## **(A typical Specimen of Cover Page & Title Page)**

**TITLE OF PROJECT**

**UCS503 Software Engineering Project Report End-Semester Evaluation**

**Submitted by:**

**(101803602)Amish Mengi**

**(101803607)NAME OF THE STUDENT (<University Roll Number>)NAME OF THE STUDENT (<University Roll Number>)NAME OF THE STUDENT BE Third Year, CoE/CoE MBA**

Submitted to Name of the Faculty

Designation

|  |  |
| --- | --- |
|  |  |
|  |  |

**Table of Contents**

**Revision History** ....................................................................................................................2

**1.Introduction** **........................................................................................................................3**

1.1 Purpose............................................................................................................3

1.2 Document Conventions....................................................................................3

.3 Intended Audience and Reading Suggestions ................................................3

.4 Project Scope ..................................................................................................4

1.5 References ......................................................................................................5

**2.Overall Description ............................................................................................................5**

2.1 Product Perspective ........................................................................................5

2.2 Product Features .............................................................................................6

2.3 User Classes and Characteristics ...................................................................6

2.4 Operating Environment ....................................................................................7

2.5 Design and Implementation Constraints ..........................................................7

2.6 User Documentation ........................................................................................7

2.7 Assumptions and Dependencies .....................................................................7

**3.System Features ................................................................................................................7**

3.1 Register User ...................................................................................................7

3.2 Login User .......................................................................................................8

3.3 Disease Prediction............................................................................................9

3.4 Appointments .................................................................................................10

3.5 Upload Reports ..............................................................................................11

3.6 View Patient Details .......................................................................................11

3.7 Prescription ....................................................................................................12

**4.External Interface Requirements ....................................................................................13**

4.1 User Interfaces ..............................................................................................13

4.2 Hardware Interfaces ......................................................................................13

4.3 Software Interfaces .......................................................................................13

4.4 Communications Interfaces ...........................................................................13

**5.Other Nonfunctional Requirements ...............................................................................13**

5.1 Performance Requirements ...........................................................................13

5.2 Safety Requirements .....................................................................................13

5.3 Security Requirements ..................................................................................14

5.4 Software Quality Attributes ............................................................................14

**1. Project Overview**

Antidote is a ‘Disease Prediction and Patient Management System’ software. People are reluctant to visit the hospitals due to the fear of contracting COVID-19 while hospitals too have closed-out patient wards for treatment. Apart from this, health officials and the government have continuously been advising the general public to avoid visiting the hospitals except for emergencies, resulting in these fears being compounded.

This portal provides an e-consultation facility to the patients. The patients can sign up at the portal. After logging in the patients would observe there are various sections that cater to their needs. A section for predicting the disease that they are likely to suffer from where they can fill a form by providing their details along with the symptoms witnessed. Based on the result of the prediction, the patients can search for the specialised doctors and share the prediction results with the selected doctor. The patients would also be provided with the section where they can upload their previous medical reports, thereby keeping a track of their medical history which often gets misplaced due to the conventional way of storing the reports in files.

The doctors across the country can register themselves on our portal specifying the fields that they have been working in. Once logged in then each doctor will receive notifications regarding the patients who have approached them along with their details including their symptoms. After analysing the reports of the patient, the doctor can provide a confirmation of the appointment and if required they can even provide the patient with the prescription that needs to be followed till the appointment date. This will help overcome the challenge of increased drop rate in the regular patient visits and also help patients to consult the doctors in case of emergency situations by fixing an appointment without the need to visit the hospital regarding the same.

**Product Features**:

* Registration/Login
* Disease prediction and Doctor suggestion
* Uploading medical reports
* Appointment Confirmation
* Prescription
* Edit profile
* Question and Answer(QnA)

This would reduce the chances of disease transmission and would also be convenient for both the patient and the doctor especially residing in high alert areas. This would also help to overcome the problem of deficiency of human resources in the health sector which is prevalent at several levels such as between regions, between rural and urban areas and between private and public sectors. This challenge is solved by providing the feature that doctors throughout the country can sign up at our portal thus providing the benefit of consulting the health care specialists in the respective fields and bridging the gap between different sections of the society.

**2. SRS**

**1. Introduction**

**1.1 Purpose**

The purpose of this document is to provide an overview of the software product that we aim to develop and its requirements i.e hardware , software specifications and intended audience. The document has been formatted in such a way that the deliverables are divided into smaller components thereby, describing the functions,goals and tasks that the system can perform. The product aims to automate the Patient Management System and helps the user predict the disease he/she is likely to have based on the symptoms.The document helps the developers to understand what functionality needs to be worked upon and in which order along with understanding the boundaries within which they need to work.

**1.2 Document Conventions**

Bold-faced text has been used to emphasize section and subsection headings.The Font is uniform across the document which is Arial with size 11. Line spacing for the whole document is 1.15 with 2 lines between different sections. The text in paragraphs is justified. Numbered lists are used to show subsections of a section.

**1.3 Intended Audience and Reading Suggestions**

The audience that the software product being developed targets:

1. The patients who want to get a prediction of the disease that they may be suffering from based on the symptoms observed and who want to keep a track of their previous medical reports so that they can access whenever they want.
2. The doctors who can review the patient’s details and the results of disease prediction, thereby providing the patient with the appointment confirmation details and the prescription that can be followed.

The document has been divided into sections and further sub-sections depending on the aspects that need to be covered while developing the product so as to meet the required specifications.The first sections gives an overview of the project- the scope and the intended audience, the conventions that are to be followed throughout the document.The second section examines the various perspectives of the product which includes the features included,viewpoints of different users to be considered, the limitations and the challenges to be faced while developing the product, the dependencies required and the assumptions to be kept in mind. This section needs to be studied in detail by the developers since it covers each and every detail which will be required while developing the product.The third section covers the detailed description of each system feature.The users and testers must carefully read this section.These sections deal with the functional requirements of the project. The last section covers various non-functional aspects such as the safety,security and performance requirements which needs to be taken care of by the project manager.

**1.4 Project Scope**

The goal here is to develop a ‘Disease Prediction and Patient Management System’ software. The scope of this product will increase in the near future if we look at the current pandemic situation. People are reluctant to visit the hospitals due to the fear of contracting COVID-19 while hospitals too have closed-out patient wards for treatment. Apart from this, health officials and the government have continuously been advising the general public to avoid visiting the hospitals except for emergencies, resulting in these fears being compounded.

The COVID-19 pandemic has drastically reduced the routine visits made by the patients which includes the routine care for pregnant women, elderly patients and timely immunisation of the infants. Everything today is moving towards digitalization. Also, a major challenge faced by many healthcare organisations is the provision of quality medical services at affordable costs which implies proper diagnosis of the patient. Most of the people lack means to consult a doctor specialised in the field that they are looking for. The product proposed by us focuses on the methods by which we can overcome these challenges.

The portal that we aim to develop would provide an e-consultation facility to the patients. The patients can sign up at the portal. After logging in the patients would observe there are various sections that cater to their needs. A section for predicting the disease that they are likely to suffer from where they can fill a form by providing their details along with the symptoms witnessed. Based on the result of the prediction, the patients can search for the specialised doctors and share the prediction results with the selected doctor. The patients would also be provided with the section where they can upload their previous medical reports, thereby keeping a track of their medical history which often gets misplaced due to the conventional way of storing the reports in files.

The doctors across the country can register themselves on our portal specifying the fields that they have been working in. Once logged in then each doctor will receive notifications regarding the patients who have approached them along with their details including their symptoms. After analysing the reports of the patient, the doctor can provide a confirmation of the appointment and if required they can even provide the patient with the prescription that needs to be followed till the appointment date. This will help overcome the challenge of increased drop rate in the regular patient visits and also help patients to consult the doctors in case of emergency situations by fixing an appointment without the need to visit the hospital regarding the same.

Moreover, this would reduce the chances of disease transmission and would also be convenient for both the patient and the doctor especially residing in high alert areas. This would also help to overcome the problem of deficiency of human resources in the health sector which is prevalent at several levels such as between regions, between rural and urban areas and between private and public sectors. This challenge is solved by providing the feature that doctors throughout the country can sign up at our portal thus providing the benefit of consulting the health care specialists in the respective fields and bridging the gap between different sections of the society.

