from a voice-controlled interface, analyze the user queries to perform a task such as playing music, placing online orders, setting alarm and reminders in the calendar, answering general questions, and controlling network-connected devices. Alexa performs a voice-operated function when integrated with Amazon Dot, Amazon Echo, Amazon Dot, and other hardware after communicating with Amazon's AWS cloud servers through local internet connections (Lopatovska, et al., 2018).

Figure 5: Information architecture of Amazon Alexa *(Lopatovska, et al., 2018).*

The integration of deep learning functionalities makes Alexa highly engaging with the user. In Alexa, Amazon has integrated Natural Language Understanding (NLU) and deep learning automatic speech recognition (ASR) to understand the intent of text for the Alexa dialogue system. In Alexa, the Speech to text (STT) model will recognize the input speech of the user, convert it into text, and send it to the Automatic speech recognition model in the cloud server to analyze the text. The ASM model in the cloud server will analyze the text and Text to Speech (TTS) will convert the analyzed text into speech and return the result to the user (Kepuska & Bohouta, 2018).

4.1.4. Microsoft XiaoIce

Microsoft XiaoIce is currently the popular open-domain social chatbot which was launched in China in May of 2014. Microsoft XiaoIce senses the human emotional state to form an emotional connection and response to the user by the sensed state. This chatbot seemingly develops the real emotional connections with its human user (Tran, et al., 2020). XiaoIce response to sensibly to some sensitive question and then shifts the conversation to a new topic that is more comfortable to both parties. After the launch of this chatbot, it has been successful in attracting 660 million subscribed users. XiaoIce is used in different platforms such as Weibo, WeChat, and Facebook messenger. XiaoIce has become everyone's choice in their loneliness (Zhou, et al., 2020).

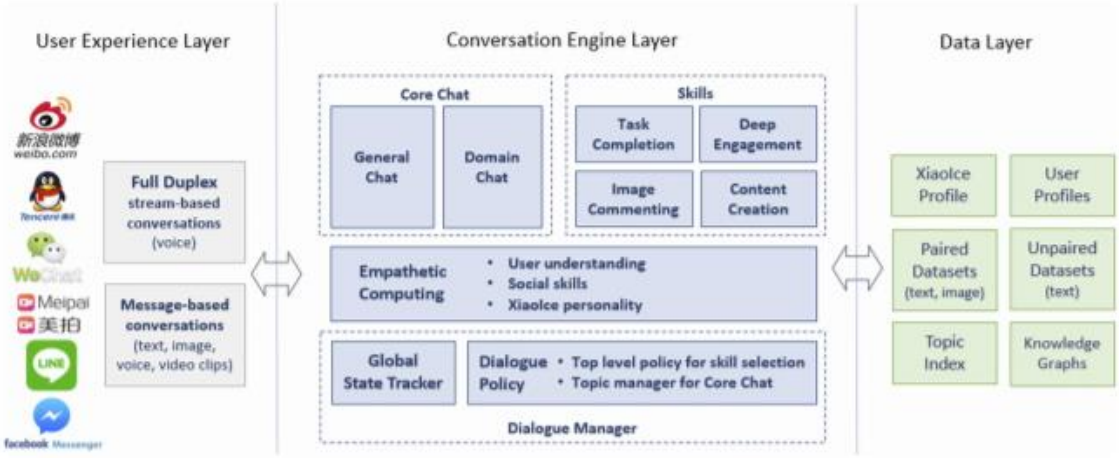


Figure 6: The system architecture of Microsoft XiaoIce *(Zhou, et al., 2020)*.

An empathetic computing framework is used to develop Microsoft XiaoIce. The empathetic framework helps the machine to recognize and understand the feelings and states of humans to respond to the user dynamically. In Microsoft XiaoIce, Intelligent quotient (IQ) is used to build up a skill to interact with the user and Emotional Quotient (EQ) to meet the sufficient user emotional needs. The overall architecture of XiaoIce consist of three layers:

i. User experience layer to connect with the user through voice and text
ii. Conversation engine layer to manage a dialogue, core chat, dialogue skills, and empathetic computing module.
iii. The data layer of XiaoIce has databases to store human conversation, non-conversation data, and knowledge graph. XiaoIce uses Contextual Query Understanding (CQU) to understand user queries.

XiaoIce image commenting skill that generates the comment for image use an image-to-text generator which is an appendage of the Microsoft Image Captioning System and boosted tree ranker. The XiaoIce poetry generation skill where it will generate expressive text from input image content uses hierarchical RNN models for words and sentences. The Conversation-turns Per Session (CPS) metric is used to measure the average time user is willing to share time with XiaoIce through conversation. The social chatbot is said to be better engaged if the CPS metric is larger (Zhou, et al., 2020). The core chat uses the neural response generator of the sequence-to-sequence framework to improve the responses of XiaoIce. The high-quality singing skill of XiaoIce is built from a high-quality parameter synthesis system with a dedicated DNN-based model. XiaoIce uses Recurrent Neural Network (RNN), an encoder-decoder model. The input from the user such as sentiment, emotion, and intent is encoded into a representation vector by Long-Short Term Memory. This representation vector is fed into a decoder to generate the response (Shum, et al., 2018). XiaoIce is exceedingly popular because of its features like entertaining visual, pleasing skills of generating poems, high quality of speaking and singing like a human (Shum, et al., 2018), image understanding and text normalization, vocal-classifier to recognize age-group and gender identity of the user (Zhou, et al., 2020).

4.2.Comparision between the techniques used in a similar system and Hosptial Management System

Chatbot	Florence	Babylon Health	Amazon Alexa	Microsoft XiaoIce	Sorabot
Description	Florence acts as a personal nurse that reminds patients to take prescriptions and keep track of the user's health (Futurist , The Medicial ;, 16).	Babylon health is a symptom checker that takes the information about the symptoms and analyze the symptoms in the vast dataset to find the possible disease or the cause of the disease (Ayanouz, et al., 2020).	Amazon Alexa is a personal assistant that is capable of playing music, placing online orders, setting alarm and reminders in the calendar, answering general questions, and controlling network-connected devices (Lopatovska, et al., 2018).	Microsoft XiaoIce senses the human emotional state to form an emotional connection and response to the user by the sensed state (Tran, et al., 2020).	Sorabot will take the user information about their symptoms and provide the best possible disease from the deep learning analyses of the symptoms.
Technology/ Approach	Deep Learning and Natural Language Processing [Page **Error! Bookmark not defined.**].	Natural Language Processing, Speech recognition [Page **Error! Bookmark not defined.**].	Natural Language Processing, Speech-To-Text (STT), Text-To-Speech (TTS) [Page **Error! Bookmark not defined.**].	Natural Language Processing, Recurrent Neural Network, Long Short Term Memory, Boosted Tree Ranker, Encoder-decoder, Automatic Speech Recognition, Emotional Computing Framework [Page **Error! Bookmark not defined.**].	Natural Language Processing and Long-Short-Term Memory.
Type of chatbot architecture	Text-To-Text (TTT) [**Error! Bookmark not defined.**].	Text-To-Text (TTT), Speech-To-Speech[Page **Error! Bookmark not defined.**]	Speech-To-Text (STT) and Text-To-Speech (TTS) [Page **Error! Bookmark not defined.**].	Text-To-Text(TTT), Image-to-Text(ITT), Speech-To-Speech(TTS) [Page	Text-to-Text

				Error! Bookmark not defined.]	
Modality	Text	Text, Speech	Speech	Text, Speech, and Image	Text
Architectural Model	Retrieval Chatbot	Retrieval Chatbot	Generative Chatbot	Generative chatbot	Retrieval Chatbot
Programming Language.			Python, Java, node JS (Gambhir, 2019).	Microsoft framework	Python, Django framework, Keras with Tensorflow, and SqlLite database.
Intent classification and entity extraction	Yes	Yes	Yes	Yes	Yes
Self-Training	Yes	No	Yes	Yes	Yes
User authentication	Yes	Yes	Yes	Yes	Yes
Web application	Yes: It is integrated with Facebook, Skype.	Yes	Yes	Yes: It is integrated on social sites like Facebook, WeChat.	Yes
Phone application	No: Use Facebook, the Skype messenger	Yes: Babylon app	Yes: Amazon Alexa app	Yes	No
Upload medical report to the sytem	No	No	No	No	Yes

Appointment booking through the chatbot	Yes: The user can book an appointment through the chatbot.	No	Yes: The user can book an appointment through Amazon Alexa.	No	No: The chatbot does not allow the user to book an appointment
Appointment booking through the system	No: The system does not have a feature of booking an appointment.	Yes: The system has the feature of booking an appointment.	No: Amazon Echo as a system does not have the feature of booking an appointment.	No: Microsoft XiaoIce does not have the feature of booking an appointment.	Yes: The system has the feature to book an appointment.

4.3.Analysis of Similar System

The above literature review is conducted on two healthcare chatbots, one open-domain social chatbot, and one virtual assistant. Now, it is easy to point out that a virtual assistant is not a chatbot because its interface and functionality are different from the chatbot. However, the above literature review shows that virtual assistants also integrate the similar mechanism of the chatbot to reason and response to the user. Some virtual assistants like Amazon Alex convert the input speech to text with intent identification and entity recognition before analyzing the input [**Error! Reference source not found.**]. In recent years, some studies have shown that the virtual assistant is more intelligent than chatbot because the virtual assistant is integrated with deep learning due to which they have better understanding and reasoning than chatbots. However, the advancement in the field of deep learning has touched the floor of chatbot where the intelligent open-domain chatbot like Microsoft XiaoIce attracting millions of users has been developed. The major takeaway from the above analysis is to develop the chatbot with deep learning like Long-Short Term Memory to interact with the user and learn their queries to provide them with meaningful responses as a human does.
Thus, bringing out more advancement in the chatbot.

From the review of similar chatbots in healthcare, it is known that there are different services provided by the chatbot. This service includes symptom checking, appointment booking, a reminder to take medication and follow-up treatments, and answering frequently asked questions. From overall findings, I have come up with the idea of implementing the symptom checker features of chatbot in my project. Even though technology has been successful to gain human trust through its enormous contribution to every sector of society, healthcare is one of the major concerns in the community. So, through many research and analysis, I concluded that the very first step to proper health care is finding out the symptom. Even though all the mentioned features are significant in different health sectors but I have come up with a conclusion to implement the first step of providing quality healthcare services to people.

This idea of implementing a symptom checker in a chatbot is introduced from Babylon healthcare chatbot. This application is a great option for the user for knowing their health. This symptom checker use intent identification and entity recognition to understand user goal. Through the help of natural language processing, this chatbot analyzes the text and responds to the user. However, the only drawback that this application has is providing a fixed set of answer options to the user as shown in [**Error! Reference source not found.**]. The user has no other choice other than to choose from the given option. This application can be bothersome use for the user if they are not suffering from any of the symptoms from the option. In this case, the chatbot cannot provide an accurate result. The major takeaway from this application for this project is to build a chatbot that will allow the user to keep their questions and answers rather than just choosing a question from provided options. This project focus on building deep learning chatbot which will respond to the user questions rather than giving own question. The chatbot in this project will use natural language processing with intent identification and entity recognition along with deep learning which will make it better than Babylon Chatbot.

4.4. Comparative study on the performance of deep learning algorithms in chatbot

The domain-specific chatbots with deep learning are one of the most emerging research areas. The research in domain-specific chatbot was conducted by (Pathak & Arya, 2019)in Facebook's bAbI dataset to test the accuracy of chatbot The learning architecture of various recurrent neural networks such as Bi-directional LSTM, Single LSTM, and Gated Recurrent Unit (GRU) with Glove vector to generate the vector representation of words was used in dynamic memory networks (Pathak & Arya, 2019).

The experiment was conducted between two-layer Single Forward LSTM, three-layer Bi-directional LSTM, and two-layer Gated Recurrent Unit in 20 different types of tasks from Facebook's dataset. The accuracy of the model was compared between the questions asked to the system and correctly answered questions by the system. According to the result from this experiment, the single forward LSTM had an average accuracy of 0.75, Bi-directional LSTM had an average accuracy of 0.74 and GRU had an accuracy of 0.72. The performance of GRU exponentially worse in comparison to LSTM in the task that required remembering the long sequence (Pathak & Arya, 2019). The figure below shows the accuracy result of single LSTM, Bi-directional LSTM, and GRU.

	LSTM				Bi-Directional LSTM				GRU			
	Layers	Batch Size	Epochs	Drop Out	Layers	Batch Size	Epochs	Drop Out	Layers	Batch Size	Epochs	Drop Out
	2	32	100	0.3	3	32	150	0.5	2	32	100	0
	Training Accuracy		Validation Accuracy		Training Accuracy		Validation Accuracy		Training Accuracy		Validation Accuracy	
QA_1 Single supporting fact	0.99		0.961		0.95		0.961		0.98		0.867	
QA_2 Two supporting facts	0.699		0.68		0.959		0.46		0.67		0.3029	
QA_3 Three supporting facts	0.9406		0.3512		0.926		0.306		0.921		0.27	
QA_4 Two argument relations	0.9958		0.99		0.971		0.99		0.995		0.98	
QA_5 three argument relations	0.9869		0.861		0.9842		0.861		0.9862		0.821	
QA_6 yes/ no questions	0.9454		0.8652		0.915		0.8862		0.9322		0.841	
QA_7 counting	0.9307		0.7821		0.937		0.769		0.9124		0.759	
QA_8 List Set test	0.9055		0.745		0.95		0.761		0.897		0.745	
QA_9 Simple Negation	0.9803		0.84		0.93		0.85		0.9212		0.8412	
QA_10 Indefinite knowledge	0.9893		0.92		0.9793		0.914		0.9642		0.9244	
QA_11 Basic coreference	0.9964		0.983		0.984		0.983		0.9886		0.99334	
QA_12 conjunction	0.9982		0.995		0.9984		0.995		0.974		0.9952	
QA_13 compound coreference	0.9986		0.995		0.99		0.9822		0.987		0.99532	
QA_14 time reasoning	0.8222		0.4521		0.802		0.374		0.852		0.3742	
QA_15 Basic deduction	0.8428		0.514		0.8028		0.5521		0.814		0.5242	
QA_16 Basic Induction	0.6543		0.482		0.6043		0.4714		0.684		0.462	
QA_17 positional-reasoning	0.9181		0.807		0.9481		0.7922		0.925		0.817	
QA_18 size-reasoning	0.9988		0.92		0.9977		0.9		0.928		0.92	
QA_19 path-finding	0.3394		0.1596		0.394		0.135		0.3145		0.115	
QA_20 Agents motivations	0.9957		0.979		0.942		0.95		0.9457		0.979	
			0.76411				**0.744655**				**0.726338**	

Figure 7: Accuracy result of Single LSTM, Bi-directional LSTM, and GRU *(Pathak & Arya, 2019).*

These experiments showed the single LSTM with almost 95% of overall accuracy than bi-directional LSTM and GRU. The overall accuracy of layers of bi-directional LSTM is similar to the single LSTM because the three layers of bi-directional LSTM need more parameters for training. However, this experiment concludes with the result of a significant improvement in the accuracy of the LSTM model than the GRU model (Pathak & Arya, 2019).

4.5.Comparative study and analyses for selection of algorithm

For selecting the algorithm for this project, I have conducted research and findings from various research sources and experiments on three algorithms namely Multinomial Naïve Bayes, Recurrent Neural Network, and Long-Short Term Memory. The findings below refer to the strength and shortcomings of each algorithm.

ALGORITHM	Multinomial Naïve Bayes	Recurrent Neural Networks	Long-Short Term Memory
STRENGTH	Multinomial Naïve Bayes is an extremely fast algorithm for both training and making predictions. It is wise to choose this algorithm for an initial baseline classification when the complexity of the model is less important and the model is fed with well-separated categorical datasets (VanderPlas, Jake;, 2016). Multinomial Naïve Bayes classify the text into certain categories for the chatbot to identity the intent in text. Intent identification is the major step in chatbot conversations so this algorithm works better (V., et al., 2020).	Recurrent neural networks propagate historical information through a chain of neural networks. The recurrent neural network is useful to processes the sequential data from the current input as well as previous output at each time step.	Long-Short Term Memory is a variant of Recurrent Neural Network that was introduced to perform better on problems with long term dependencies. The LSTM allows memorizing information for an extended number of timesteps. LSTM is widely used when it is important to learn from previous experiences that have very low time lags (Zaremba, et al., 2015). LSTM performs better with long-term dependencies and helps to keep the gradients step high thus providing high accuracy in model training (Donges, 2020). LSTM has explicit memory cells for storing information for a long period (Zaremba, et al., 2015). LSTM memory can be referred to as a gated cell which will decide to delete the old information, update the old information, retrieve it or keep it for the next time step based on their importance. LSTM learns to assign information based on its importance over time (Donges, 2020).

SHORTCOMING	However, this algorithm becomes problematic with more complicated models (VanderPlas, Jake;, 2016). This algorithm relies on the frequency of the appearance of the word in a particular category. However, this algorithm considers the word as an entire set and selects the word that has the highest probability to determine the class of input ignoring the rest of the word in which they appear. This can turn out to be problematic causing the input and their output to differ (V., et al., 2020).	However, being one of the most promising algorithms in deep learning, RNN suffers a major setback when the gradient vector grows or decay exponentially over long sequences during the training phase. This problem with RNN is termed as exploding or vanishing gradient. Due to this problem, RNN cannot learn long-distance correlations in a sequence (Liu, et al., 2016).	The data in LSTM moves from one cell to another cell for evaluation due to which the cell becomes complex with additional features. This complex module requires a lot of data and time to train. The LSTM model is prone to model overfitting and is difficult to apply an optimization method to overcome this overfitting (Sinha, 2020).

Therefore taking into consideration the functionality of chatbot and overall performance, complexity, accuracy, and flexibility of algorithm I have to consider using deep learning neural networks called Long-Short Term Memory. This algorithm has its shortcomings of model fitting and high requirements in training. However, after being through these shortcomings this algorithm gives the most satisfying result in deep learning. It is also important to note that it is not always the algorithm that provides the highest accuracy and better performance but the major piece of effort also depends on the dataset that is used to fed train and test the model.