**1.5 References**

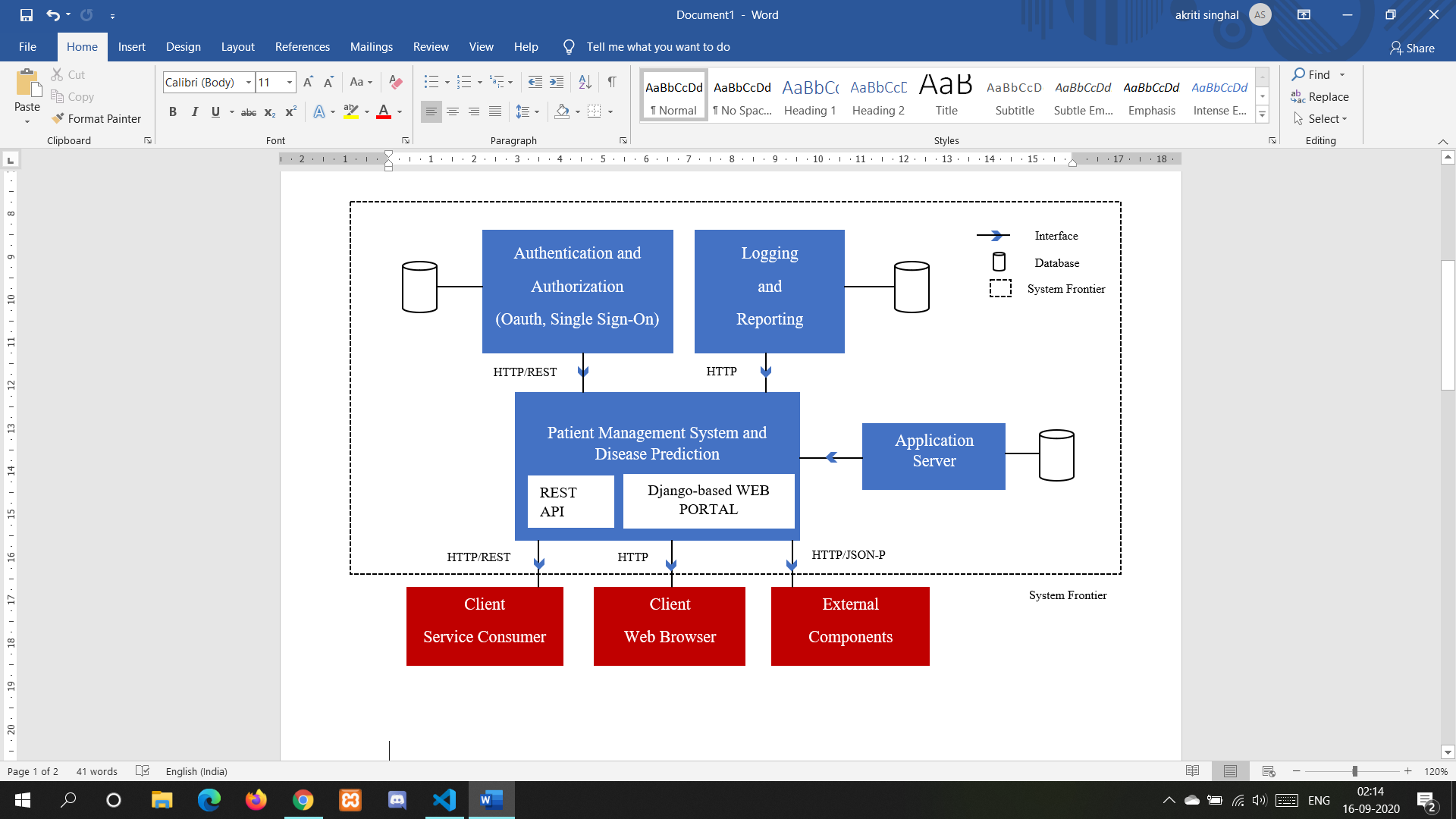
1. <https://krazytech.com/projects/sample-software-requirements-specificationsrs-report-airline-database>
2. <https://www.c-sharpcorner.com/article/software-requirements-specification/>
3. <https://belitsoft.com/custom-elearning-development/srs-for-e-learning-management-system>
4. <https://medium.com/@protoio/top-7-design-constraints-to-learn-how-to-work-around-8a67132e90b6>
5. [https://www.academia.edu/28734143/HOSPITAL\_MANAGEMENT\_SYSTEM\_Software\_Requirement\_Specificatio](https://www.academia.edu/28734143/HOSPITAL_MANAGEMENT_SYSTEM_Software_Requirement_Specification)

**2. Overall Description**

**2.1 Product Perspective**

The project can be considered a small scale version of hospital management system and can be further integrated with it as per the requirements of the hospitals. The main objective behind our vision of developing this product are the challenges faced during this pandemic. Along with this it will also increase the efficiency of the hospitals and bring the specialists from the nooks and corners of the country available at a single platform which can be easily accessed by any patient just by registering themselves on the web portal. It has been designed with regard to the stakeholder’s (here, the patients and the doctors) interests and goals. This project gives a procedural approach of how a patient can get an analysis of the disease he/she may be suffering from, how the patient can approach a doctor, get an appointment confirmation date or what precautions the doctor suggest him/her to follow in order to recover from the disease that he/she is likely to suffer from.

The usage of the database to store patient’s and doctor’s details will accommodate easy access,retrieval,search and manipulation of the data. The doctor and the patient will only be able to view their own profile and receive the concerned notifications . However, the admin can review the records of doctors as well as patients thus, managing the database. These access limitations enhance the security of the system.



**2.2 Product Features**

**Registration/Login:** The patients and the doctors need to register on the portal before they can start accessing the functionalities which are available to them. After login, they will be assigned a token to make sure that they are authenticated users.

**Disease prediction and Doctor suggestion :** Patients fill their symptoms in a form and according to that the system predicts the disease through a machine learning model.

The models are trained using various algorithms such as:

* 1. **Decision Tree algorithm** - It is used for classification as well as regression tasks.It is a flowchart-like structure where each internal node represents a test on the various symptoms entered by the user and each leaf node represents the class label i.e. the diseases.
  2. **Random Forest algorithm** - It is similar to the decision tree algorithm. The difference lies in the process of finding the root node(start point) and splitting the feature nodes here the symptoms randomly.The accuracy of the results lie in the number of trees formed.
  3. **Logistic Regression** - It is most suited for binary classification though it can be used for multinomial classification. It makes use of a logistic function i.e. softmax function to categorise the input data. It uses a linear model similar to linear regression to calculate the weights of each input feature and thereby producing a linear equation. The linear equation produced is then used to calculate logits which are scores. These scores are then used by a softmax function to calculate the probabilities. These probabilities are then used by cross-entropy function to calculate the distance from the target classes available. The one which results in the minimum distance is our predicted target class.
  4. **Naive Bayes algorithm** - It is a classifier which works on Bayes theorem.The prediction of membership probabilities is made for every class of disease such as the given set of symptoms is associated with a given disease (eg: fungal infection).

The result of each prediction is displayed on the screen. The disease class which is predicted by a majority of the models is the final result.

The patient can search for the concerned doctor within our database or any other doctor based on their location.

**Uploading medical reports:** Patients can keep a track of their medical history by uploading their reports along with the details such as previous appointments, doctor details etc on our platform which they can view anytime.This also provides an added benefit of forwarding the reports to the doctor if requested. Accessing the past reports and managing the present ones would be easier.

**Appointment Confirmation:** The doctor can set up an appointment with the patient for further consultation and diagnosis if he/she feels the need for the same. This will also replace the manual procedures of recording the appointment dates thereby increasing the reliability , efficiency and performance of the product.

**Prescription:** Doctors can send a prescription which can then be viewed by the patient. Depending upon the prediction, the doctor can decide whether to keep the further follow ups in an offline mode by fixing the appointment through the portal and asking the patient to follow the prescription till the next check-up or a course of the medicine for a certain period suggested is enough to recover from the disease.

**Edit profile:** The patients and the doctors can edit the information given at the time of registration by navigating to the ‘Edit profile’ tab and saving the changes which updates the information stored in the database.

**Question and Answer(QnA):** The patients and doctors are provided with a chat functionality for any active treatment. This facility helps the concerned patient and doctor to communicate with one another when it comes to fixing appointment dates or telling the precautions that the patient needs to take in case of emergency.

**2.3 User Classes and Characteristics**

1. Users of the project include the patients and the doctors.
2. Patients are the members who after logging in, would get a form to fill their witnessed symptoms.
3. Prediction of the disease would be carried out on the basis of the patient’s symptoms.
4. The patients would have an option to search for the concerned doctor from either our database or any doctor based on their location.
5. The patients can edit their details.
6. The patients can ask questions to the doctor with whom they have active treatments or reply to queries of doctors.
7. The patients would also be able to upload their old reports which they could access on a later date. They can also send these reports to the doctor if required.
8. The doctors are the users who would be contacted by the patients regarding an appointment.
9. The doctors will give proper medical advice to patients and can even prescribe medicines.They will also be able to access patient’s reports.
10. The doctors can edit their details.
11. The doctors can ask questions to the doctor with whom they have active treatments or reply to queries of patients.
12. Users should know how to use computers and access the internet.
13. Admin will get and analyze all details of patients and doctors.

**2.4 Operating Environment**

As this is a web based application so no such operating environment requirements are there. But as the web site relies on browsers and their capability to load data and handle requests so browsers with latest updates are preferred for best experience. Old and discontinued browsers such as microsoft internet explorer or older versions of opera are not preferred. Also as the screens are to be rendered in real time so a system with 2 gb of ram and processors powerful enough to support a web browser will be preferred for best intended experience .

**2.5 Design and Implementation Constraints**

1. The users need to register themselves on the web portal before they can access any functionality provided.
2. The database used is SQLite. Although multiple processes can access and query an SQLite database at the same time,only one process can make changes to the database at any given time. This means SQLite supports great concurrency however it doesn’t offer the level of concurrency as offered by client/server relational database management systems like MySQL or PostgreSQL.
3. The user would require a high bandwidth to access the application since being a serverless database, it doesn’t provide direct network access to its data.
4. SQLite can support a database upto 140TB in size, as long as the disk drive and file system also support the database’s size.
5. The entire database can be accessed by the admin by creating a superuser which provides higher privileges to access the data by the admin.
6. Predicting the disease on the basis of symptoms may take a little time since predicting using machine learning models is a CPU intensive process so the user might have to wait for a while before the results load.