5. Artifact

5.1. Description of Artefact

The hospital management system will be the first and main artifact of this project.

The different subsystem that of hospital management system are [Figure 2]:

5.1.1. User Management

i. Guest user is allowed to register
ii. The registered user is allowed to login and log out of the system
iii. After logging into the system, the user can enjoy various services of a system that includes interacting with the chatbot to receive emergency first aid and other health-related services.

5.1.2. Doctor Management

i. The admin will add a doctor to the system.
ii. The admin will update the doctor profile if necessary in the system.
iii. The admin will remove the doctor from the system.

5.1.3. Appointment Management

i. The user can book an appointment with appropriate physicians available in the system.
ii. The user can also cancel the appointment.
iii. The user can update the appointment date and time.

5.1.4. Transaction Management

Before booking an appointment, the user has to make a transaction to make sure that the booking becomes valid. Transaction management will manage all the transaction with user information.

5.1.5. Report Management

The user is allowed to upload the medical report in pdf format in the system.

5.1.6. Chatbot

The chatbot in the hospital management system will provide health-related service to the user. To receive the service from the chatbot, the user must be a registered user of the hospital management system. The main features that will be integrated into the chatbot are:

i. It will interact with chatbots through text. The chatbot will take inquiries about the symptoms from users and provide help them to discover their disease.
ii. The chatbot will also provide answers to emergency first aid service to the user.

5.1.7. Database

The hospital management system will integrate the SQLite database to store the data of all the information that is related to the hospital. The database will store:

i. The information of the registered user
ii. The information about medical personnel

iii. The information about the time of appointment and user who has booked the appointment.

iv.

5.2.Justification of Artefact

The hospital management system will be an advanced system with the integration of generative chatbot. This web-based system will be built using the Django framework. Django is based on the Model-View-Template pattern. The generative chatbot will be integrated with a deep learning model from where it will predict the symptom of the user. Users will interact with the system through Django RESTful API. The chatbot model will be deployed in the server through the Django API. The user will send the inquires about their symptom to the chatbot and the API will handle these requests and fetch the predication from the LSTM model and through the View where the logic behind the whole system is implemented, the API will respond to the user. Between API and view, there will be a URL that will handle the request to and from the appropriate view. The user information, appointment booking, and transaction information will be stored in the database. The Model will fetch the data from the database and send it to view from where it is displayed in the browser through the template.

5.3. Full Details of Artefact

5.3.1. Development Cycle

Scrum is the most widely used agile software development framework. Scrum provides iterative and incremental development of software. Scrum adds flexibility, adaptability, and productivity in the development process to release the product according to customer requirements, time pressure, competition, product quality, and available resources (Anwer, et al., 2017).

The development cycle of the scrum for the project is explained below:

i. Sprint Planning

Sprint is a time box period during which a specific list of work in the sprint backlog must be completed (Anwer, et al., 2017).

In this phase, the product owner creates the product backlog. In my project I will create the product backlog with a clear selection of tasks with the highest priority and scrum master will my supervisor to track the progress of my project. Since this project is individual, the development team will be replaced by myself (Anwer, et al., 2017).

a. Product Backlog

The product backlog is the list of all the requirements that have to implement during the development process. For my project, the requirement in product backlog will be features of chatbot and other activities that include the design of user interface, database design, hospital management system, documentation, testing, etc (Anwer, et al., 2017).

b. Sprint Backlog

Sprint backlog contains a specific list of requirements from the product backlog which needs to be complete in each sprint (Anwer, et al., 2017).

ii. Initiation and Development Phase

In this phase, analyses and design to meet the requirement of the features are developed, and features are implemented into code so that it is functional. Lastly, the feature is documented through the various phase of testing. This phase mainly focuses on how to build a product until the end of the sprint (Anwer, et al., 2017).

iii. Monitoring and Control

a. Sprint Report and Review

Sprint reports will contain any issues that arise during the sprint. Sprint review meetings will be held with the supervisor to present and track the current project status (Anwer, et al., 2017).

iv. Closure

This phase marks the closing of the project. The product is presented to the supervisor and other academic readers. (Anwer, et al., 2017).

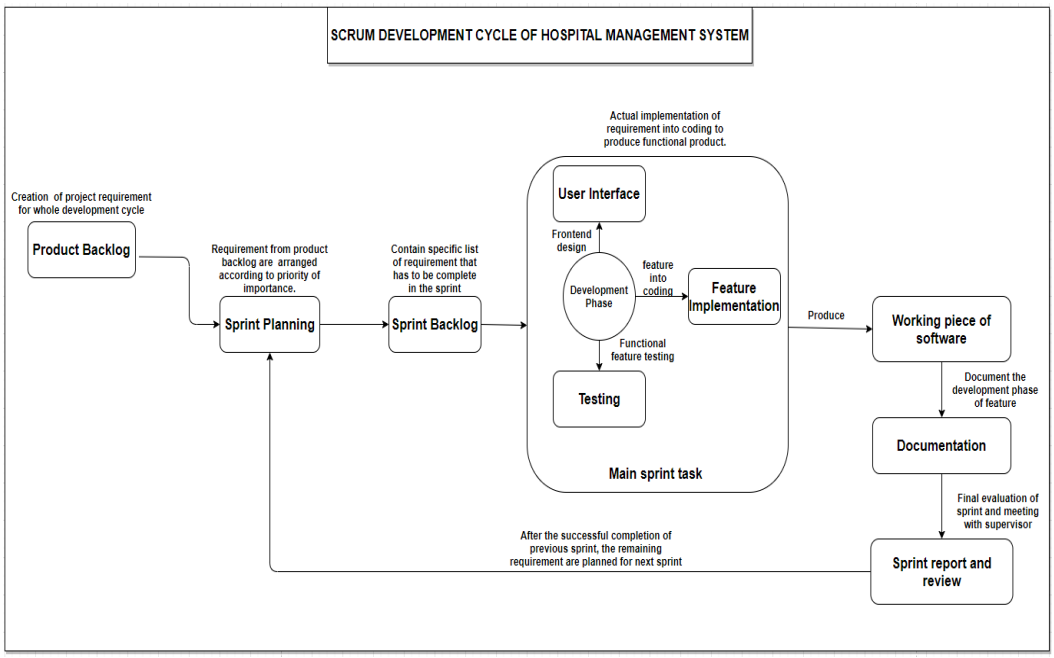


Figure 8: Scrum development cycle of Hospital Management System.

5.3.2. Tools

i. Planning and scheduling tool

Microsoft Excel will be used to produce a work breakdown structure and a Gantt chart to breakdown and schedule the project task.

ii. Modeling tool

Draw.io is a free online tool for UML diagrams to visually represent a system. For coursework, I will use UML models like the data model for database design, a process model for showing the flow of activities in the system, and Adobe Creative Cloud to design a clear user interface.

iii. Programming Language

The healthcare chatbot system is a web-based application. So, I will use python programming. Python programming has a vast source of built-in libraries and modules for machine learning algorithms that are easy to implement and flexible.

iv. Framework

For web-based applications, I will use the Django framework which is one of the most used web-based python frameworks. Django provides many built-in features for python programming. For deep learning model build-up, I will use the Keras library with TensorFlow. Keras provides a high-level Application Programming Interface with an easy and flexible user interface. Keras runs on top of TensorFlow which provides efficient numerical computation for deep learning model development.

v. Software and Hardware

The software used for programming the system will be Visual studio. Visual Studio is an application development tools that support all programming language. The hardware that will be used for this project is a laptop.

vi. Application Programming Interface

This project will be built using the Django framework. Therefore for the deployment of the model and user interaction with the system Django Application User Interface will be used [Figure 1].

vii. Database

To store the information of all the user data and hospital data, the system will use an SQLite database which is a structured database. The data are kept in a structured format which will be easy and convenient to retrieve and manipulate data. Since the SQLite database comes as a builtin database in Django it is also easy to integrate into the system.

viii. Project Management tool

Since I will be working with the scrum methodology, I prefer to use the Jira tool to plan, track, and manage the overall progress of the project throughout the development.

5.3.3. Techniques

For coursework, I prefer to use articles, reports, case studies, journals, conference papers, news, the document from the website, and website for research.

5.4.Consideration of Methodology

5.4.1. Extreme Programming

Extreme programming is a lightweight and flexible approach to software development with the ability to manage the rapidly changing requirements. XP is a collection of values, principles, and practices to develop high quality of software. It carries out frequent feedback and releases to manage customer requirements and improve customer satisfaction. (Anwer, et al., 2017).

5.4.2. Waterfall

The waterfall model is a very simple and easy to understood software development approach. It is a linear-sequential phase where each phase starts with the completion of the previous phase. This methodology works well in small projects where the requirements are fixed and well understood from the beginning. (ACADEMIA;, 2020).

5.4.3. Scrum

Scrum is the most widely used agile software development model. Scrum provides iterative and incremental development of software. Scrum adds flexibility, adaptability, and productivity in the development process to release the product according to customer requirements, time pressure, competition, product quality, and available resources (Anwer, et al., 2017)

5.5.Justification of the Methodology

The reason why I choose to implement Scrum in this project is that it is a familiar approach that I have implement in the Software management project. I have a clear vision of working with Scrum. Scrum deals with project management related issues. Scrum divides works into sprint due to which it is very helpful to understand the flow of the project and carry the clear vision of the project through the development phases. Scrum allows the change in requirements during the development phase. This becomes easy to plan the sprint and complete the project as per change in requirement with time [Page 16]. (Anwer, et al., 2017).

5.6.Testing

There are different types of software testing approaches. For this project, we will mainly focus on the following types of software testing:

i. Black Box Testing

Black box testing is based on the identification and analysis of the performance of the software system without being known to its internal working mechanism. Black box testing only takes into account that the input is properly fed into the system and the correct output is provided by the system keeping aside the internal complex computation to produce the output (Khan, 2010).

ii. White Box Testing

White box testing is based on identification and analysis of the performance of the software system knowing internal implementation details and architecture of the system. These types of testing are widely used to detect bugs and errors and resolve problems caused by those bugs in the system (Sneha & M, 2017).

6. Project Planning

6.1.Work Breakdown Structure

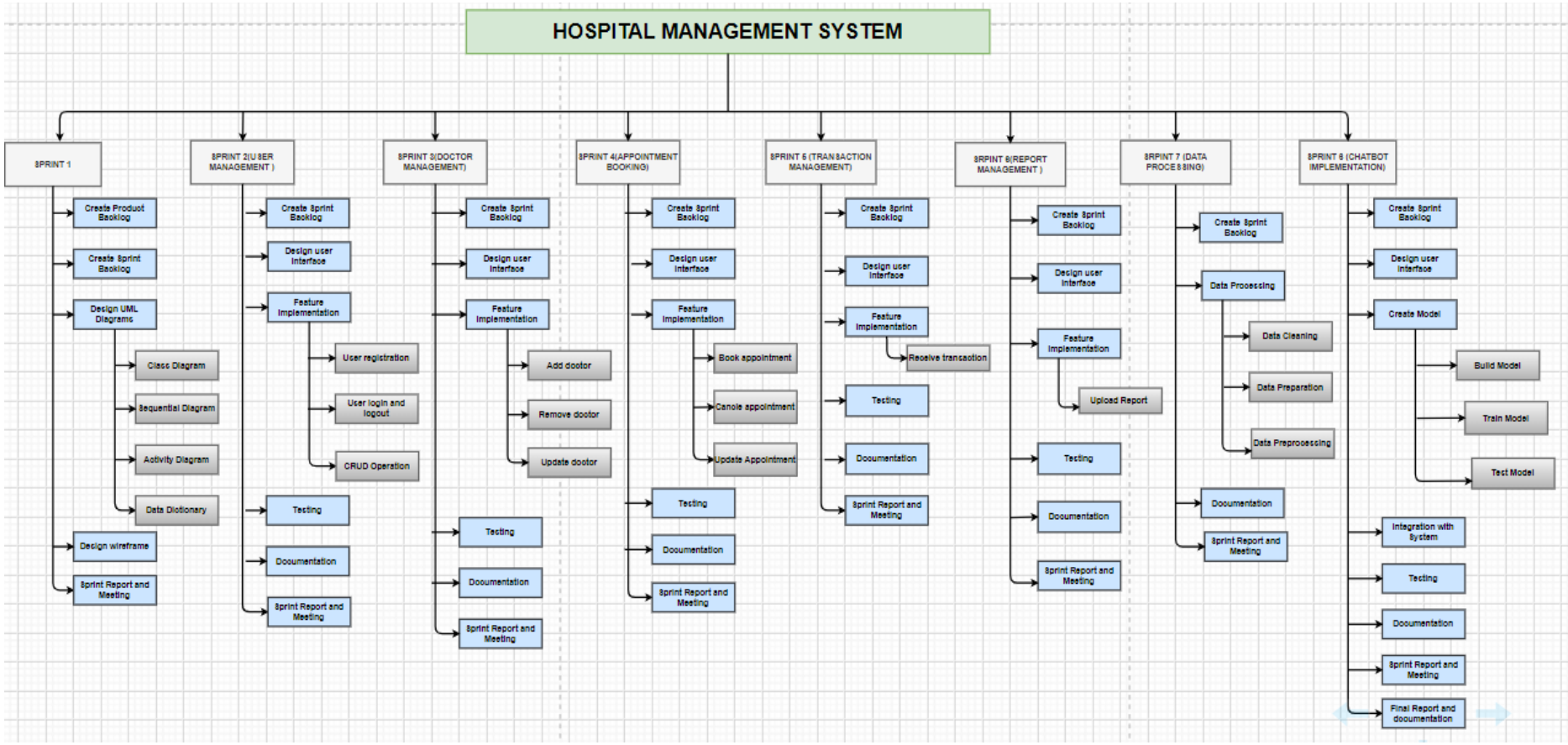


Figure 9: Work Breakdown Structure for Hospital Management System

6.2.Gantt chart

ACTIVITIES	START DATE	END DATE	DURATION
SPRINT 1	24-Sep	30-Sep	7
Create Product Backlog	24-Sep	25-Sep	2
Create Sprint Backlog	24-Sep	24-Sep	1
Design UML Diagrams	25-Sep	28-Sep	4
Design wireframe	29-Sep	29-Sep	1
Sprint Report and meeting	30-Sep	30-Sep	1
SPRINT 3	01-Oct	11-Nov	42
Create Sprint Backlog	01-Oct	01-Oct	1
Data preprocessing	02-Oct	08-Nov	38
Documentation	08-Nov	10-Nov	3
Sprint Report and meeting	11-Nov	11-Nov	1
SPRINT 4 (User Management)	12-Nov	02-Dec	21
Create Sprint Backlog	12-Nov	12-Nov	1
Design user interface	13-Nov	15-Nov	3
User authentication	16-Nov	28-Nov	13
Testing	28-Nov	30-Nov	3
Documentation	28-Nov	01-Dec	4
Sprint Report and meeting	01-Dec	02-Dec	2
SPRINT 5 (APPOINTMENT BOOKING)	03-Dec	24-Dec	22
Create Sprint Backlog	03-Dec	03-Dec	1
Design user interface	04-Dec	06-Dec	3
Feature Implementation of transaction	06-Dec	18-Dec	13
System Testing	19-Dec	20-Dec	2
Documentation	19-Dec	21-Dec	3
Sprint Report and meeting	21-Dec	22-Dec	2
SPRINT 6(TRANSACTION MANAGEMENT)	23-Dec	13-Jan	22
Create Sprint Backlog	23-Dec	23-Dec	1
Design user interface	23-Dec	25-Dec	3
Feature implementation of transaction	25-Dec	08-Jan	15
Testing	09-Jan	11-Jan	3
Documentation	11-Jan	12-Jan	2
Sprint Report and meeting	13-Jan	13-Jan	1
SPRINT 7(CHATBOT IMPLEMENTATION)	14-Jan	30-Apr	107
Create Sprint Backlog	14-Jan	14-Jan	1
Design user interface	14-Jan	16-Jan	3
Build Model	16-Jan	10-Feb	26
Training Model	11-Feb	28-Feb	18
Testing Model	01-Mar	25-Mar	25
Integration with System	26-Mar	14-Apr	20
Testing	15-Apr	20-Apr	6
Documentation	18-Apr	28-Apr	11
Sprint Report and meeting	29-Apr	30-Apr	2
Final Report and documentation	01-May	06-May	6
Final Presentation	07-May	07-May	1

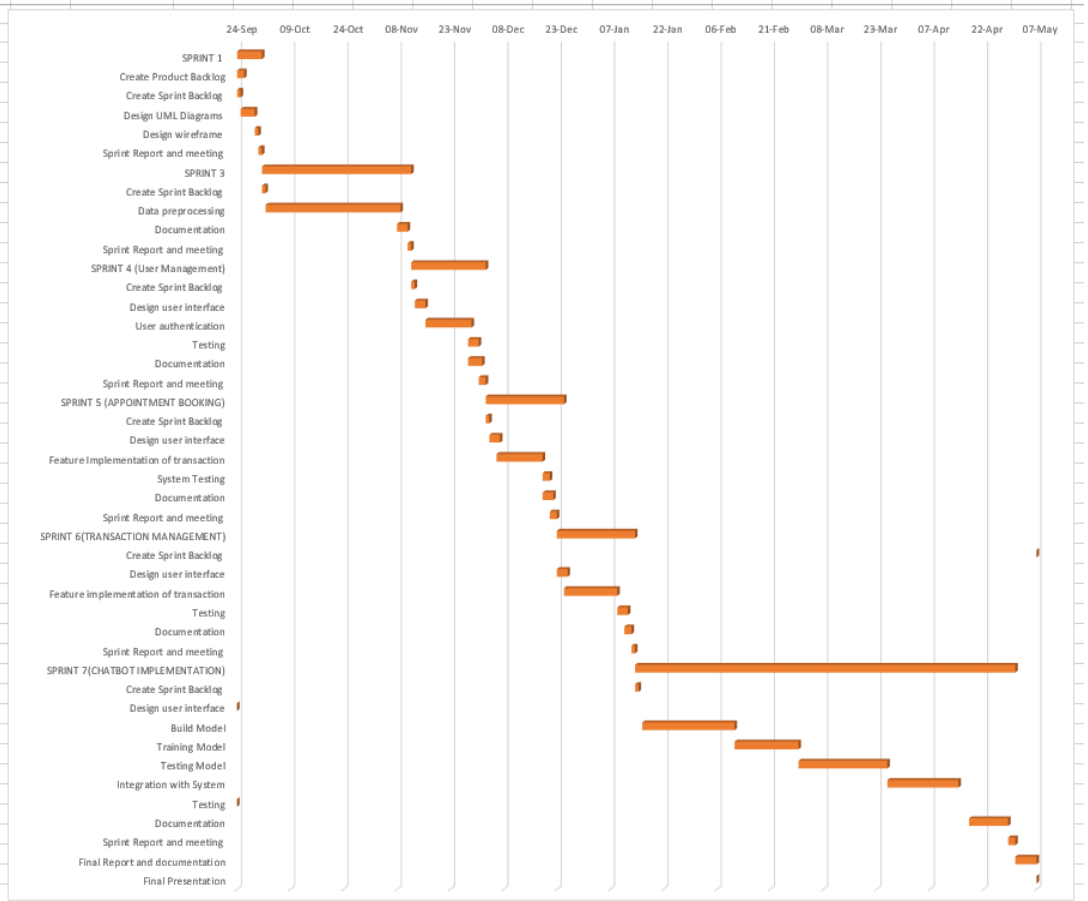


Figure 10: Gantt Chart of Hospital Management System.

SECTION 4: SOFTWARE REQUIREMENT SPECIFICATION

1. Software Requirement Specification

Software Requirement Specification is an important document in software projects that contains the specific set of software requirements of the software system and their descriptions. It is essential to write SRS before the project because it specifics the needs and demands of clients as well as the system design. Software requirement specification makes sure that the client understands the aim of software development (Sabriye & Zainon, 2017).

There are three types of Software Requirement Specifications:

1.1. Functional Requirement

A functional requirement is the main requirement description of desired future product behavior (Lutters, 2014). It is a fundamental action that the system must do or must not do. In action, it can be described as accepting the input and generating the output in the software (Susilowati, et al., 2019).

1.2. Non-Functional Requirement

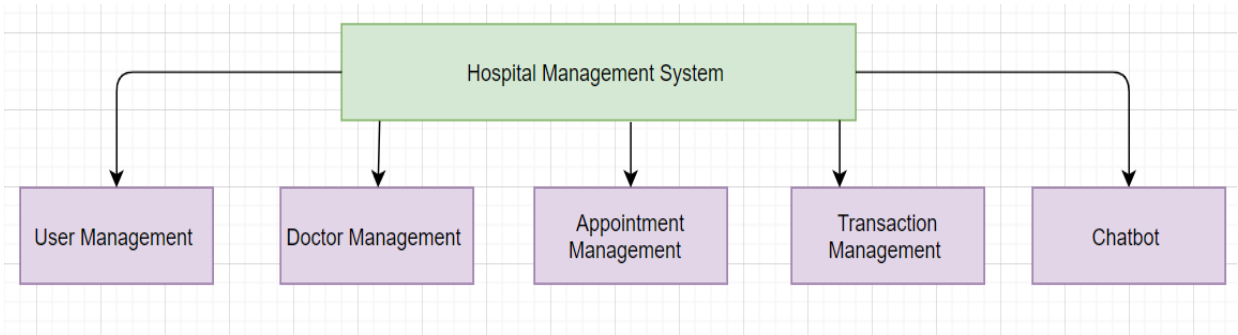
A non-functional requirement is the broad set of qualities that a software system should have such as reliability, performance, security, maintainability, usability, and scalability. Non-functional requirement plays an important role in user experience and should be identified and considered during requirement specification (Lu & Liang, 2017).

1.3. Usability

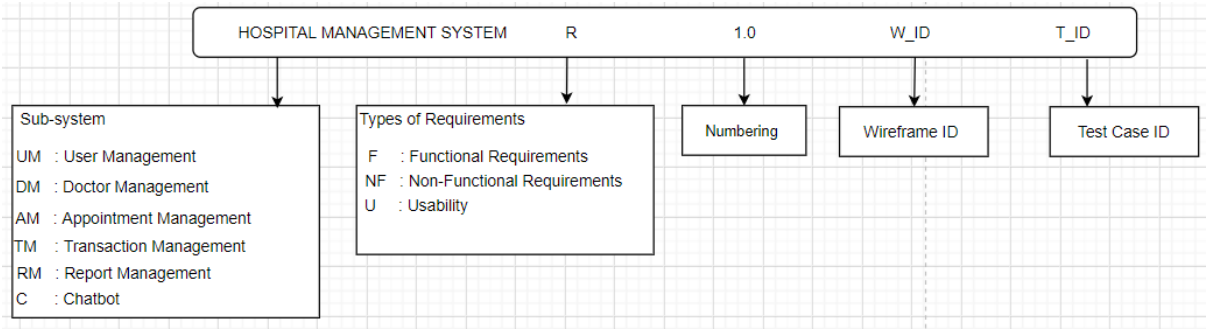
Software usability is an essential non-functional requirement of a software project that defines the quality of software in terms of user experience while interacting with the system. Usability is about effectiveness, efficiency, and overall user satisfaction (Abdelaziz, et al., 2016).

1.4. Requirement Specification

FUNCTIONAL DECOMPOSITION DIAGRAM



LEGEND



The software requirement specification for Hospital Management System is prepared below:

1.4.1. User Management

1.4.2. Doctor Management

S_ID	Description	Moscow	W_ID	T_ID
DM-F-1.0	The admin must be able to add medical personnel's in the system.	Must have		
DM-F-1.1	The admin must be able to update the doctor's information in the system.	Must have		
DM-F-1.2	The admin must be able to remove the doctor from the system.	Must have		
DM-U-1.3	The information of doctor with their qualification and specialization must be provided in the system to make the user easy to find their doctors.	Must have		
DM—NF-1.3	The qualification and specialization of doctor of at most 500 characters.	Must have		
DM-NF-1.4	The list of doctors must be displayed in the admin dashboard.	Must have		

1.4.3. Appointment Booking

S_ID	Description	Moscow	W_ID	T_ID

AM-F-1.0	The user must be able to book an appointment with the appropriate doctor.	Must have		
AM-F-1.1	The user must be able to update an appointment date and time if necessary.	Must have		
AM-F-1.2	The user must be able to cancel an appointment.	Must have		
AM-NF-1.3	The user must enter a valid date and time to book an appointment.	Must have		
AM-NF-1.4	The system should display the list of time table for doctor available.	Should have		
AM-U-1.5	The system could display an error message if the user enters an invalid date or time or both.	Could have		

S_ID	Description	Moscow	W_ID	T_ID
UM-F-1.0	The user must be able to register in the system to access the system.	Must have		
UM-F-1.1	The user and admin must have their separate dashboard.	Must have		
UM-F-1.1	The user must be able to login to the system.	Must have		
UM-F-1.2	The user must be able to logout into the system.	Must have		
UM-F-1.4	The admin must be able to remove the user.	Must have		
UM-F-1.5	The user must be able to add their balance to the system if necessary.	Must have		
UM-NF-1.6	The user should be able to reset their password in case of a forgotten password.	Should have		
UM-NF-1.7	The system password must be a combination of special characters.	Must have		
UM-NF-1.8	The user must not be able to access the admin dashboard.	Must have		
UM-U-1.8	The system could provide a remember me option while logging into the system.	Could have		
UM-U-1.9	The system should display an error message in case of invalid input into the form field.	Should have		
UM-U-1.10	The system should allow the user to search their doctor.	Should have		

AM-F-1.6	The individual user should book only one appointment for a specific time.	Should have		
AM-NF-1.7	If a user tries to book two appointments for the same time system can display a message.	Could have		

1.4.4. Transaction Management

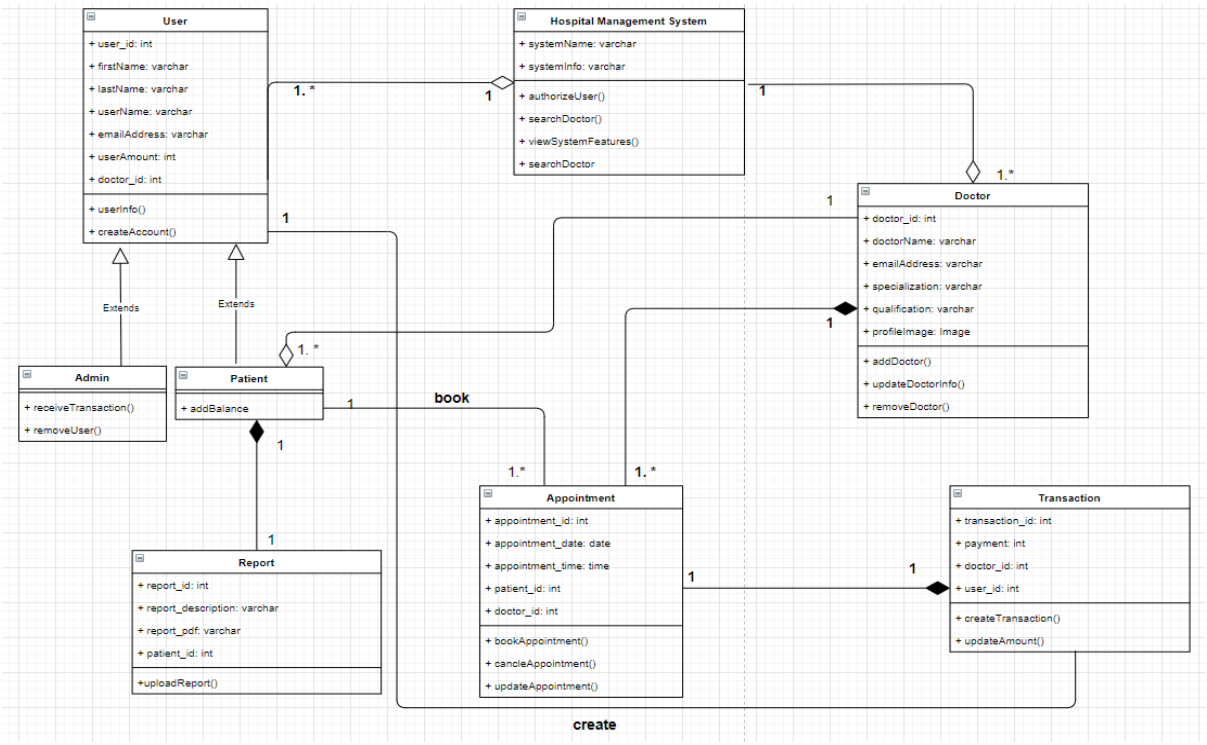
S_ID	Description	Moscow	W_ID	T_ID
TM-F-1.0	The user must be able to make a transaction in the system before confirming the appointment.	Must have		
TM-F-1.1	The user must be allowed to update their balance if they do not have sufficient balance to make a transaction	Must have		
TM-F-1.2	The balance of the user must be deducted.	Must have		
TM-NF-1.3	The system must not allow the user to make a transaction if the balance is less than the amount for the appointment.	Must have		
TM-NF-1.4	The system must store the transaction amount with detailed information about the user.	Must have		

1.4.5. Report Management

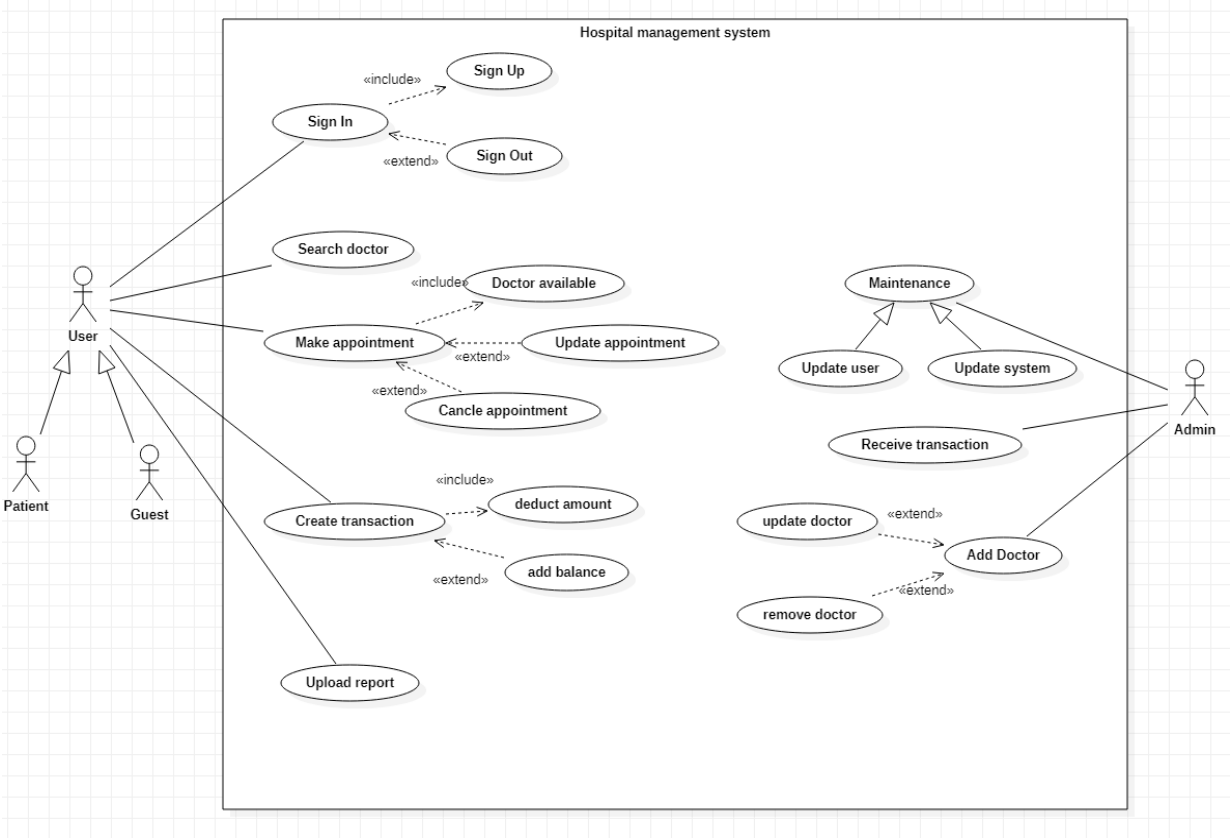
2. S_ID	Description	Moscow	W_ID	T_ID
RM-F-1.0	The user must be able to upload their medical report in the system.	Must have		
RM-NF-1.1	The system must allow only pdf format of the report.	Must have		
RM-U-1.2	The system must display an error message for an invalid report format.	Must have		
RM-NF-1.3	The report can be images, documents in pdf format.	Must have		