**2.6 User Documentation**

This project is developed such that the total appearance of it is more user friendly. General users with basic

computer skills and knowledge of accessing the internet can use this software. This project will include an user manual.The user manual will include complete overview of system.configuration of tools used, technical details,backup procedures and also contact information which will include contact number,phone number and email address.

**2.7 Assumptions and Dependencies**

1. As we are using hosting from third party websites so the availability and speed of connection will depend upon the services provided by them.
2. As the api for prediction will be hosted online using third party web sites so its availability and request handling will depend upon the providers capability .
3. As the files are stored on other third party servers so the performance and storage capacity will depend on the service providers capability and the services they offer.

**3. System Features**

**3.1 Register User(Patient/Doctor)**

**3.1.1 Description and Priority**

The given feature will create an account for the user on our website using which their records can be stored on our database and they can then use all the intended features. For privacy reasons we cannot let users use the website without an account so their data is first required to be stored in our database so that they can use this application. As without it anything else cannot be used so this feature is of high priority.

**3.1.2 Stimulus/Response Sequences**

1. Users would click on the register button.
2. The user will be presented with a form in which they will fill all the required details.
3. The System will check the validity of the password and email followed by whether the data is valid or not
4. A user account will be created if all the details are valid.

**3.1.3 Functional Requirements**

1. **Navigation to registration page:**
   1. **Input -** The user needs to click on the register button.
   2. **Output and Errors -** The user is presented with a form having some fields
2. **User Details:**
   1. **Input -** The user inputs their details in all the required fields of the form.
   2. **Output and Errors -** The user will get a prompt of successfully registered if the details are valid and their account will be created. If a user misses any important field then the form will not submit and the system will prompt the user to fill that detail. If the details are invalid or the password doesn't match then the user will be prompted with an error message that the password is weak or the email is already registered depending on the condition.

**3.2 Login User(patient/doctor)**

**3.2.1 Description and Priority**

The given feature will help the user to login to their account. Using this, the data of the user is saved and also they can access all the functionality of the website For example after logging in the patient can upload their reports or get a prediction and view their details. Similarly after logging in, doctors can view all the patients they are currently dealing with and provide prescription or appointment. As this feature is required to access all other features so it has a high priority.

**3.2.2 Stimulus/Response Sequences**

1. User will Click on the login button.
2. The user will be presented with a form in which they will enter their email and password
3. The System will check the validity of the password and email followed by whether the data is valid or not.
4. If details are valid, Then the user will be logged in.

**3.2.3 Functional Requirements**

1. **Navigation to login page:**
2. **Input -** The user needs to click on the login button.
3. **Output and Errors -** The user is presented with a form having some fields.

**2. User Details:**

1. **Input -** The user inputs their email and password.
2. **Output and Errors -** The user will be logged in if the details are valid and their account details will be displayed and they can use the functionality of the website. If the password or email does not match to the records in the database then the user will be prompted to fill the form again with correct details or they will be given an option to register if they haven't done it already.

**3.3 Disease Prediction**

**3.3.1 Description and Priority**

The given feature will help the patient predict the disease he/she is likely to suffer from. The parameters used to carry out this prediction would include the various symptoms witnessed by the patient. This feature will help overcome the challenges faced by people who are looking online for health information regarding diseases, diagnosis and different treatments. Prediction of the disease being an important part of the treatment needs to be executed first before the patient can start using other functionalities offered by the system. Thus, it would have a high priority.

**3.3.2 Stimulus/Response Sequences**

1. After logging into the portal, the patient will have to navigate to the tab which requires the patient to fill in the details along with the symptoms.
2. The GUI of this section would provide fields for the patient to enter the name and age along with drop-down menus where he/she will have to select the appropriate symptom from the list so offered.
3. Based upon these symptoms, the patient will be provided three to four chances of prediction using various Machine Learning models.
4. The result of each prediction is displayed on the screen. The disease class which is predicted by a majority of the models is the final result.

**3.3.3 Functional Requirements**

1. **Navigation to the disease prediction section:**
2. **Input** - After logging in with the necessary credentials, the patient needs to navigate to the prediction corner where he/she can add the necessary details.
3. **Output and Errors** - This section displays a form where the patient will need to enter his name and age along with the various symptoms witnessed.If the user fails to navigate to the above section due to network issue (failure in loading the prediction section) then the user won’t be able to access this functionality thus, failing to access other functionalities as listed below.

**2. Symptoms Input:**

* 1. **Input** - The patient needs to select appropriate symptoms from the drop-down menus which will then be tested on the various machine learning models.
  2. **Output and Errors** - The symptoms enlisted will be sent to the backend where the model will be executed and the prediction results will be displayed on the frontend i.e. the screen he/she will be viewing. If the patient misses to fill even one of the symptoms then the result won’t be predicted since each field will be mandatory and play a vital role in the further processing. Thus, this will lead to invalid output.

**3.4 Appointment**

**3.4.1 Description and Priority**

After prediction of the disease, the patient can search for the concerned doctor. The patient would ask the doctor for an appointment and the doctor would provide the appointment confirmation details.

This feature would have a medium priority because this is not a compulsory feature. This would depend on the patient whether he/she wants to refer a doctor for the predicted disease or not.

**3.4.2 Stimulus/Response Sequences**

1. The patient would search for the concerned doctor and ask him/her for an appointment.
2. The doctor would send the appointment confirmation details.

**3.4.3 Functional Requirements**

1. **Navigation to the appointment section:**
   1. **Input -** After searching for the concerned doctor, the patient needs to navigate to the appointment option in the details section of the doctor that are provided.
   2. **Output and errors-** This section displays a button, which when clicked will allow Patients to ask the doctor for an appointment. If the user fails to navigate to the above section due to network issues then the user won’t be able to access this functionality.

**3.5 Upload reports**

**3.5.1 Description and Priority**

The given feature will help the patient to upload their medical reports. This feature will help to overcome the the challenges faced by the people to keep a record of all their previous reports as manually keeping record of all medical reports can be a cumbersome process for many people.There are some diseases which require constant track of previous medical reports as doctor needs to see whole medical background of that person.The best example of this is diabetes,as for prescribing any medicine, doctor also needs to know what were the results of the previous test. Therefore, developing a feature to store previous reports will make the life of users easy and comfortable.This feature is not compulsory for every user to access, it’s an optional feature provided to users for their comfort.Thus, it has the medium priority.

**3.5.2 Stimulus/Response Sequences**

1. After logging into the portal, the patient will have to navigate to the tab which requires the patient to upload his reports.
2. The GUI of this section would provide the patient to select from where he wants to upload his reports like from his device, from dropbox or from google drive.
3. Then the patient will select the particular report which he wants to upload.

**3.5.3 Functional Requirements**

1. **Navigation to the upload reports section:**
   1. **Input** - After logging in with the necessary credentials, the patient needs to navigate to the upload reports section where he/she can select the reports which they want to upload.
   2. **Output and Errors** - This section displays a button, which when clicked will allow the patient to choose from where he/she wants to upload the report.If the user fails to navigate to the above section due to network issues then the user won’t be able to access this functionality.

**3.6 View Patient details**

**3.6.1 Description and Priority**

The doctor after logging in would be able to see a list of patients who have asked for an appointment and the patients whom the doctor is currently attending to.

This feature would have a high priority as it is necessary for the doctor to be able to view the patient's details otherwise he would not be able to perform any further actions.

**3.6.2 Stimulus/Response Sequences**

1. The doctor would login and view the list of active patients.
2. The system would provide the doctor with the current patient details.

**3.6.3 Functional Requirements**

1. **Navigation to the upload reports section:**
   1. **Input** - After logging in with the necessary credentials, the doctor would have to navigate to the section containing the list of active patients.
   2. **Output and Errors** - This section displays a button, which when clicked will allow the doctor to view the list of patients he/she is currently attending .If the doctor fails to navigate to the above section due to network issues then he/she won’t be able to access this functionality.

**3.7 Prescription**

**3.7.1 Description and Priority**

The doctor, after confirming the appointment details would have an option to send the prescription to the patient. This would be done after the doctor has viewed the prediction report/ old reports of the patient based on which he has formed the prescription. The patient would be able to view the prescription once the doctor has sent it. This is an additional feature and the main functionality will not be affected by this so it has a medium priority.

**3.7.2 Stimulus/Response Sequences**

1. The doctor would view the prediction and old reports of the patient and send him a prescription for the same.
2. The patient would be able to view the prescription sent by the doctor.

**3.7.3 Functional Requirements**

1. **Navigation to the upload reports section:**
   1. **Input** - After logging in with the necessary credentials, the doctor needs to navigate to the upload prescription section where he/she can select the prescription which they want to upload.
   2. **Output and Errors** - This section displays a button, which when clicked will allow the doctor to choose from where he/she wants to upload the prescription .If the doctor fails to navigate to the above section due to network issues then he/she won’t be able to access this functionality.

**3.8 Edit profile**

**3.8.1 Description and Priority**

This feature provides both the patient and the doctor the ability to edit or update the information given at the time of registration. It also provides the option to change the password thus ensuring safety of the user in case privacy breach. The user can update the information as per the requirement. It is not necessary so this functionality has low priority.