SECTION 5: SYSTEM DIAGRAMS

6.1.Class Diagram

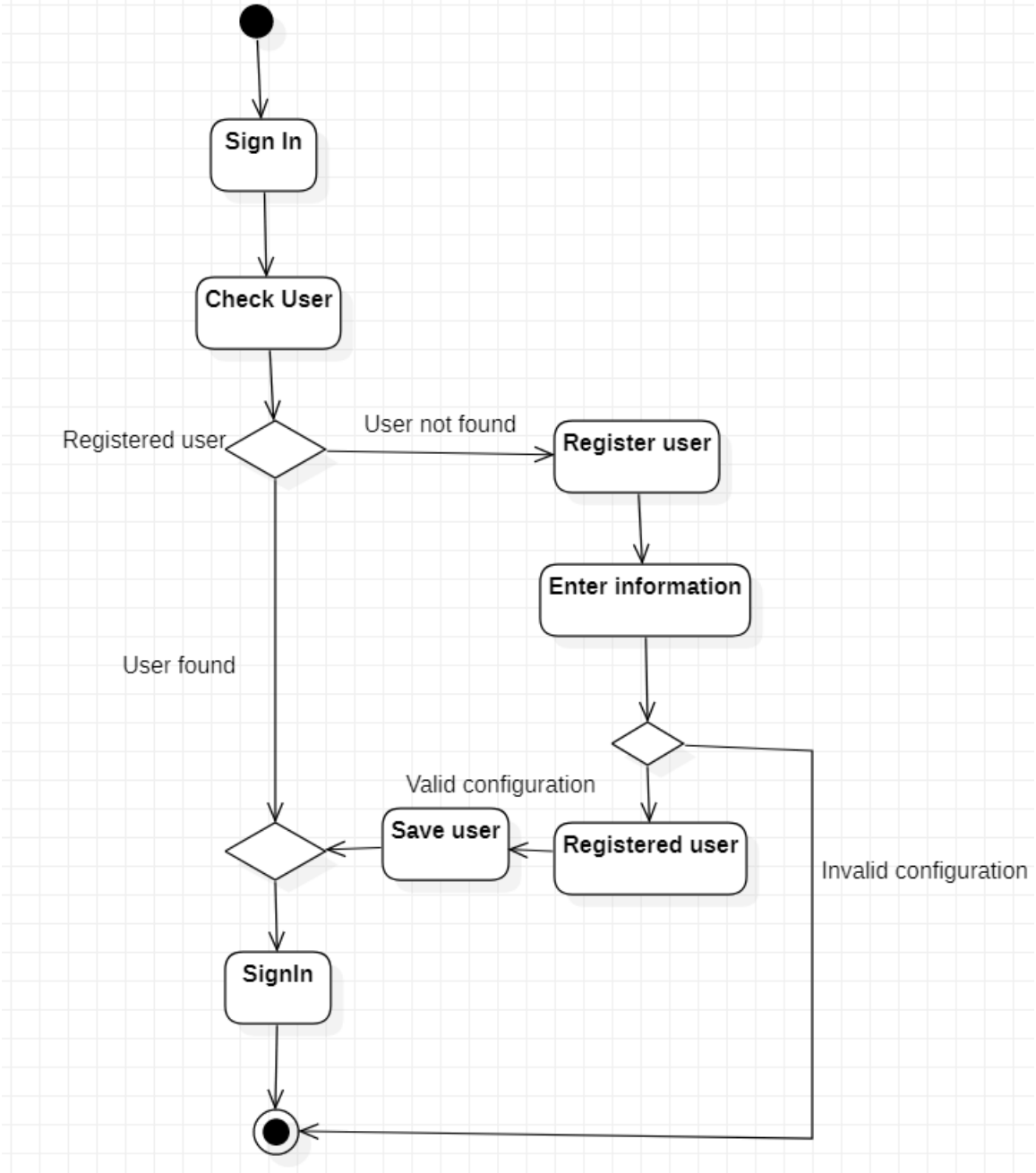


6.2.Usecase Diagram

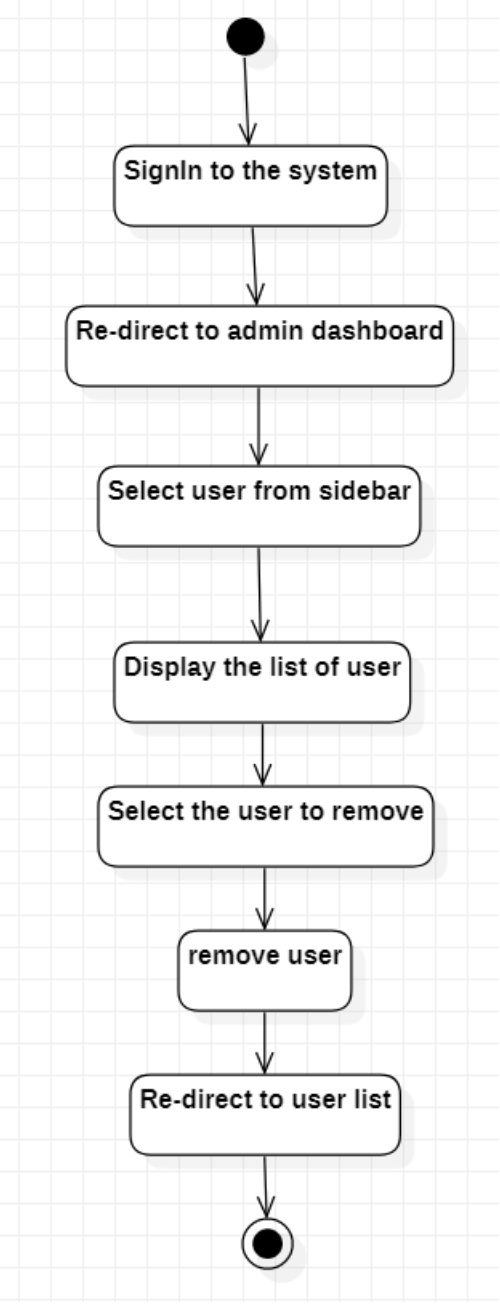


6.3.Activity Diagram

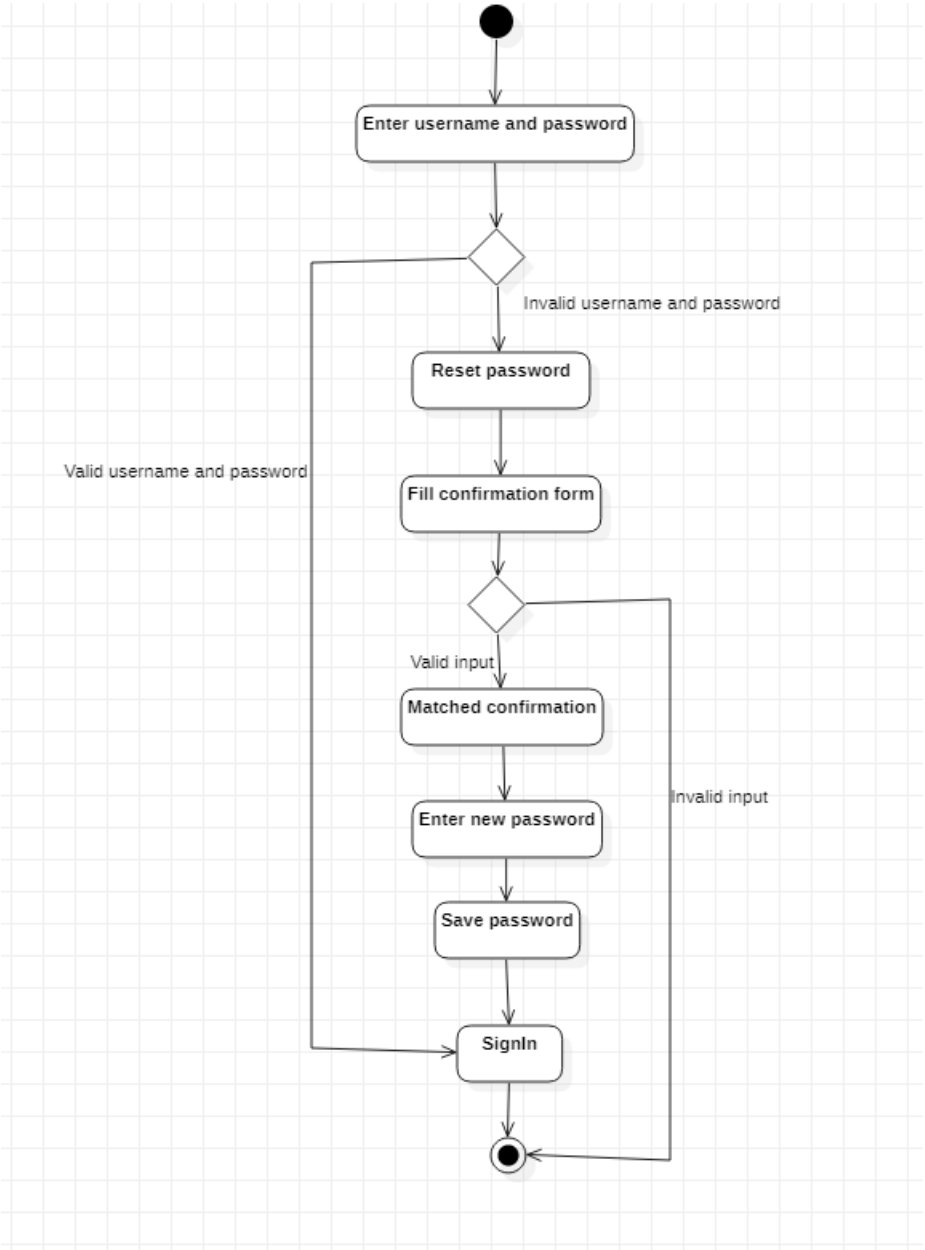
6.3.1. User Registeration



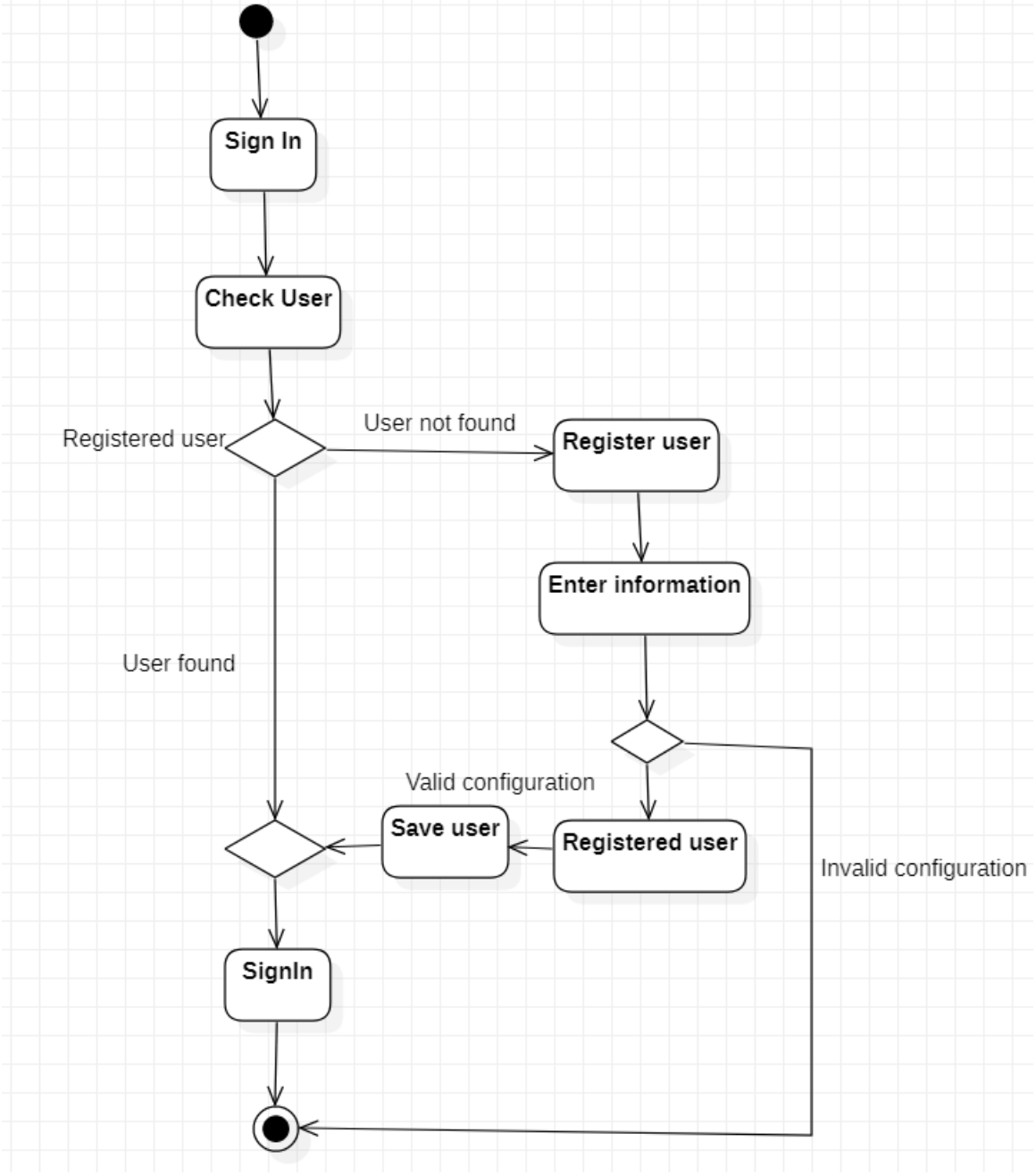
6.3.2. Remove User



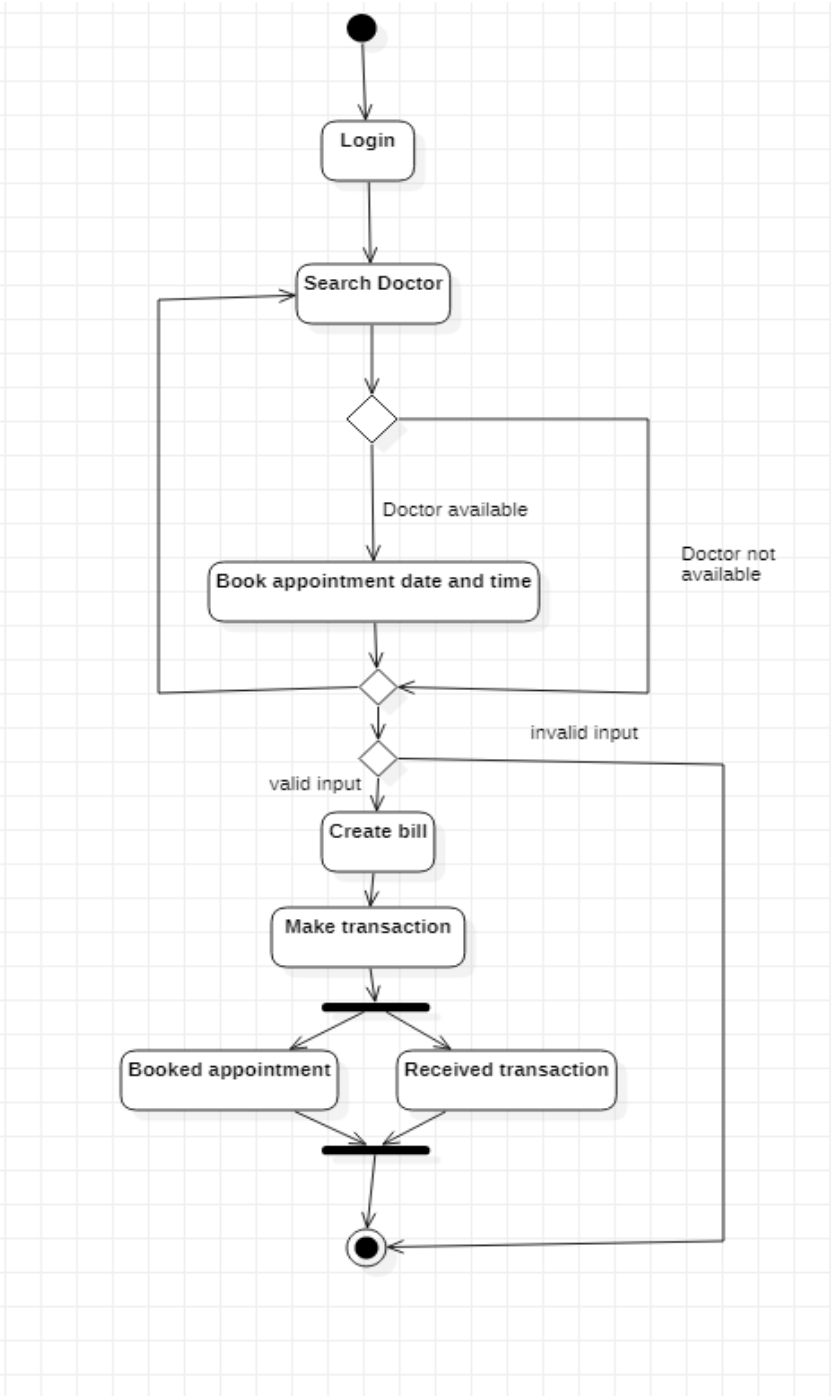
6.3.3. Reset Password



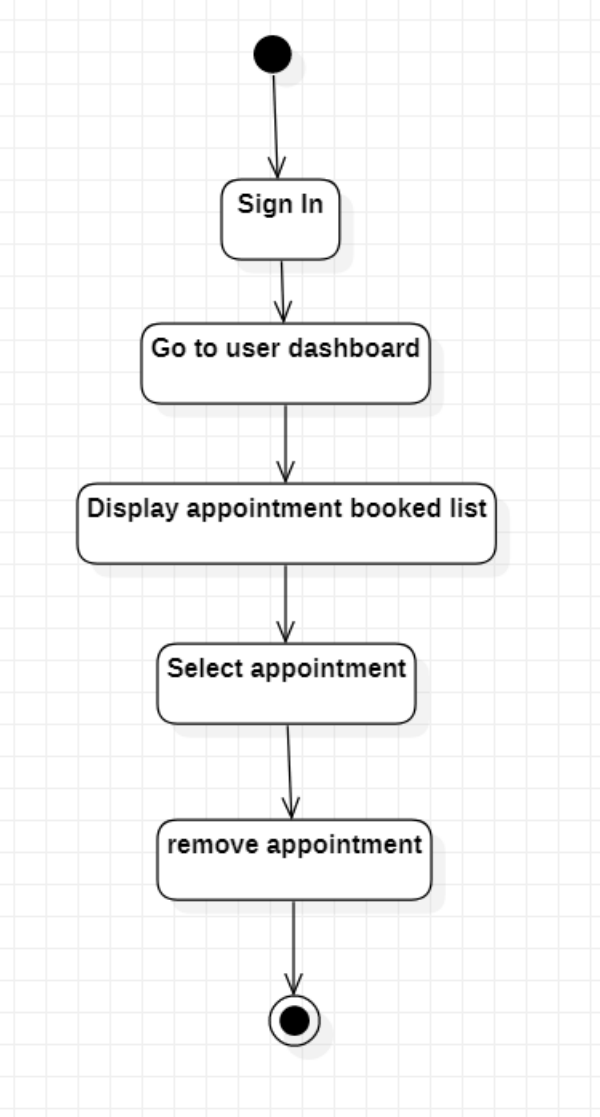
6.3.4. Sign In user



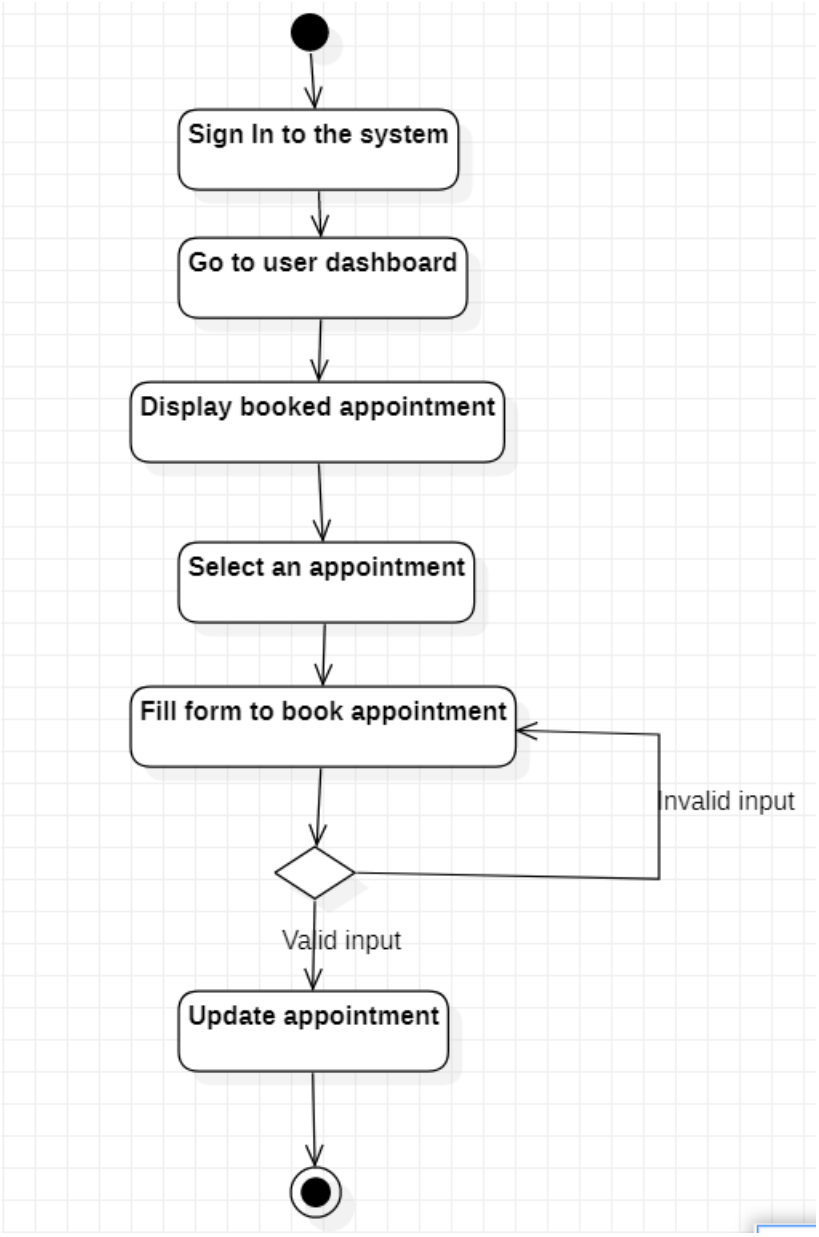
6.3.5. Book Appointment



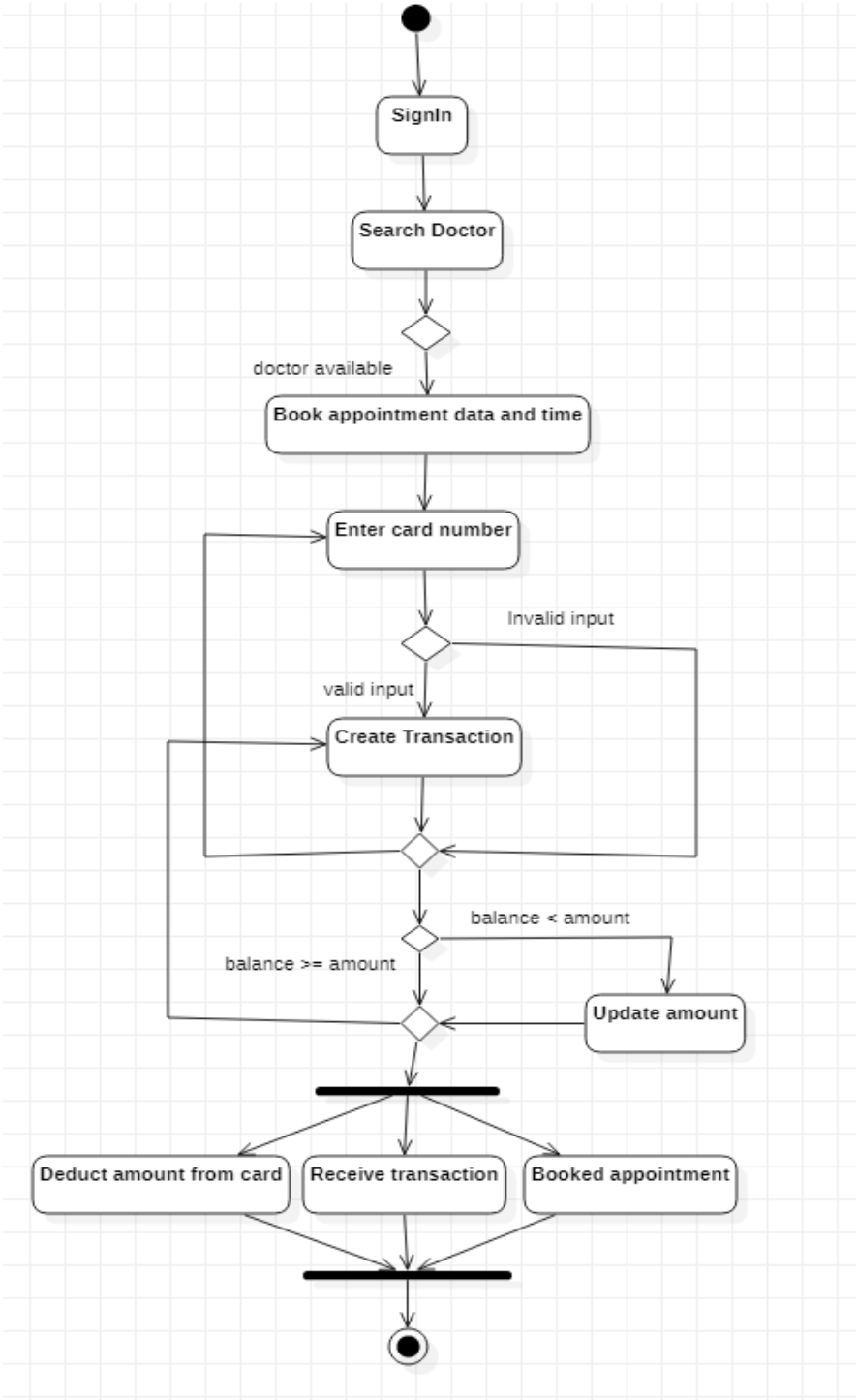
6.3.6. Remove Appointment



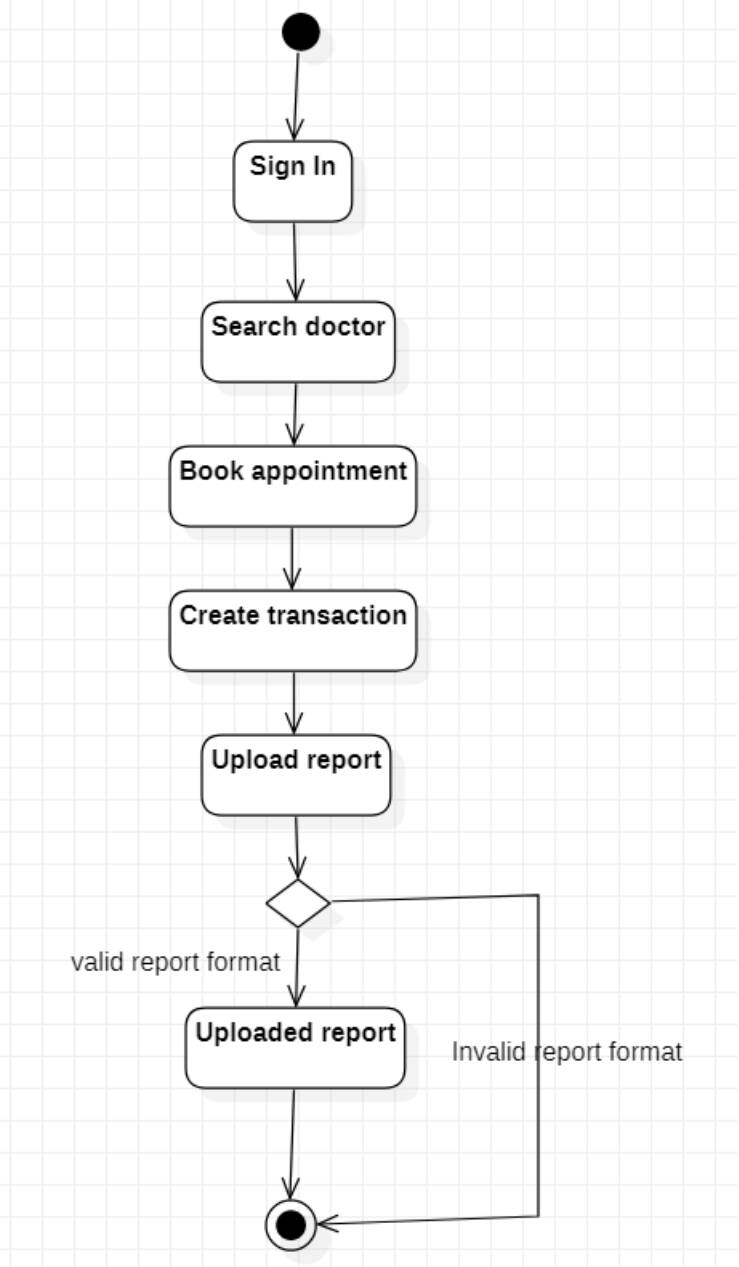
6.3.7. Update Appointment



6.3.8. Make Transaction

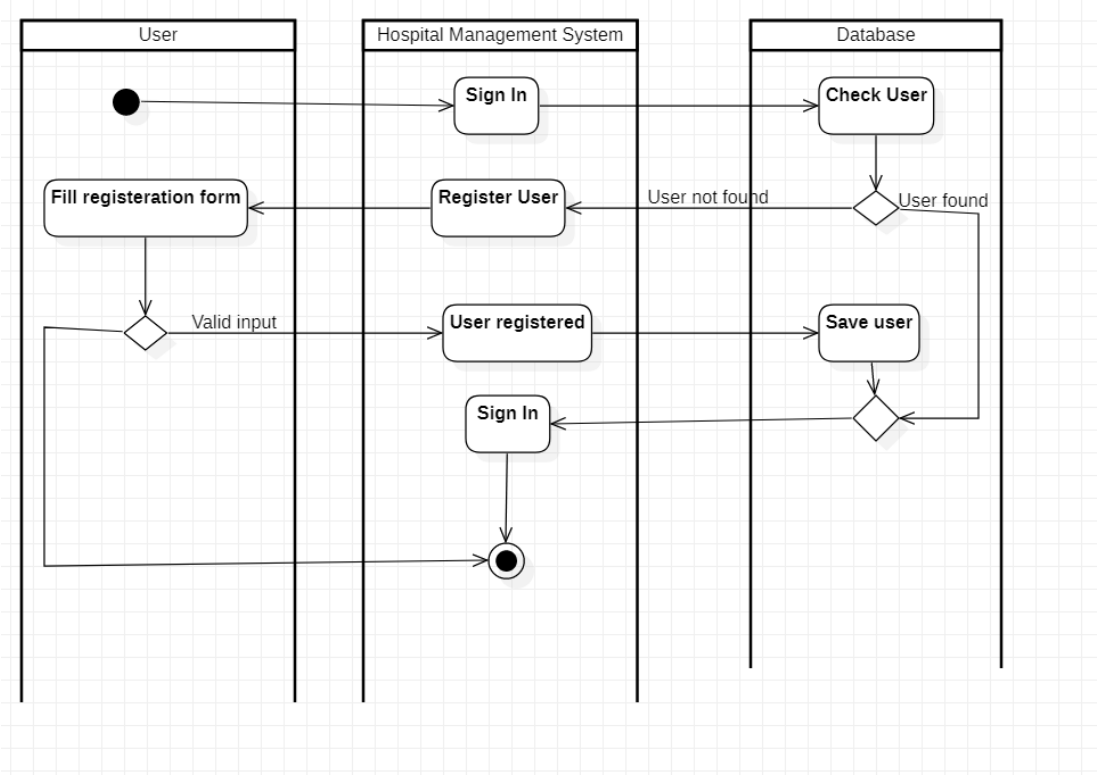


6.3.9. Uplaod Report

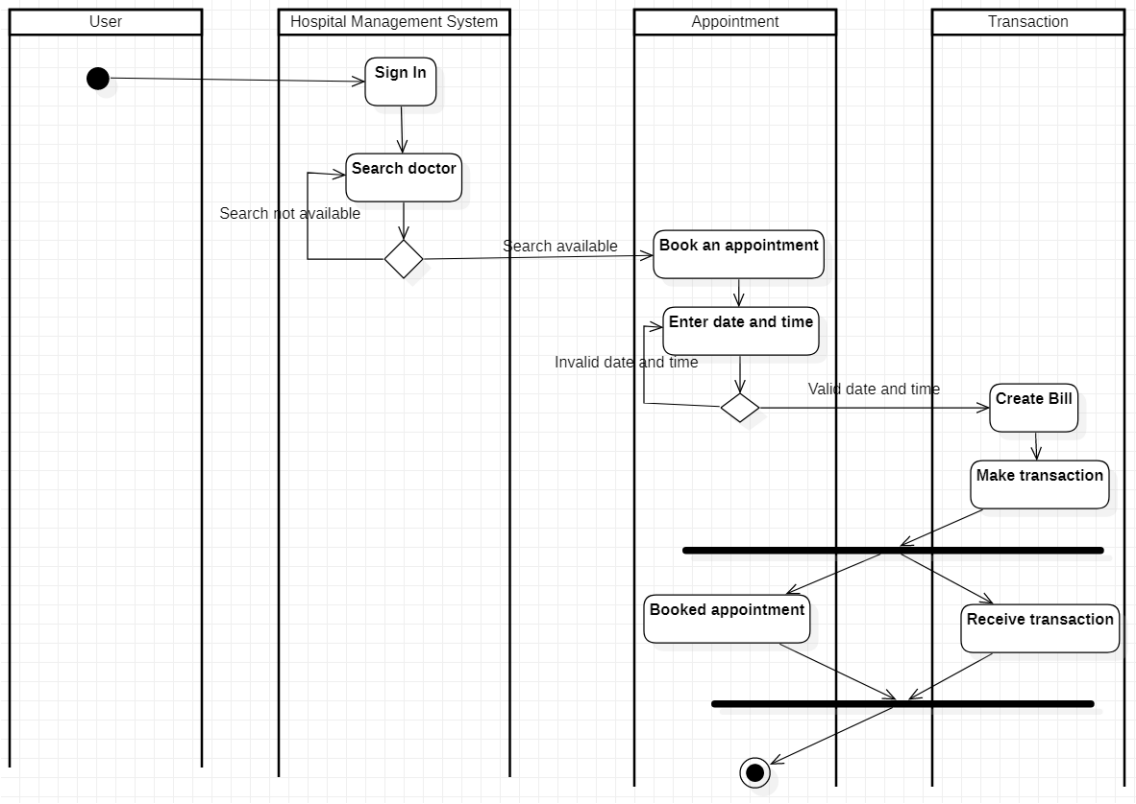


6.4.Swimlane Diagram

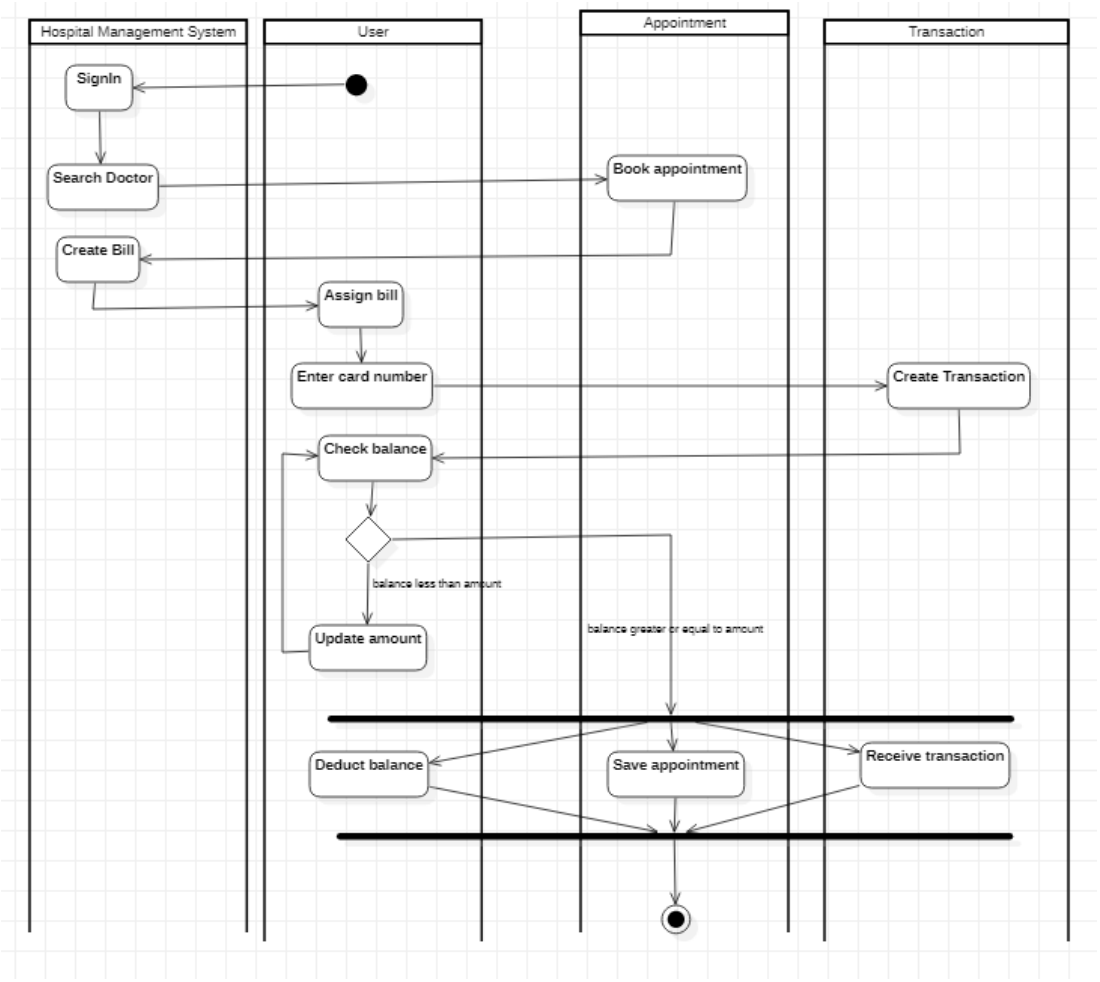