**3.8.2 Stimulus/Response Sequences**

1. After logging into the portal, the user will have to navigate to the tab which requires the user to edit the details.
2. The GUI of this section would present the user with a form with fields containing the required information to be edited.
3. Then the user needs to update the information and click on the save button which saves the changes in the database.

**3.8.3 Functional Requirements**

1. **Navigation to the edit profile section:**
2. **Input** - After logging in with the necessary credentials, the user needs to navigate to the edit profile section where he/she can change the information.
3. **Output and Errors** - The user must have a stable internet connection otherwise he/she will not be able to connect to the portal and navigate to the required section. Second, after the user has finished updating the information he/she needs to click on the save button at the bottom of the form in order to save the changes. If the user forgets to click the button then the new changes will not be saved in the database and thus, old information will be reflected in the profile.

**3.9 Question and Answer**

**3.9.1 Description and Priority**

This feature helps the concerned patient and the doctor to communicate with one another through a small chat window provided at the end of any active treatment. The patient and doctor can decide a common date to fix an appointment by using this functionality since the recipient will be notified whenever a message is received from the intended user. The doctor can even suggest some precautions and the medicines that the patient must take through the chat window. Similarly, the patient can ask any doubt from the doctor.

**3.9.2 Stimulus/Response Sequences**

1. After logging into the portal, the user will have to navigate to the treatments tab which displays a list of new, active and closed treatments.
2. The user whether patient or doctor needs to select any active treatment to access this functionality
3. Then the user will see a chat window at the end of treatment details which provides the user with a field to write a message and then click on the send button.

**3.9.3 Functional Requirements**

1. **Navigation to the treatments section:**
   1. **Input** - After logging in with the necessary credentials, the patient needs to navigate to the treatments section and select an active treatment.
   2. **Output and Errors** - The user after writing the appropriate message needs to click on the send button in order to ensure that the message has been delivered to the intended recipient else the message won’t be received on the other end. Second, the user must have a stable internet connection. If there are any discrepancies in the network then the user won’t be able to send the message and get the accepted result.

**4. External Interface Requirements**

**4.1 User Interfaces**

1. **Patient interface:**

* The patient would see a register button upon clicking on which a registration form would open.
* After registration, the patient would have a Login button which on clicking would open a login form.
* After the authentication is done, the patient would be able to view all the options that our portal provides and the basic instructions.
* A tab for disease prediction would open on clicking the predict disease button.This would contain a form which would ask the patient to select the symptoms he/she is witnessing.
* A report would be generated by our system based on the symptoms.
* The Patient would have a button option to search for the concerned doctor .It would show a list of doctors whom the patient can consult by asking for an appointment. The patient would receive a notification once the doctor has confirmed the appointment.
* The patient can also go to the upload report tab where he/she can upload an old report/ prescription along with the necessary details like doctor’s name, date, disease name etc.
* This tab would also have an option to send a particular report to a doctor once the doctor has confirmed the appointment.
* A tab is provided to review the prescription sent by the doctor which the patient needs to follow up with until the next appointment.

1. **Doctor interface:**

* The doctor would see a register as a doctor button upon clicking on which a registration form would open(in case if he/she has not registered before).
* After registration, the doctor would have a Login button which on clicking would open a login form.
* After the authentication is done, the doctor would be able to view all the options that our portal provides and the basic instructions.
* A tab for active patient’s details would open on clicking the view details button.This would contain all the details of the active patient such as age,weight etc.
* There will also be another tab to review appointment requests of patients in which the doctor would have an option to schedule an appointment for the patient.
* The doctor can also prescribe patients appropriate medication based on the disease predicted. Also there will be an option to update or modify the prescriptions as in some cases with time dosage of medicine to be taken changes.

**4.2 Software Interfaces**

|  |  |  |
| --- | --- | --- |
| **S.no** | **Software Used** | **Description** |
| 1 | Operating System | Windows is used for its user friendly interface. |
| 2 | Python | Python is used for its vast libraries which help create the website and train the model. |
| 3 | Database | SQLite is the default database used by django.SQLite is included in Python, so you won't need to install anything else to support your database. |

**4.3 Communications Interfaces**

HTTPS Protocol is the standard protocol for communicating on the web and it will be used to access the website so that the information that is personal to the user is not intercepted by any attacker or any unauthorized personnel. HTTPS encrypts the data so that even if it is intercepted the attacker will still not be able to decipher any information.

POST requests are used to send form data so that the data is not directly visible so that the privacy of the user is maintained and no information is leaked.

**5. Other Nonfunctional Requirements**

**5.1 Performance Requirements**

1. The performance of the system should be fast and accurate.
2. Disease Prediction and Patient Management system should handle expected and non expected errors in ways that prevent loss in information and long downtime periods.
3. The system should be able to handle large amounts of data.Thus it should be able to handle large amounts of patients and doctors without any fault.

**5.2 Safety Requirements**

1. The database may get crashed or damaged due to some viruses or operating system requirements.Therefore it is mandatory to have backup for your data.
2. UPS/inverter facilities should be there in case of power failure.

**5.3 Security Requirements**

1. System will require a secured database.
2. Proper user authentication will be provided.
3. There will be seperate account for users and admin.So that no one else can access the database except admin.
4. When a user will perform any action, proper authorization will be made for him/her to perform allowed action and an error message will be displayed- Action is unauthorized.

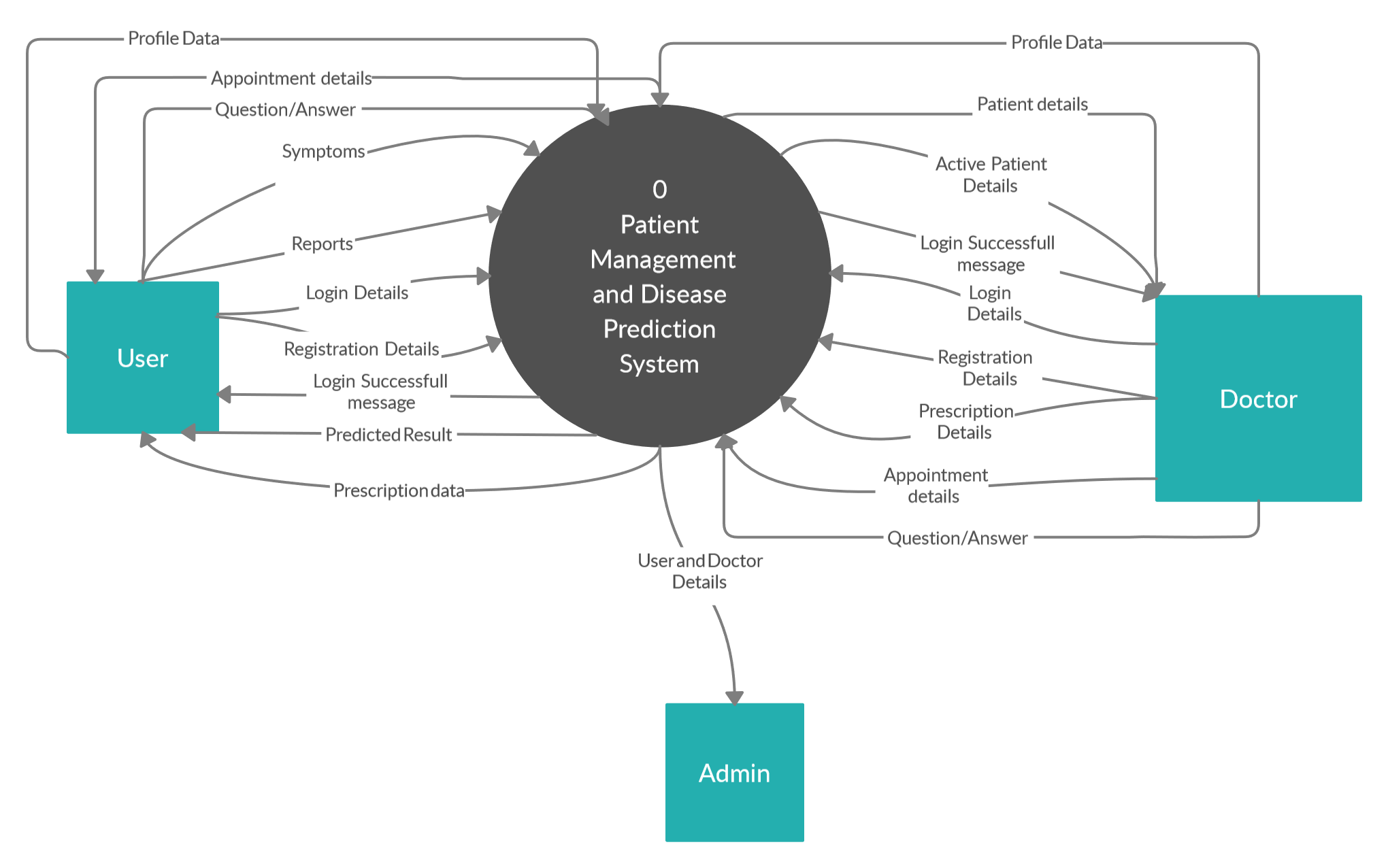
**5.4 Software Quality Attributes**

1. Good quality of framework produces robust,bug-free software which contains all necessary requirements for user satisfaction.