6.4.1. User Registeration



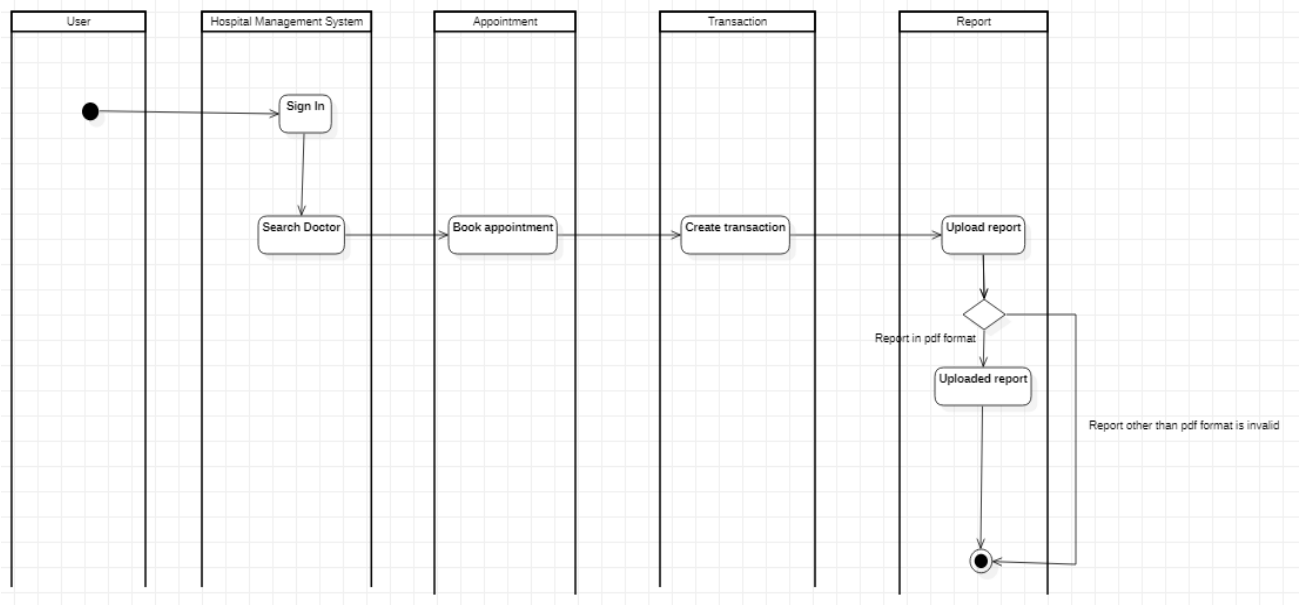
6.4.2. Book appointment



6.4.3. Make Transaction

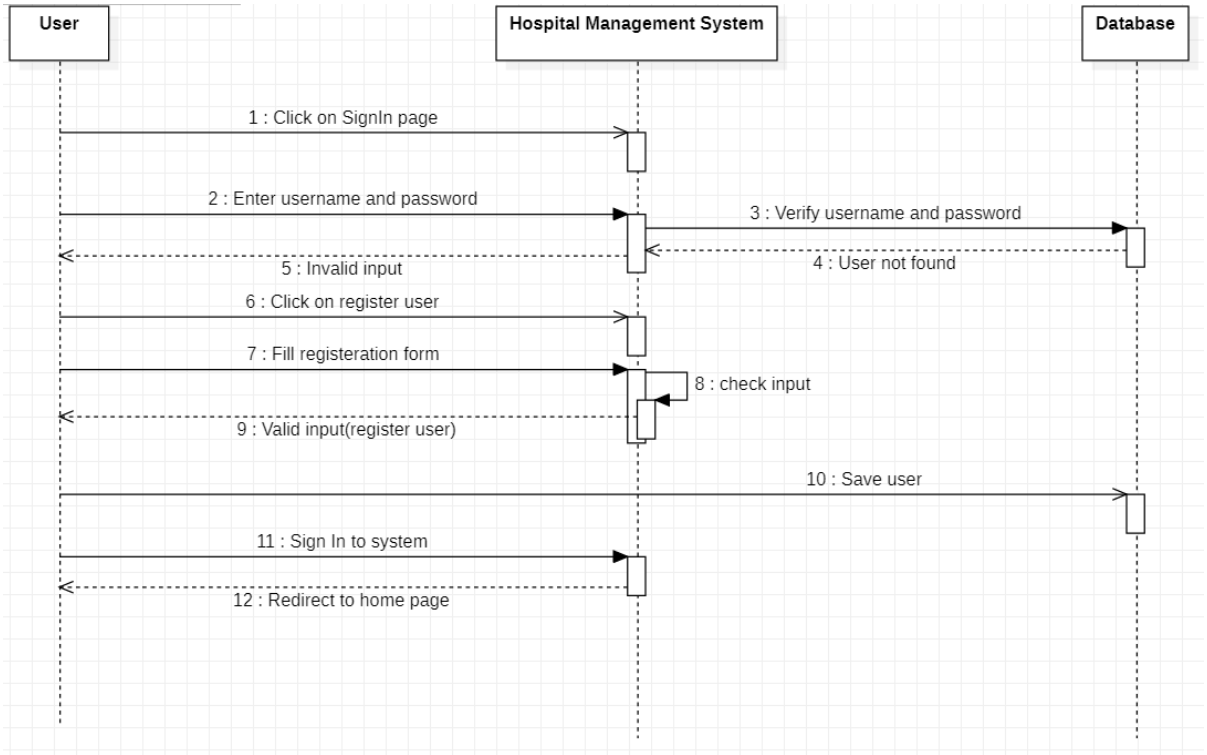


6.4.4. Upload Report

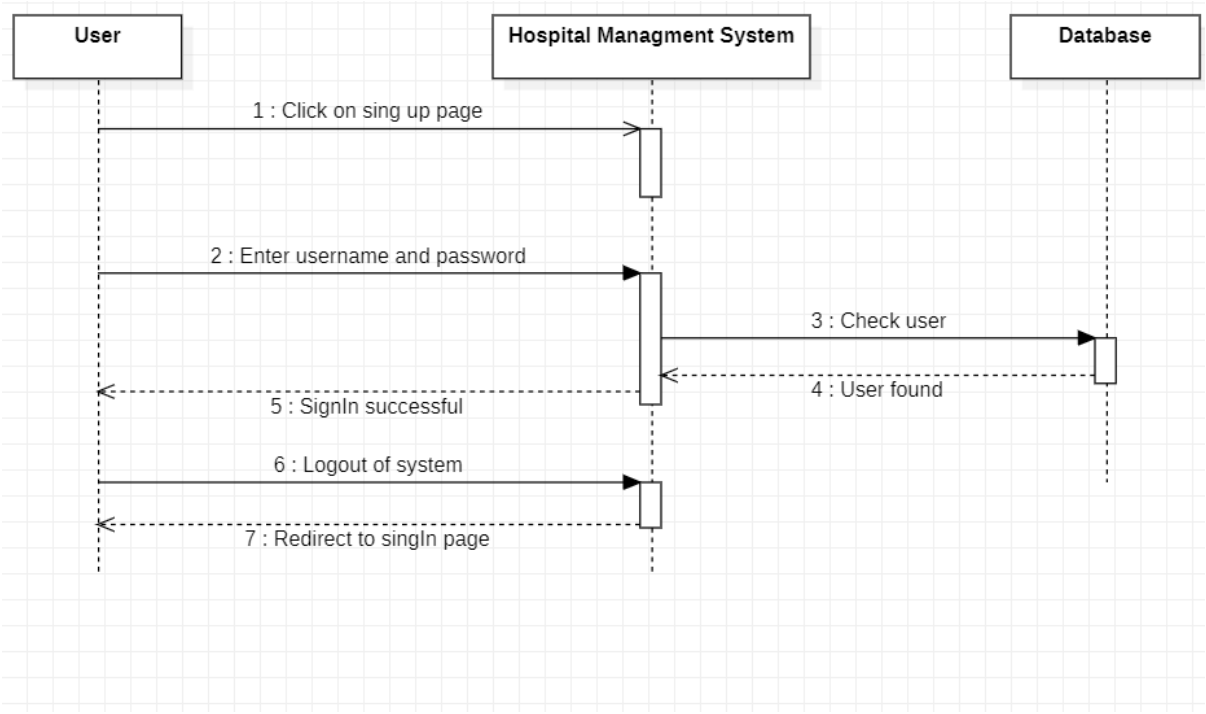


6.5.Sequence Diagram

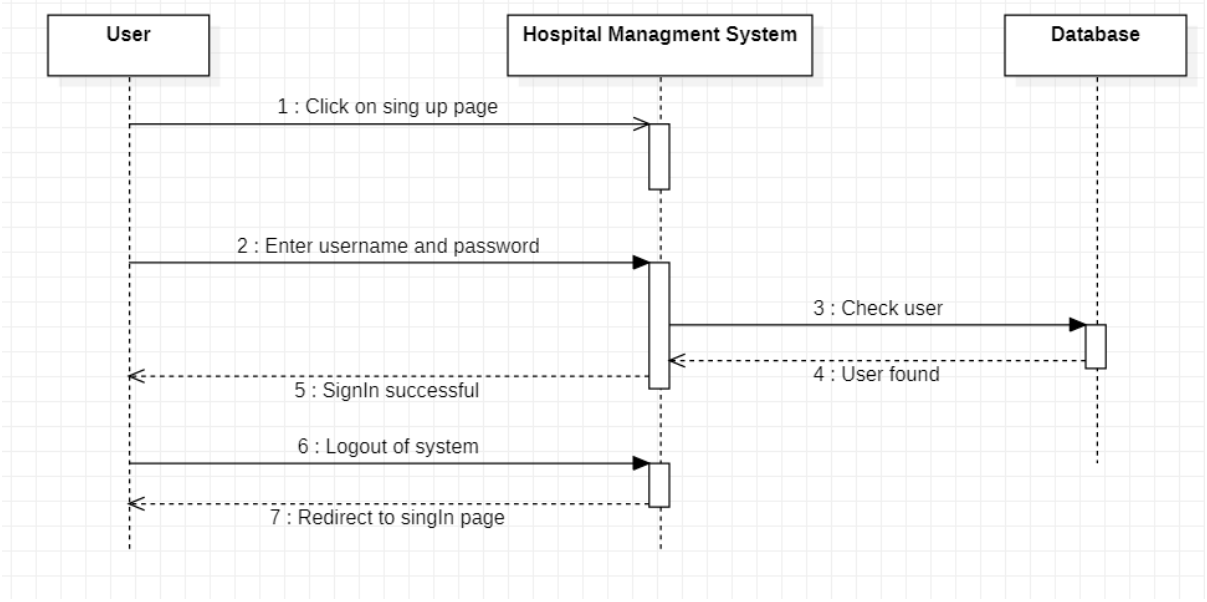
6.5.1. User Registeration



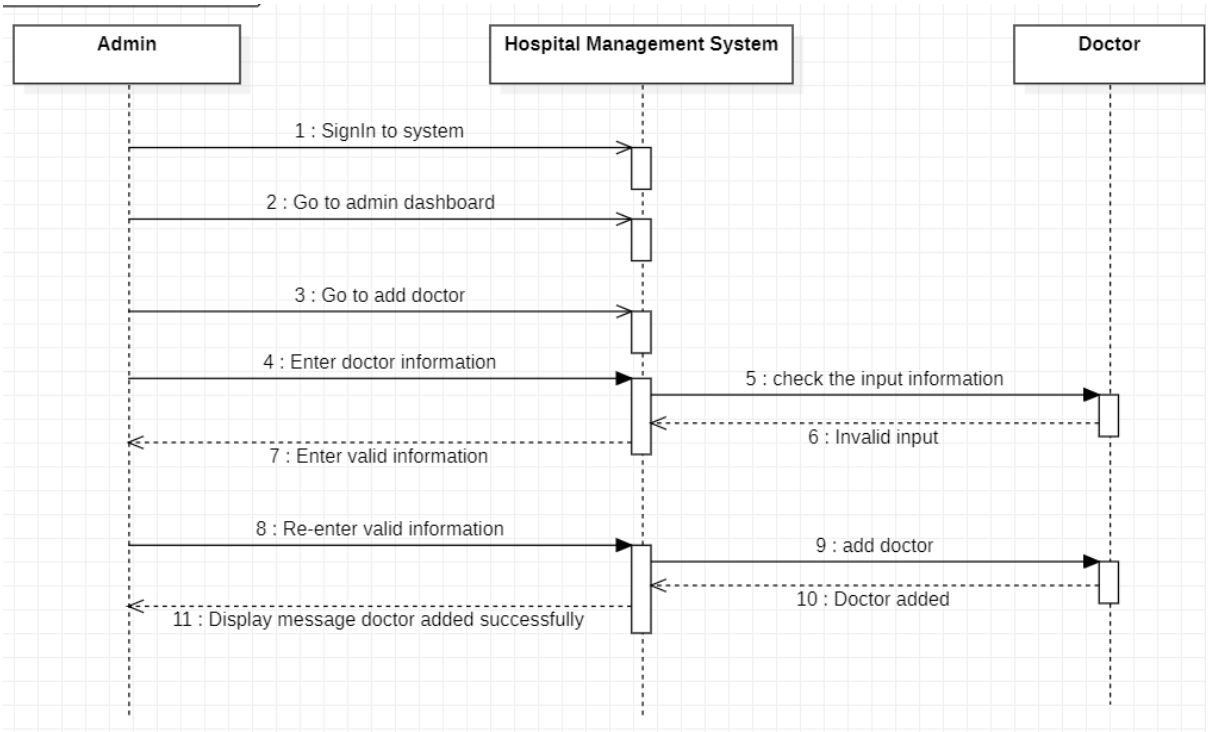
6.5.2. Sign In user



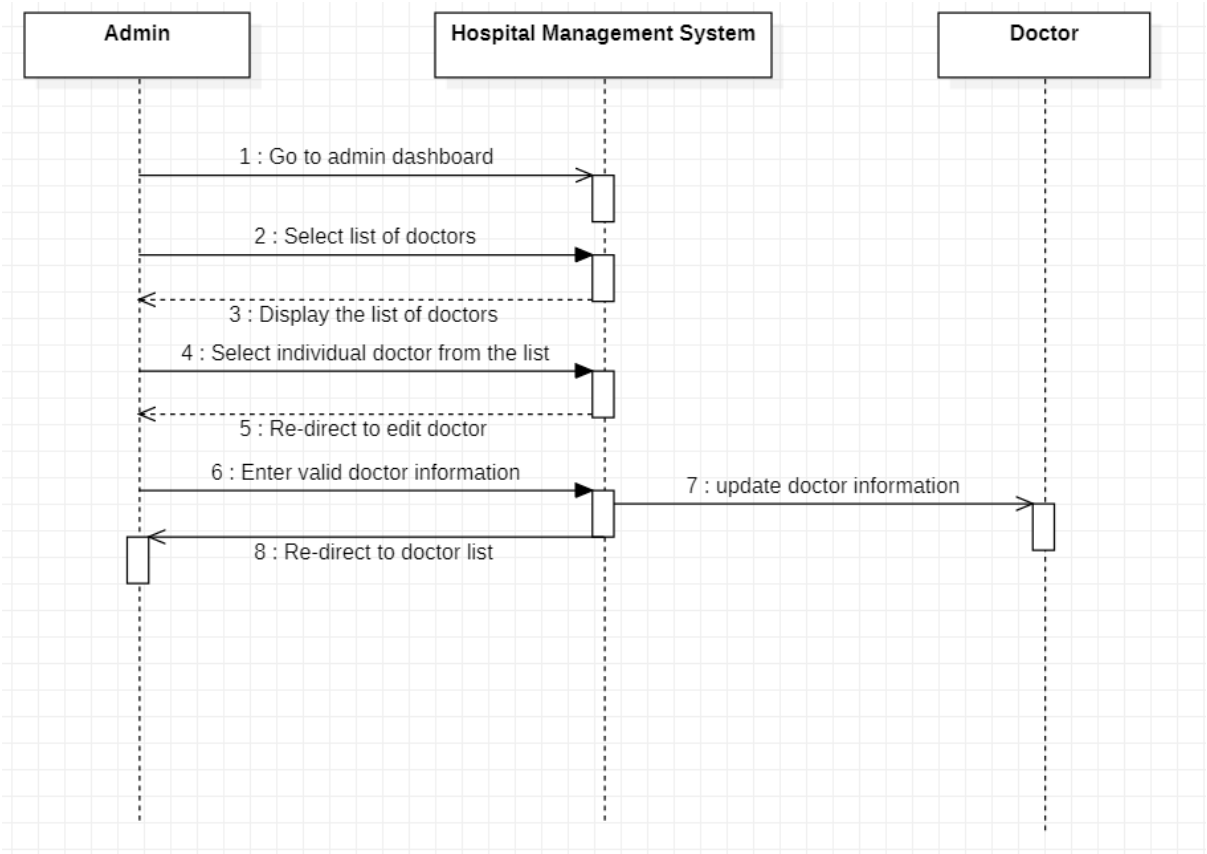
6.5.3. Sign Up user



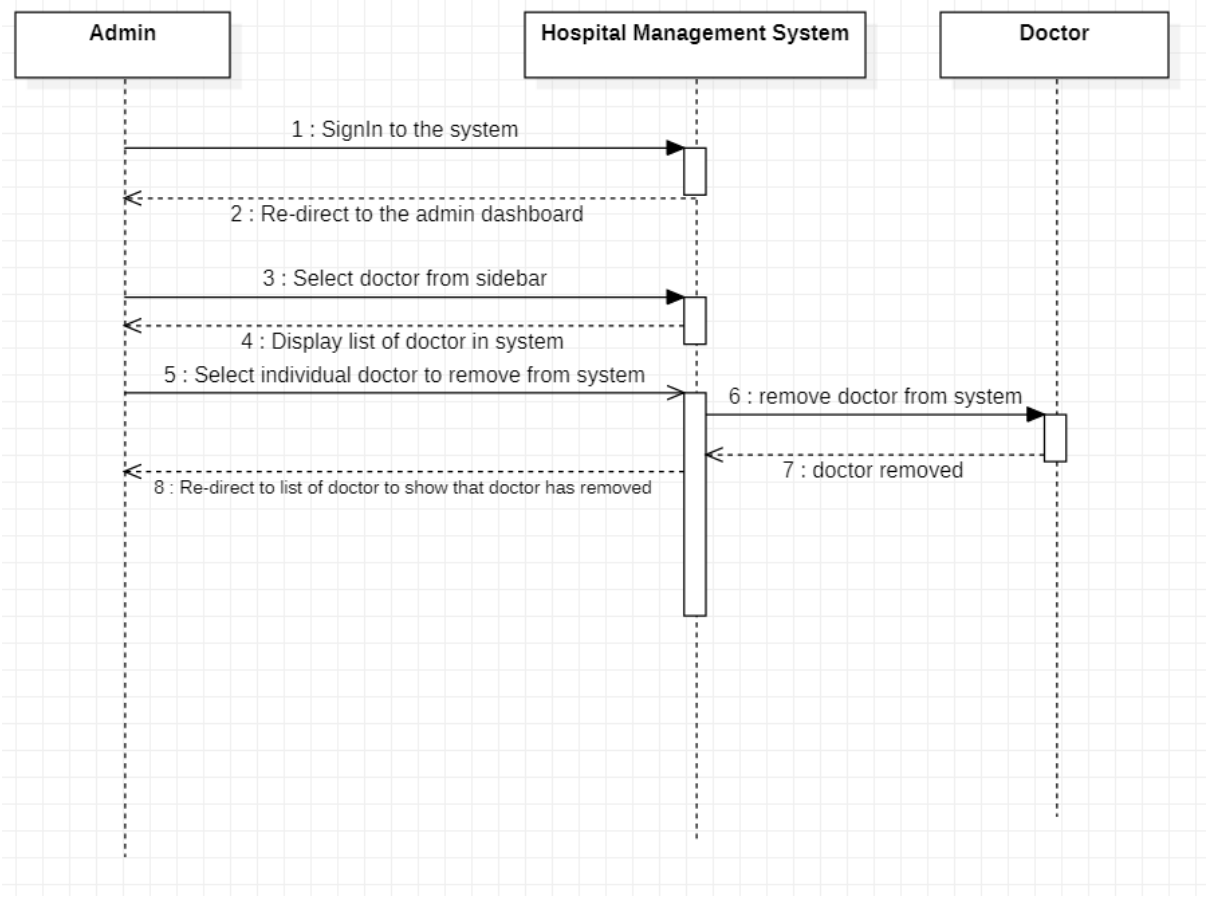
6.5.4. Add doctor



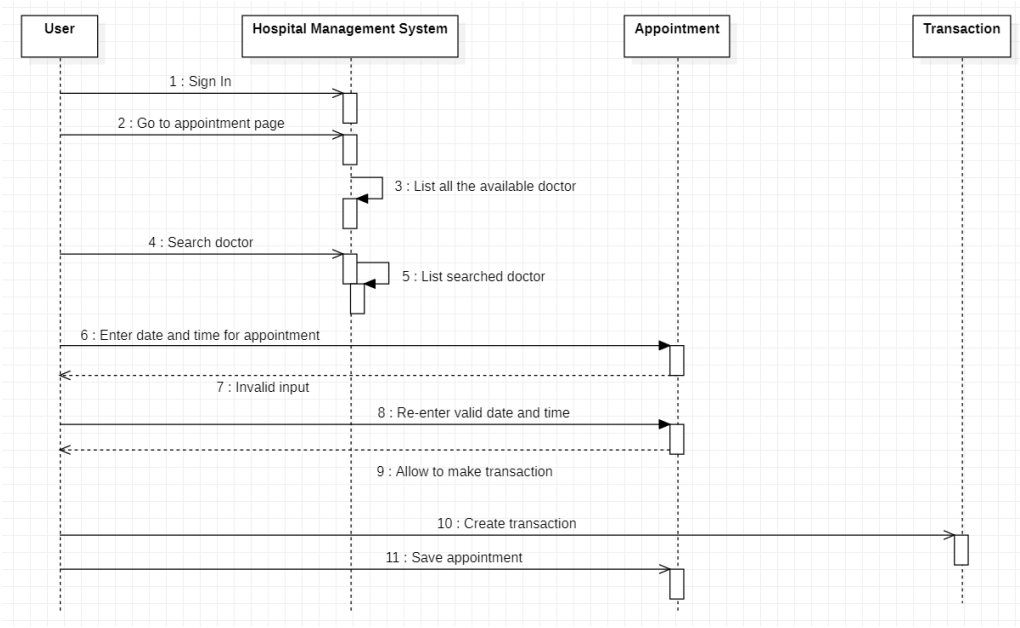
6.5.5. Update Doctor



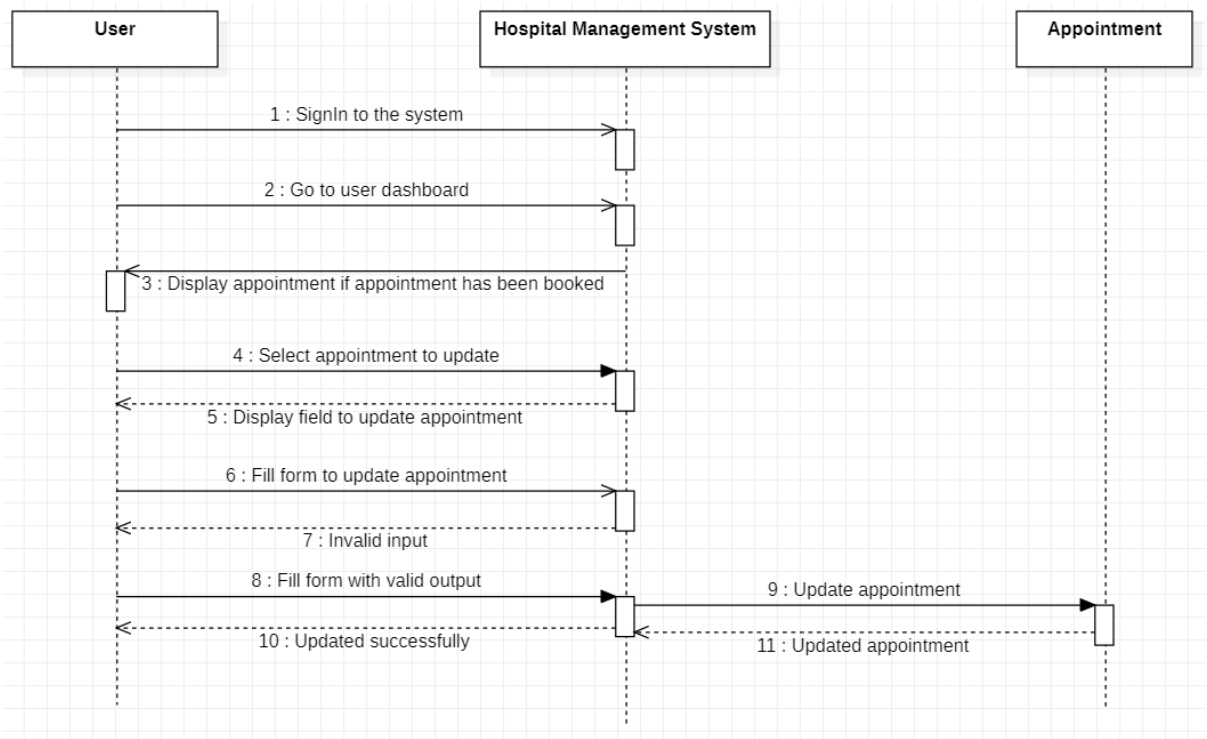
6.5.6. Remove Doctor



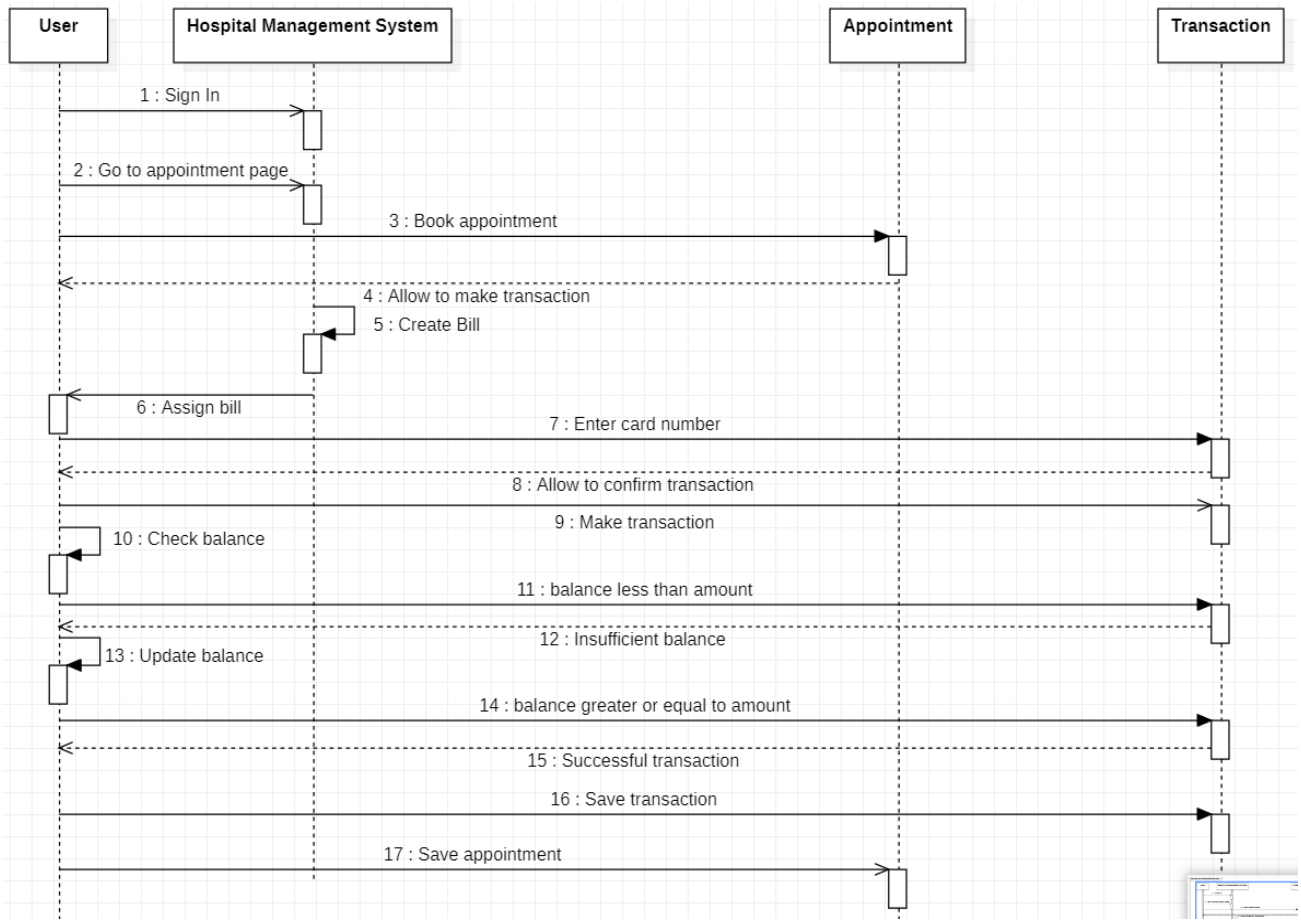
6.5.7. Book Appointment



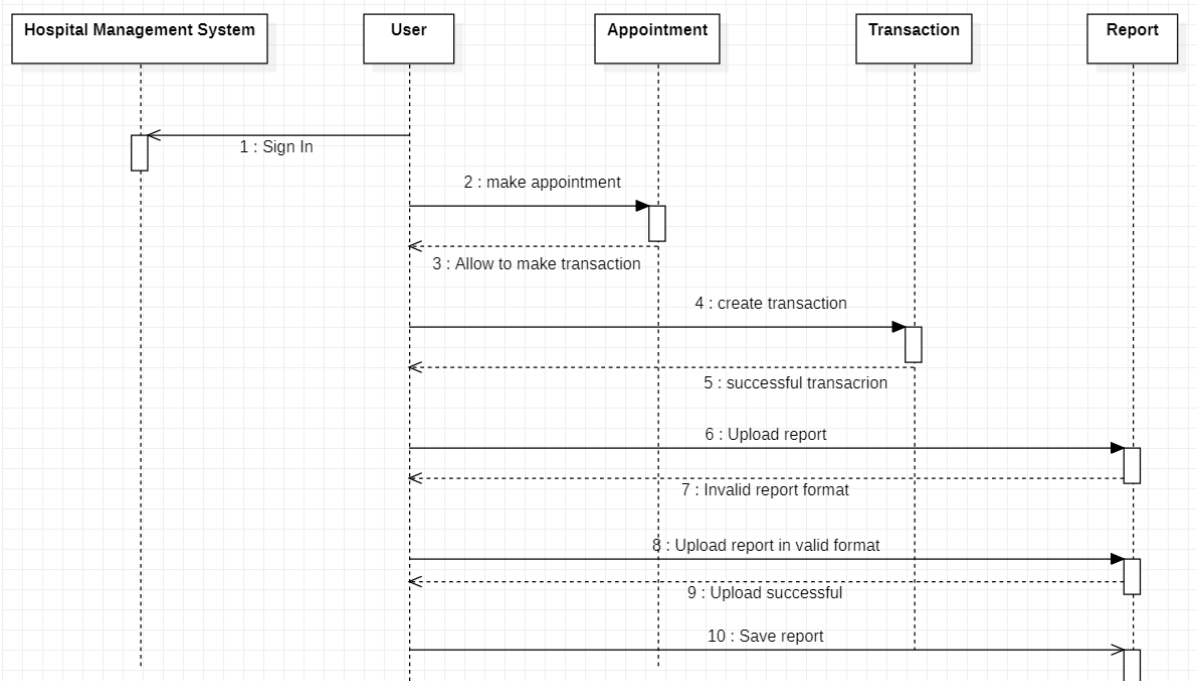
6.5.8. Update Appointment



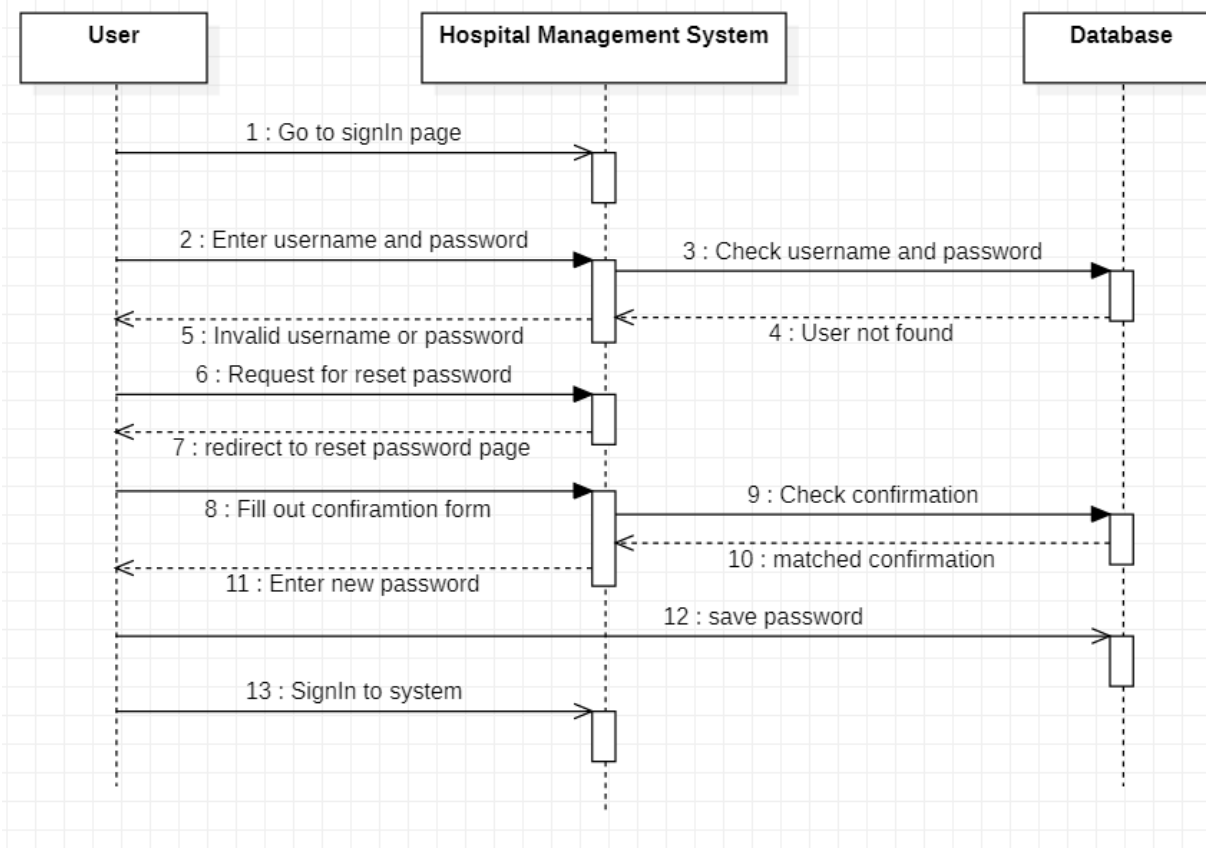
6.5.9. Make Transaction