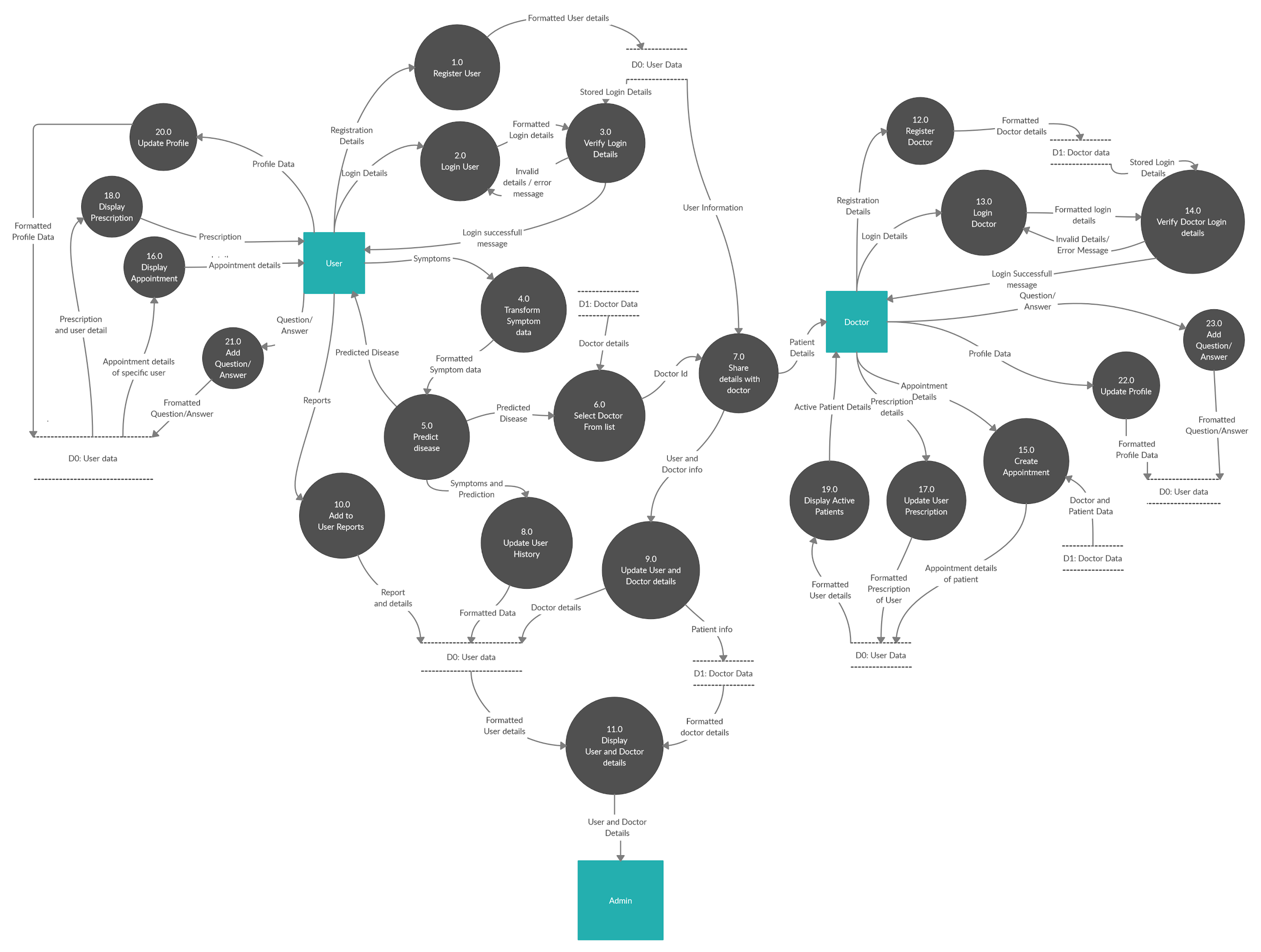
3. Structured Analysis

3.1 Data Flow Diagrams

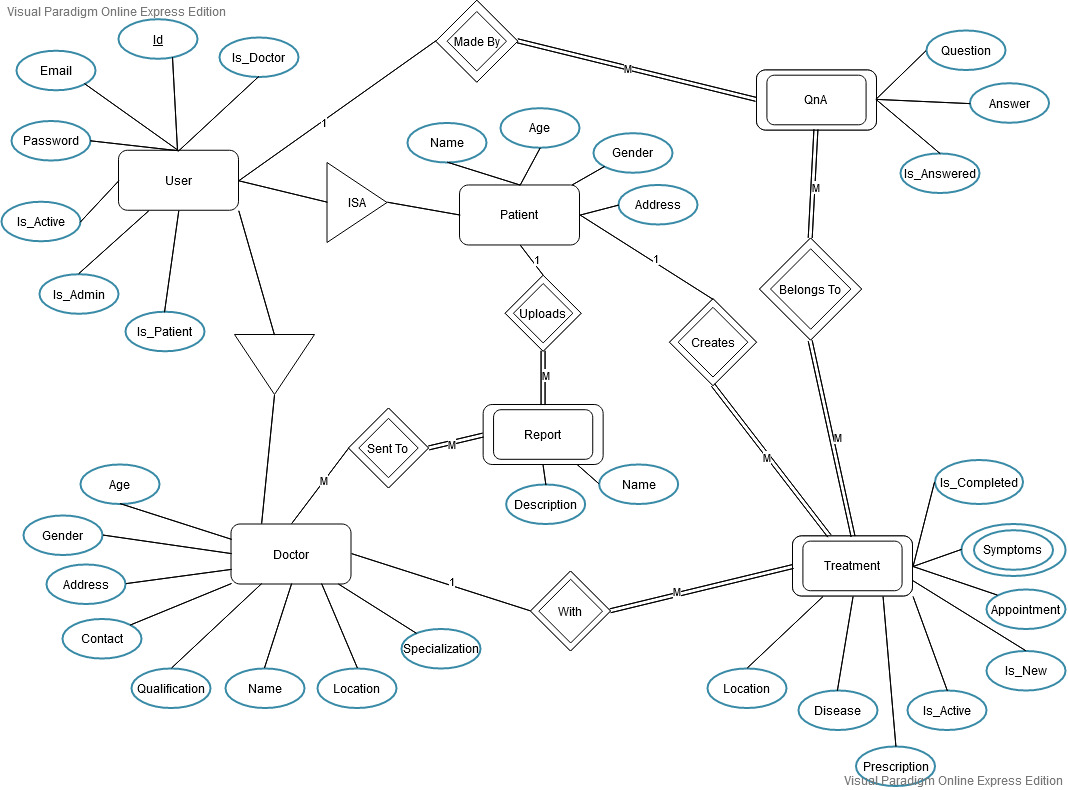
3.1.1 DFD Level 0



3.1.2 DFD Level 1

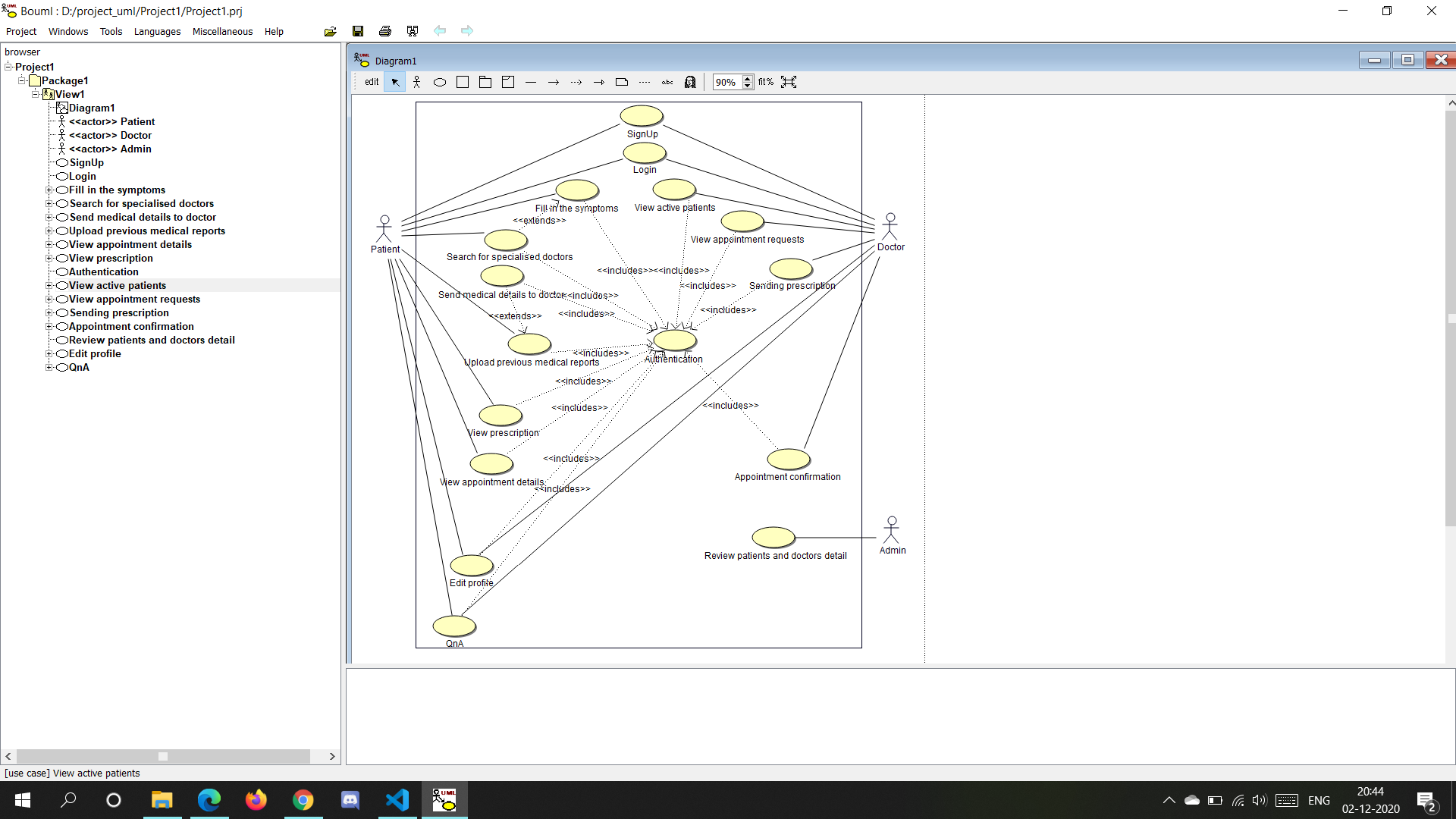


3.2 ER DIAGRAM:



4. Object Oriented Analysis

4.1 Use Case Diagram



4.2 Use Case Templates

|  |  |
| --- | --- |
| 1. Use Case Title | Sign-Up |
| 1. Abbreviated Title | Sign-Up |
| 1. Use Case Id | 1 |
| 1. Actors | 1.Doctor  2.Patient |
| 1. Description   This facility allows the doctors and the patients to register at the portal with their credentials being stored in the database. | |
| * 1. Pre-Condition   1.The user must have a stable internet connection so that they can have access to our portal. | |
| * 1. Task Sequence  1. Navigate to the portal window. 2. Click on the Register tab and enter the valid credentials. 3. Click on the register button. | |
| * 1. Post-Condition  1. The user will be prompted to validate your account through the email-id registered. 2. The user can then log in either as a patient or as a doctor depending upon the designation chosen at the time of registration. | |
| 1. Modification History   Date: 1-December-2020 | |
| 1. Author   Akriti, Saumyaa, Parth, Amish | |

|  |  |
| --- | --- |
| 1. Use Case Title | Login |
| 1. Abbreviated Title | Login |
| 1. Use Case Id | 2 |
| 1. Actors | 1.Doctor  2.Patient |
| 1. Description   This facility allows the doctors and the patients to login at the portal by entering the valid credentials provided at the time of registration. | |
| 5.1. Pre-Condition  1. The user must be registered on the portal.  2. The user must have verified the account through a validation mail sent on the  email-id used at the time of registration. | |
| * 1. Task Sequence  1. Navigate to the portal window. 2. Click on the Login tab and enter the valid credentials. 3. Click on the Login button. 4. In case the user forgets the password he/she can access the ‘forgot password’ functionality and change the password. | |
| * 1. Post-Condition  1. The user can access the various functionalities provided depending upon whether he/she is logged in as a patient or as a doctor. | |
| 1. Modification History   Date: 1-December-2020 | |
| 1. Author   Akriti, Saumyaa, Parth, Amish | |

|  |  |
| --- | --- |
| 1. Use Case Title | Authentication |
| 1. Abbreviated Title | Authentication |
| 1. Use Case Id | 3 |
| 1. Actors | Patient  Doctor |
| 5. Description  This facility ensures that the patients and the doctors who use the functionalities offered by the portal are  registered and have validated their credentials before accessing the portal. | |
| 5.1. Pre-Condition  1.The user must have a stable internet connection. | |
| 5.2. Task Sequence  1.The user navigates to the portal window.  2.Then, the user creates an account and validates the credentials through an email sent on the email id  used at the time of registration.  3.The user logs into the portal. | |
| 5.3. Post-Condition  1.The user is able to access the entire functionality provided depending whether he/she is logged in as a  patient or as a doctor. | |
| 6. Modification History  Date: 1-December-2020 | |
| 7. Author  Akriti, Saumyaa, Parth, Amish | |

|  |  |
| --- | --- |
| 1. Use Case Title | Edit profile |
| 1. Abbreviated Title | Edit profile |
| 1. Use Case Id | 4 |
| 1. Actors | Patient  Doctor |
| 1. Description   This functionality allows the user to edit his/her details i.e. update the profile information in case of any change. | |
| 5.1. Pre-Condition  1.The user must be logged into the portal, thus the user must be authenticated.  2.The user needs to navigate to the ‘Edit profile’ tab.  3.The user must have a stable internet connection. | |
| 5.2 Task Sequence  1.The user navigates to the ‘Edit profile’ tab.  2.Then, the user changes the required credentials such as name, address etc.  3.Click the save button to update the changes. | |
| * 1. Post-Condition   1.The changes get saved in the database. Thus, the user can operate the portal’s other functionalities with  the changed info. | |
| 1. Modification History   Date: 1-December-2020 | |
| 1. Author   Akriti, Saumyaa, Parth, Amish | |

|  |  |
| --- | --- |
| 1. Use Case Title | Fill in the symptoms |
| 1. Abbreviated Title | Fill in the symptoms |
| 1. Use Case Id | 5 |