6.5.10. Upload Report



6.5.11. Reset Password



6.6.Data Dictionary

6.6.1. User

Attributes	Data Type	Length	Key	Null value	Description
User_id	integer	5	Primary Key	Required	User_id is the unique identifier for each registered user in the system.
firstName	varchar	20		Required	firstName is the first name of the user and it can also include a middle name.
lastName	varchar	20		Required	lastName is the last name of the user.
userName	varchar	20		Required	This is the unique name that each registered user should have to be recognized in the system after User_id. useName is required to login to the system.
emailAddress	varchar	100		Required	emailAddress is the email address of the user that is used to register into the system.
doctor_id	integer	5	Foreign Key	Optional	doctor_id is the foreign key reference from the Doctor table. This key will indicate used to identify to doctor assign to the user if necessary.

6.6.2. Doctor

Attributes	Data Type	Length	Key	Null value	Description
doctor_id	Integer	5	Primary Key	Required	doctor_id is the unique identifier for each registered user in the system.
doctorName	varchar	20		Required	doctorName is the name of the doctor that includes both first name, middle name, and last name.
emailAddress	varchar	100		Required	emailAddress is the email of the doctor.
specialization	varchar	500		Required	specialization is the number of specialized fields in medicine.
qualification	varchar	500		Required	qualification is the number of degrees that the doctor is specialized to.
profileImage	image	-		Optional	profileImage is the image of the doctor. The image will be helpful for the patient to recognize their doctor.