| 1. Actors | Patient |
| 1. Description   This facility helps the patient to get a prediction of the disease he/she is likely to suffer from on  the basis of the symptoms entered. | |
| * 1. Pre-Condition   1.The patient must be logged into the portal, thus the patient must be authenticated.  2.The patient needs to navigate to the ‘Predictions’ tab.  3.The patient must have a stable internet connection. | |
| * 1. Task Sequence   1.The patient navigates to the ‘Create New’ tab.  2.The patient is then required to choose at most 5 symptoms from the drop-down menu that  appears and then clicks on the predict button. | |
| * 1. Post-Condition   1.The patient is then directed to another page where the results of the prediction are shown.  2.The patient is also prompted to select a doctor if he/she wishes to consult one and send an  appointment request. | |
| 1. Modification History   Date: 1-December-2020 | |
| 1. Author   Akriti, Saumyaa, Parth, Amish | |

|  |  |
| --- | --- |
| 1. Use Case Title | Search for specialised doctors |
| 1. Abbreviated Title | Search for specialised doctors |
| 1. Use Case Id | 6 |
| 1. Actors | Patient |
| 1. Description   This facility helps the patient to select a nearby doctor who is a specialist of the disease that the patient is likely to suffer from and send an appointment request to the one selected. | |
| 5.1. Pre-Condition  1.The patient must be logged into the portal, thus the patient must be authenticated.  2.The patient must have filled in the symptoms and received a prediction of the disease. | |
| 5.2. Task Sequence  1.After prediction, the patient is directed to another page where a list of specialists concerned will appear.  2.The patient can select any doctor.  3.A form will appear where the patient can upload any previous medical reports to be shared with the  doctor(optional) and send an appointment request. | |
| 5.3. Post-Condition  1.Once the patient sends the request he/she can communicate with the doctor through a little chat  window at the bottom and decide an appointment date. | |
| 1. Modification History   Date: 1-December-2020 | |
| 1. Author   Akriti, Saumyaa, Parth, Amish | |

|  |  |
| --- | --- |
| 1. Use Case Title | Upload previous medical reports |
| 1. Abbreviated Title | Upload previous medical reports |
| 1. Use Case Id | 7 |
| 1. Actors | Patient |
| 1. Description   This facility helps the patient to upload the previous medical reports to keep a track to the medical history  and if required can share these documents with the doctor with whom he/she has an ongoing treatment. | |
| 5.1. Pre-Condition  1.The patient must be logged into the portal, thus the patient must be authenticated.  2.The patient needs to navigate to the ‘Reports’ tab.  3.The patient must have a stable internet connection. | |
| 5.2. Task Sequence  1.The patient navigates to the ‘Reports’ tab.  2.A form will appear where the patient can fill in the title and the description of the report if required and  then upload a report stored in the local device.  3.Click the upload option. | |
| 5.3. Post-Condition  1.A list of the doctors with whom the patient has a treatment currently in progress will be displayed. If  needed the patient can select a doctor and share the required report. | |
| 6. Modification History  Date: 1-December-2020 | |
| 7. Author  Akriti, Saumyaa, Parth, Amish | |

|  |  |
| --- | --- |
| 1. Use Case Title | Send medical details to doctor |
| 1. Abbreviated Title | Send medical details to doctor |
| 1. Use Case Id | 8 |
| 1. Actors | Patient |
| 5. Description  This facility allows the patient to send the medical reports uploaded to the concerned doctor if required. | |
| 5.1. Pre-Condition  1.The patient must be logged into the portal, thus the patient must be authenticated.  2.The patient needs to navigate to the ‘Reports’ tab.  3.The patient must have a stable internet connection. | |
| 5.2. Task Sequence  1.The patient navigates to the ‘Reports’ tab.  2.After uploading the report, the patient will be prompted to select the doctor to whom he/she needs to  send the report.  3.Click on the send button. | |
| 5.3. Post-Condition  1.The report is sent and the doctor is notified regarding the same. | |
| 6. Modification History  Date: 1-December-2020 | |
| 7. Author  Akriti, Saumyaa, Parth, Amish | |