6.6.3. Appointment

Attributes	Data Type	Length	Key	Null value	Description
appointment_id	integer	5	Primary Key	Required	appointment_id is the unique identifier for each appointment made by the user.
appointment_date	date	-		Required	appointment_date is the date that the user has chosen to meet their doctors. The date should be valid.
appointment_time	time	-		Required	appointment_time is the time that the user has chosen to meet their doctors. Time should be valid.
patient_id	integer	5	Foreign Key	Required	patient_id is the foreign key reference that refers to the patient who has made an appointment.
doctor_id	integer	5	Foreign Key	Required	Doctor_id is the foreign key reference that refers to the doctor with whom the patient has made an appointment.

6.6.4. Transaction

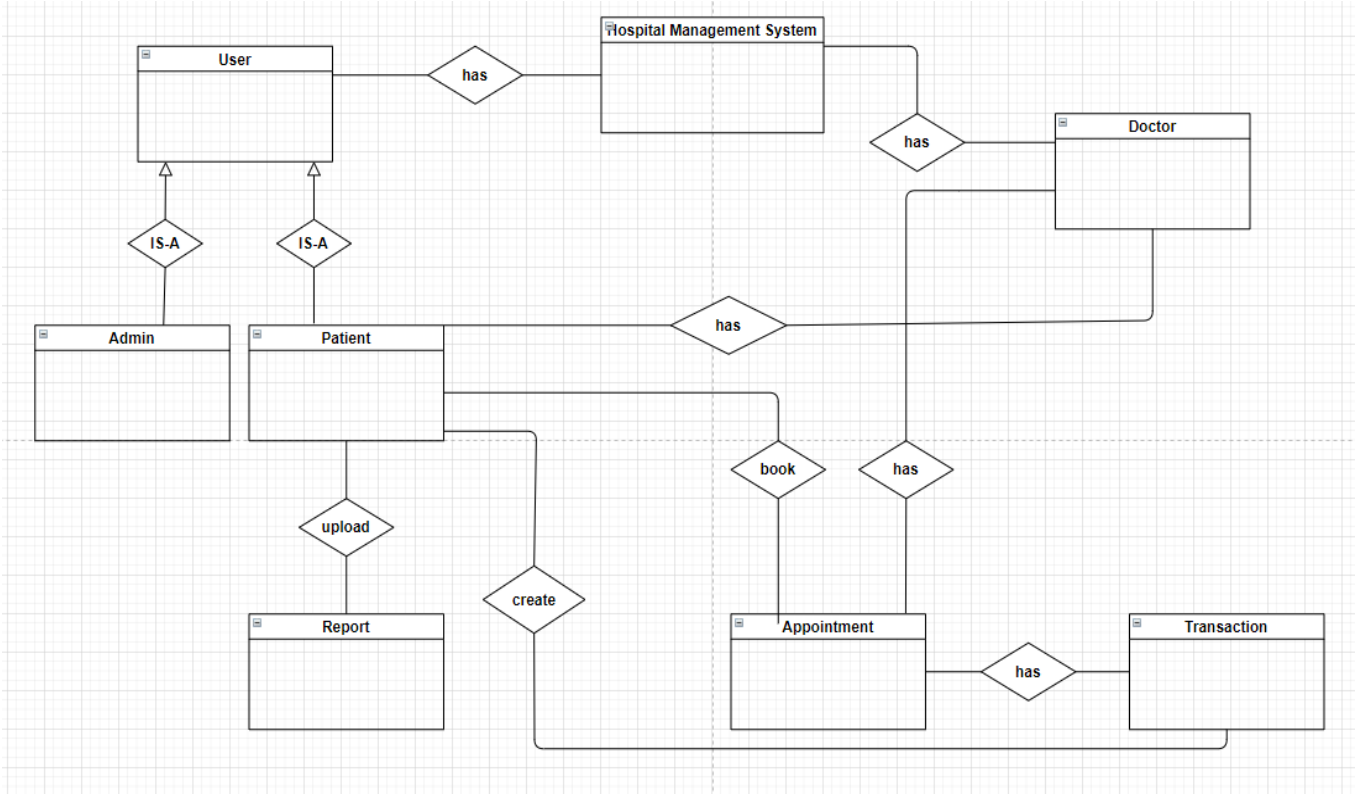
Attributes	Data Type	Length	Key	Null value	Description
transaction_id	integer	5	Primary Key	Required	transaction_id is the unique identifier that identifies each transaction made by the user.
payment_amount	integer	10		Required	payment_amount is the amount of appointment fee that the user has to pay to consider the appointment to be valid.
doctor_id	integer	5	Foreign Key	Required	doctor_id is the foreign key reference that is used to denote the transaction made for a specific doctor.
user_id	integer	5	Foreign Key	Required	user_id is the foreign key reference that is used to denote the user who has made a transaction.

6.6.5. Report

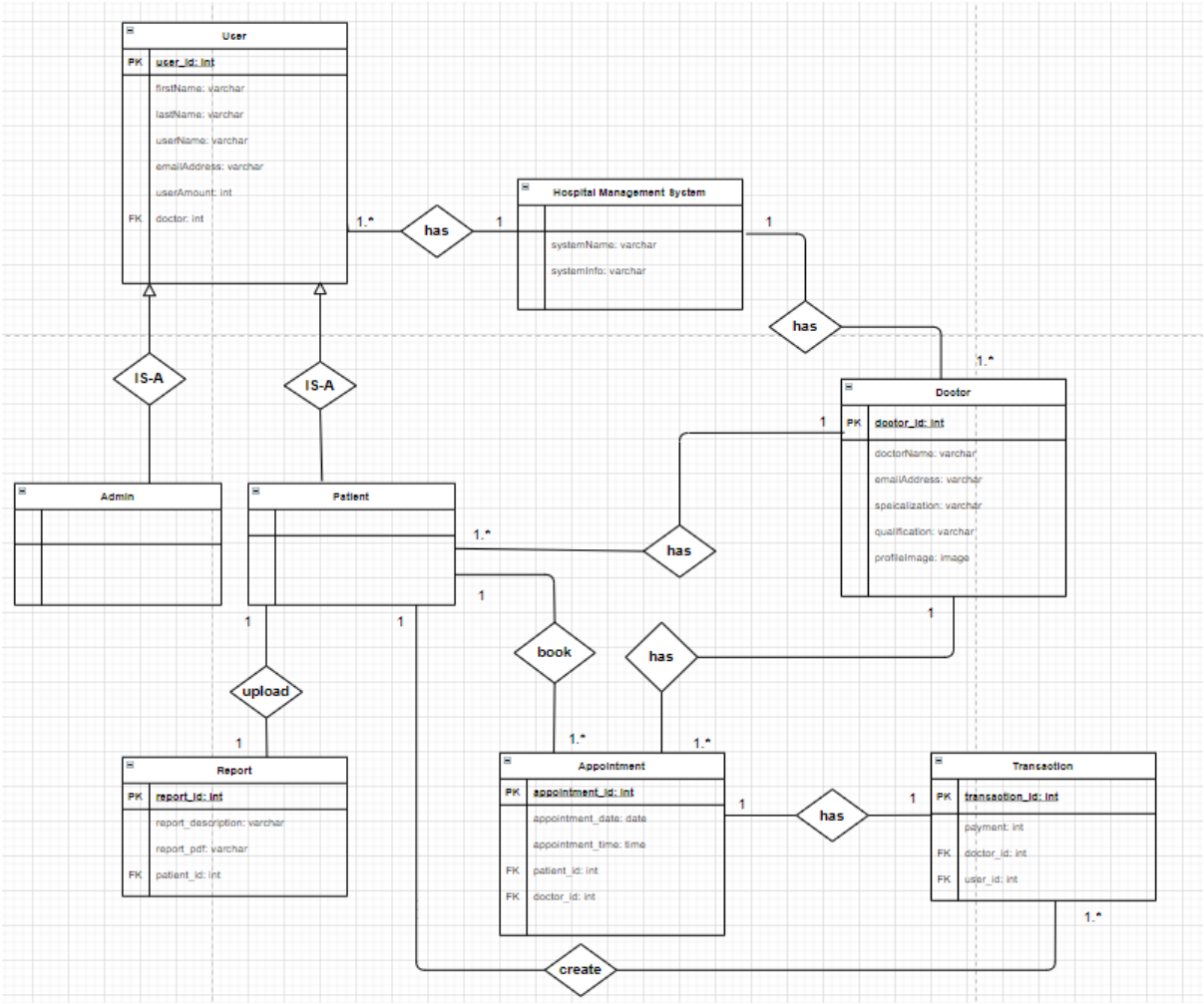
Attributes	Data Type	Length	Key	Null value	Description
report_id	integer	5	Primary Key	Required	report_id is the unique identifier to the report that the user has uploaded.
report_description	varchar	500		Optional	report_descrption is the description of what the medical report is about.
report_pdf	pdf	-		Required	peport_pdf is the report in pdf format.
patient_id	integer	5	Foreign Key	Required	patient_id is the foreign key reference to identify the patient who has uploaded the report.

6.7.Entity-Relationship Diagram

6.7.1. Conceptual Entity-Relationship Diagram



6.7.2. Detailed Entity-Relationship Diagram



Bibliography

Abdelaziz, T. M., Maatuk, A. M. & Rajab, F., 2016. AN APPROACH TO IMPROVEMENT THE USABILITY IN SOFTWARE PRODUCTS. *International Journal of Software Engineering & Applications (IJSEA),* 7(2), pp. 11-18.

ACADEMIA;, 2020. *Waterfall Model.* [Online]
Available at: https://www.academia.edu/38314403/WATERFALL_MODEL
[Accessed 22 May 2020].

AI, WELCOME;, 2019. *Buoy Health.* [Online]
Available at: https://www.welcome.ai/buoy-health
[Accessed 22 May 2020].

Anwer, F., Aftab, S., Muhammad, S. S. & Waheed, U., 2017. Comparative Analysis of Two Populalr Agile Process Models: Extreme Programming and Scrum. *International Journal of Computer Science and Telecommunications,* 8(2), pp. 1-7.

Anwer, F., Aftab, S., Muhammad, S. S. & Waheed, U., 2017. Comparative Analysis of Two Popular Agile Process Models: Extreme Programming and Scrum. *International Journal of Computer Science and Telecommunication,* 8(2), pp. 1-7.

Ayanouz, S., Abdelhakim, B. A. & Benhmed, M., 2020. A Smart Chatbot Architecture based NLP and Machine Learning for Health Care Assistance. *NISS2020: Proceedings of the 3rd International Conference on Networking, Information Systems & Security,* March, pp. 1-6.

Babylon;, 2020. *Understanding you current and future health.* [Online]
Available at: https://www.babylonhealth.com/us/what-we-offer/healthcheck
[Accessed 1 October 2020].

Bargagli-Stoffi, F. J., Niederreiter, J. & Riccaboni, M., 2020. *Supervised Learning for the Prediction of Firm Dynamics,* s.l.: s.n.

Bathaee, Y., 2018. THE ARTIFICIAL INTELLIGENCE BLACK BOX AND THE FAILURE OF INTENT AND CAUSATION. *Harvard Journal of Law & Technology,* 31(2), pp. 901-902.

Bohmer, R. M., Pisano, G. P., Sadun, R. & Tsai, T. C., 2020. How Hospitals Can Manage Supply Shortages as Demand Surges. *Harvard Business Publishing* , 03 April, p. 1.

Dand, M., 2018. 10 Questions to Help You Build Your First Chatbot. *Medium,* 8 October, p. 1.

Deng, L. & Liu, Y., 2018. A Joint Introduction to Natural Language Processing and Deep Learning. In: Y. L. Li Deng, ed. *Deep Learning in Natural Language Processing.* 1 ed. USA: Springer Singapore, pp. 1-3.

Donges, N., 2020. A GUIDE TO RNN: UNDERSTANDING RECURRENT NEURAL NETWORKS AND LSTM. *builtin,* 3 September, p. 1.

Futurist , The Medicial ;, 16. The Top 12 Health Chatbots. *The Medical Futurist*, 2020 January, p. 1.

Futurist , The Medicial ;, 16. The Top 12 Health Chatbots. *The Medical Futurist*, 2020 January, p. 1.

Gambhir, P., 2019. *REVIEW OF CHATBOT DESIGN AND TRENDS.* Delhi, Artificial Intelligence and Speech Technology 2019.

Garha, M., 2017. Health care in Nepal: An observational perspective. *Journal of Nursing Education and Practice,* 7(1), pp. 114-117.

Haristiani, N., 2019. *Artificial Intelligence (AI) Chatbot as Language Learning Medium: An inquiry.* s.l., International Conference on Education, Science and Technology 2019.

Hawig, D., 2018. *Florence.chat: How to improve the adoption of Chatbots in Healthcare.* [Online]
Available at: https://www.healthcare.digital/single-post/2018/02/17/Florencechat-How-to-improve-the-adoption-of-Chatbots-in-Healthcare
[Accessed 6 October 2020].

James, E., 2019. Overview of Django - MVT (Model View Template) Structure. *c-shrapcorner*, 30 October, p. 1.

Kepuska, V. & Bohouta, G., 2018. *Next-Generation of Virtual Personal Assistants (Microsoft Cortana, Apple Siri, Amazon Alexa and Google Home).* Las Vegas, USA, 2018 IEEE 8th Annual Computing and Communication Workshop and Conference (CCWC)..

Khan, M. E., 2010. Different Forms of Software Testing Techniques for Finding Errors. *IJCSI International Journal of Computer Science Issues,* 7(3), p. 11.

Kousa, E., 2019. *Exploring Success Factors in Chatbot Implementation Projects*, s.l.: ARCADA.

Liu, P., Qiu, X. & Huang, X., 2016. *Recurrent Neural Network for Text Classification with Multi-Task Learning,* 825 Zhanheng Road, Shanghai, China: School of Computer Science, Fudan University.

Lopatovska, I. et al., 2018. Talk to me: Exploring user interactions with the Amazon Alexa. *Journal of Librarianship and Information Science*, 7 March, pp. 1-14.

Lu, M. & Liang, P., 2017. *Automatic Classification of Non-Functional Requirements from Augmented App User Reviews.* Karlskrona, Sweden, Association for Computing Machinery.

Lutters, E., 2014. Requirement Specification. In: I. A. f. P. E., L. Laperriere & G. Reinhart, eds. *CIRP Encyclopedia of Production Engineering.* Berlin: Springer, pp. 1-4.

Marr, B., 2019. Services Babylon Offers. *The Amazing Ways Babyon Health Is Using Artificial Intelligence To Make Healthcare Universallt Accessible*, 16 August, p. 1.

neveonProjects;, 2020. *Artificial Intelligence HealthCare Chatbot System.* [Online]
Available at: https://nevonprojects.com/artificial-intelligence-healthcare-chatbot-system/
[Accessed 22 March 2020].

Pathak, K. & Arya, A., 2019. *A Metaphorical Study Of Variants Of Recurrent Neural Network Models For A Context Learning Chatbot.* Mathura, India, India, IEEE, pp. 768-772.

Projects, nevon;, 2019. *Artificial Intelligence Healthcare Chatbot System.* [Online]
Available at: https://nevonprojects.com/artificial-intelligence-healthcare-chatbot-system/
[Accessed 22 March 2020].

Sabriye, A. O. J. & Zainon, W. M. N. W., 2017. *A Framework For Detecting Ambiguity In Software Requirement Specification.* Amman, Jordan, IEEE.

S, D. et al., 2018. A Self-Diagnosis Medical Chatbot Using Artificial Intelligence. *Journal of Web Development and Web Designing,* 3(1), pp. 1-7.

Sennaar, K., 2019. Chatbots for Healthcare- Comparing 5 Current Applicatons. *emerj*, 13 December, p. 1.

Sennar, K., 2019. Chatbots for Healthcare - Comparing 5 Current Applications. *Emerj*, 13 December, p. 1.

Shum, H.-. y., He, X. & Li, D., 2018. From Eliza to XiaoIce: Challenges and Opportunities with Social Chatbots. *Frontiers of Information Technology & Electronic Engineering*, 08 January.

Sinha, A., 2020. Understanding of LSTM Networks. *geeksforgeeks,* 11 May, p. 1.

Sneha, M. K. & M, M. M. G., 2017. *Research on Software Testing Techniques and Software Automation Testing Tools.* s.l., International Conference on Energy, Communication, Data Analytics and Soft Computing (ICECDS).

Sreelakshmi, K., Rafeeque, P. C., Sreetha, S. & Gayathri, E. S., 2018. *Deep Bi-Directional LSTM Network for Query Intent Detection.* s.l., Elseview B.V..

Susilowati, M., Ahsan, M. & Kurniawan, Y., 2019. What does the software requirement specification for local E-Government of citizen database information system? An analysis using ISO/IEC/IEEEE 29148-2011. *Journal of Physics: Conference Series* , 1402(2), pp. 1-5.

Tran, N., Cori, P., Morales, A. & Johnson, E., 2020. *PLATICA: Personalized Language Acquisition Training & Instructions Chatbot Assistant,* California: Santa Clara University.

V., V., Cooper, J. B. & J., R. L., 2020. *Algorithm Inspection for Chatbot Performance Evaluation.* s.l., Elsevier B.V..

VanderPlas, Jake;, 2016. Bayesian Classification. In: D. Schanafelt, ed. *Python Data Science Handbook.* United States of America: O'Reilly Media, Inc., p. 383.

WelcomeAI;, 2020. *Buoy Health.* [Online]
Available at: https://www.welcome.ai/buoy-health
[Accessed 11 September 2020].

Zaremba, W., Sutskever, I. & Vinyals, O., 2015. *RECURRENT NEURAL NETWORK REGULARIZATION.* s.l., International Conference on Learning Representations .

Zhou, L., Gao, J., Li, D. & Shum, H.-. Y., 2020. The Design and Implementation of XiaoIce, and Empathetic Social Chatbot. *Computational Linguistics,* 46(1), pp. 53-93.