|  |  |
| --- | --- |
| 1. Use Case Title | View appointment details |
| 1. Abbreviated Title | View appointment details |
| 1. Use Case Id | 9 |
| 1. Actors | Patient |
| 1. Description   This functionality allows the patient to view appointment details once confirmed from the doctor. | |
| * 1. Pre-Condition   1. The patient must be logged into the portal, thus the patient must be authenticated.  2. The patient needs to navigate to the Treatment details tab.  3. The patient must have a stable internet connection. | |
| * 1. Task Sequence  1. Navigate to the portal window. 2. Click on the Treatment details tab. | |
| * 1. Post-Condition  1. A list of all the appointments fixed so far is displayed. 2. The patient can click on any treatment to view the appointment date and other details such as the medicines or precautions prescribed. | |
| 1. Modification History   Date: 1-December-2020 | |
| 1. Author   Akriti, Saumyaa, Parth, Amish | |

|  |  |
| --- | --- |
| 1. Use Case Title | View prescription |
| 1. Abbreviated Title | View prescription |
| 1. Use Case Id | 10 |
| 1. Actors | Patient |
| 1. Description   This facility helps the patient to view the treatment details along with the prescription if provided by the concerned doctor. | |
| 5.1. Pre-Condition  1.The patient must be logged into the portal, thus the patient must be authenticated.  2.The patient needs to navigate to the ‘Treatments’ tab.  3.The patient must have a stable internet connection. | |
| 5.2. Task Sequence  1.The patient navigates to the ‘Treatments’ tab.  2.In case of no active treatments, a prompt saying ‘No treatments yet’ would appear.  3.Else a list of active treatments will be displayed such that the patient can get detailed information  regarding the particular treatment including prescription if provided by the doctor. | |
| 5.3. Post-Condition  1.The patient can get the details of all the treatments required by just clicking on the particular treatment. | |
| 6. Modification History  Date: 1-December-2020 | |
| 7. Author  Akriti, Saumyaa, Parth, Amish | |

|  |  |
| --- | --- |
| 1. Use Case Title | View active patients |
| 1. Abbreviated Title | View active patients |
| 1. Use Case Id | 11 |
| 1. Actors | Doctor |
| 1. Description   This facility helps the doctor to view the list of the patients who have a treatment under process along  with the details required. | |
| 5.1. Pre-Condition  1.The doctor must be logged into the portal, thus the doctor must be authenticated.  2.The doctor needs to navigate to the ‘Active Patients’ tab.  3.The doctor must have a stable internet connection. | |
| 5.2. Task Sequence  1.The doctor navigates to the ‘Active Patients’ tab.  2.Under the active treatments heading, if there is any ongoing treatment then its details along with the  patient’s details will be shown. | |
| 5.3. Post-Condition  1.The doctor has the option to close the treatment once its completed. The treatment closed will appear  under the closed treatments heading and will be removed from the active treatments heading. | |
| 6. Modification History  Date: 1-December-2020 | |
| 7. Author  Akriti, Saumyaa, Parth, Amish | |

|  |  |
| --- | --- |
| 1. Use Case Title | View Appointment requests |
| 1. Abbreviated Title | View Appointment requests |
| 1. Use Case Id | 12 |
| 1. Actors | Doctor |
| 1. Description   This functionality allows doctors to view pending appointment requests and give confirmation. | |
| * 1. Pre-Condition   1.The doctor must be logged into the portal, thus the doctor must be authenticated.  2.The doctor needs to navigate to the NewRequests tab.  3.The doctor must have a stable internet connection. | |
| * 1. Task Sequence  1. Navigate to the portal window. 2. Click on the NewRequests tab. | |
| * 1. Post-Condition  1. The doctor is able to view the list of appointment requests. | |
| 1. Modification History   Date: 1-December-2020 | |
| 1. Author   Akriti, Saumyaa, Parth, Amish | |

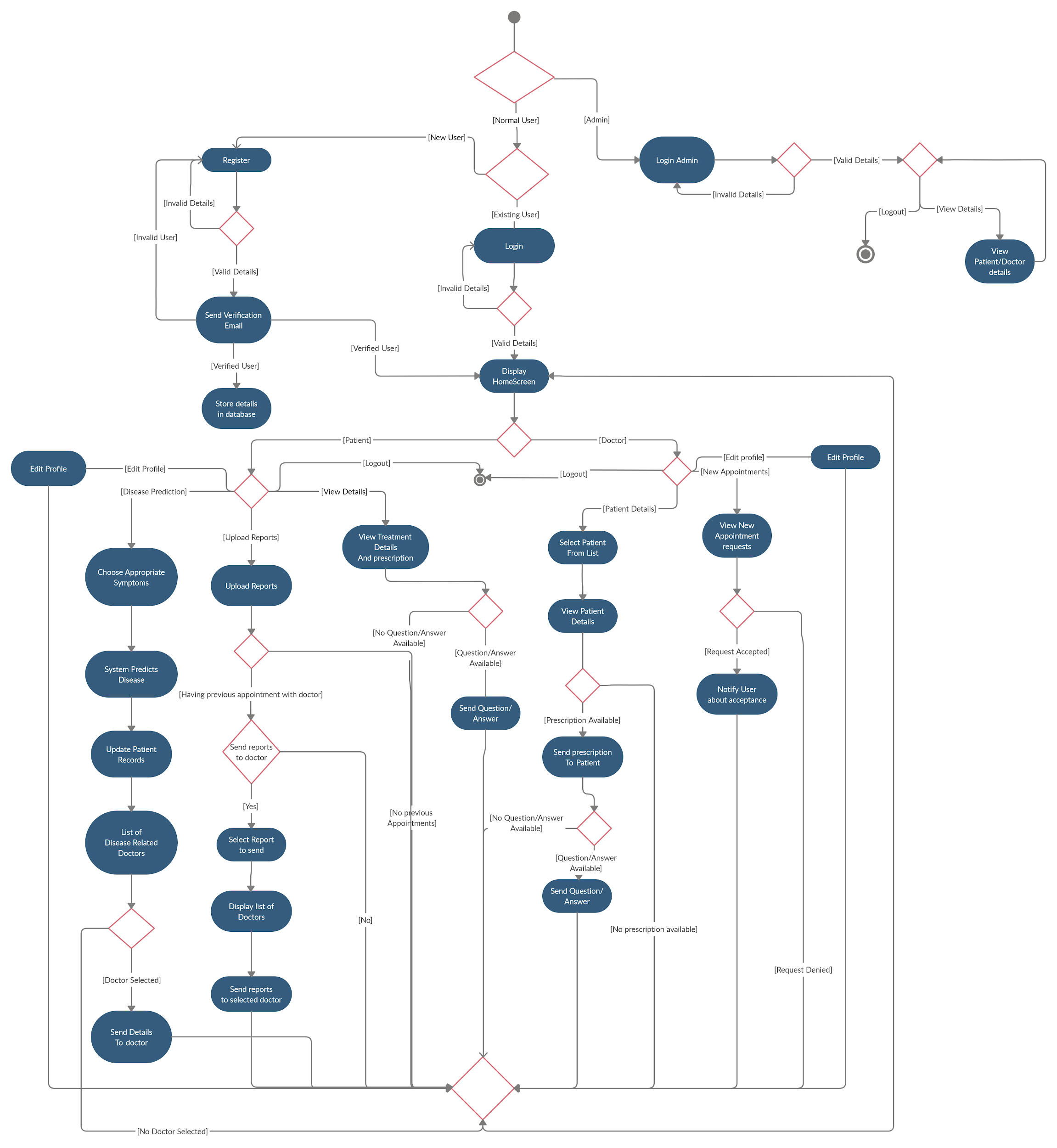
|  |  |
| --- | --- |
| 1. Use Case Title | QnA |
| 1. Abbreviated Title | QnA |
| 1. Use Case Id | 13 |
| 1. Actors | Patient  Doctor |
| 1. Description   This functionality allows the patient and the doctor to communicate with one another and ask relevant questions in case of any emergency. The users will be notified as and when they receive a message. | |
| 5.1. Pre-Condition  1.The user must be logged into the portal, thus the user must be authenticated.  2.The user needs to navigate to the ‘Treatments’ tab.  3.The user must have a stable internet connection. | |
| 5.2 Task Sequence  1.The user navigates to the ‘Treatments’ tab.  2.Clicks on any active treatment. A small chat window appears at the bottom of the page.  3.The user types a message and clicks on the send button. | |
| 5.3. Post-Condition  1.The message is delivered to the concerned user and a reply from him/her is directed to the recipient. | |
| 1. Modification History   Date: 1-December-2020 | |
| 1. Author   Akriti, Saumyaa, Parth, Amish | |

|  |  |
| --- | --- |
| 1. Use Case Title | Send Prescription |
| 1. Abbreviated Title | Send Prescription |
| 1. Use Case Id | 14 |
| 1. Actors | Doctor |
| 1. Description   This facility allows the doctor to send prescriptions to a patient. | |
| * 1. Pre-Condition   1.The doctor must be logged into the portal, thus the doctor must be authenticated.  2.The doctor needs to navigate to the Active Patients tab.  3.The doctor must select a patient.  4.The doctor must have a stable internet connection. | |
| * 1. Task Sequence  1. Navigate to the portal window. 2. Click on the Active Patients tab and select the patient.. 3. Select the prescription and click on send. | |
| * 1. Post-Condition  1. The prescription would be sent to the patient. | |
| 1. Modification History   Date: 1-December-2020 | |
| 1. Author   Akriti, Saumyaa, Parth, Amish | |

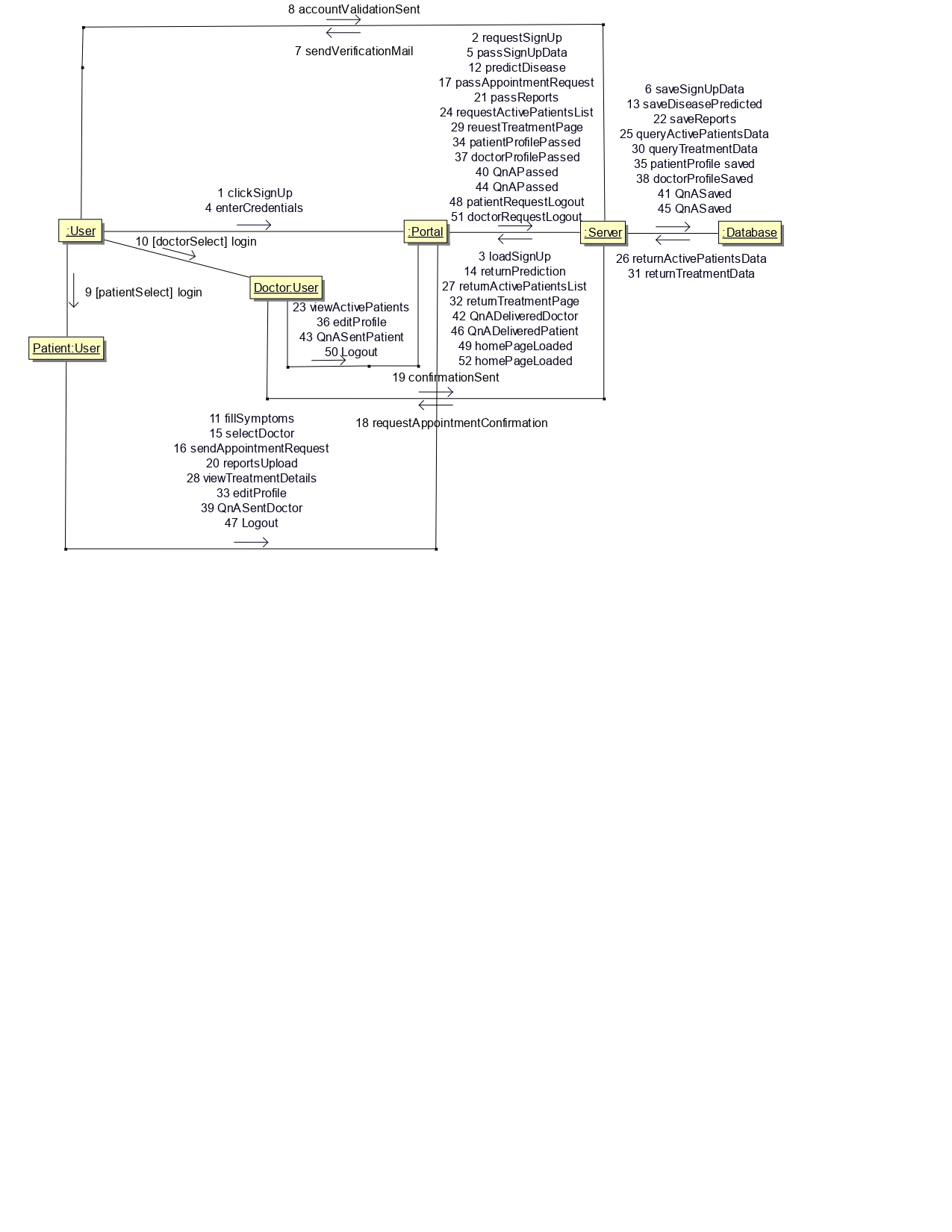
|  |  |
| --- | --- |
| 1. Use Case Title | Appointment confirmation |
| 1. Abbreviated Title | Appointment confirmation |
| 1. Use Case Id | 15 |
| 1. Actors | Doctor |
| 1. Description   This functionality allows doctors to confirm appointment requests. | |
| * 1. Pre-Condition   1.The doctor must be logged into the portal, thus the doctor must be authenticated.  2.The doctor needs to navigate to the NewRequests tab.  3.The doctor must have a stable internet connection. | |
| * 1. Task Sequence  1. Navigate to the portal window. 2. Click on the NewRequests tab. 3. Click on a particular request. 4. The doctor will have a small chat window appearing at the bottom of the screen where he can communicate with the patient and fix an appointment date. 5. The doctor fills in the final date and then clicks the confirm button. | |
| * 1. Post-Condition  1. The appointment request is confirmed and the patient is notified regarding the same. | |
| 1. Modification History   Date: 1-December-2020 | |
| 1. Author   Akriti, Saumyaa, Parth, Amish | |

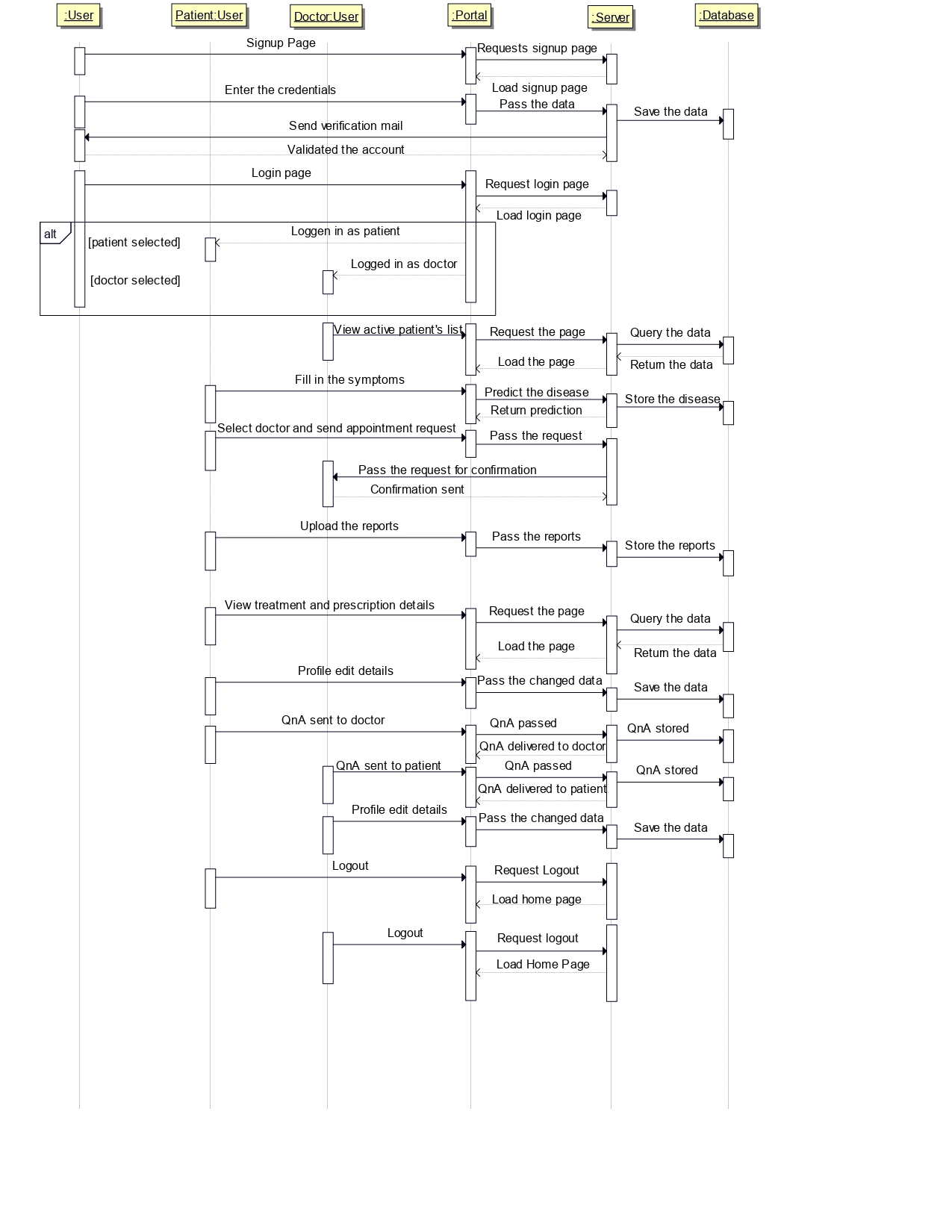
|  |  |
| --- | --- |
| 1. Use Case Title | Review patient’s and doctor’s details |
| 1. Abbreviated Title | Review patient’s and doctor’s details |
| 1. Use Case Id | 16 |
| 1. Actors | Admin |
| 1. Description   This functionality allows the admin to view all the details of the patients and the doctors who have registered on the portal and the functionality they have used. | |
| * 1. Pre-Condition   1.The admin must have a stable internet connection.  2.The admin must be logged into the admin page. | |
| * 1. Task Sequence  1. The admin navigates through the panel and can view the detailed tables which hold the record of   patients and the doctors. | |
| * 1. Post-Condition  1. The admin can create, edit or delete any entry in the database. | |
| 1. Modification History   Date: 1-December-2020 | |
| 1. Author   Akriti, Saumyaa, Parth, Amish | |

4.3 Activity Diagram

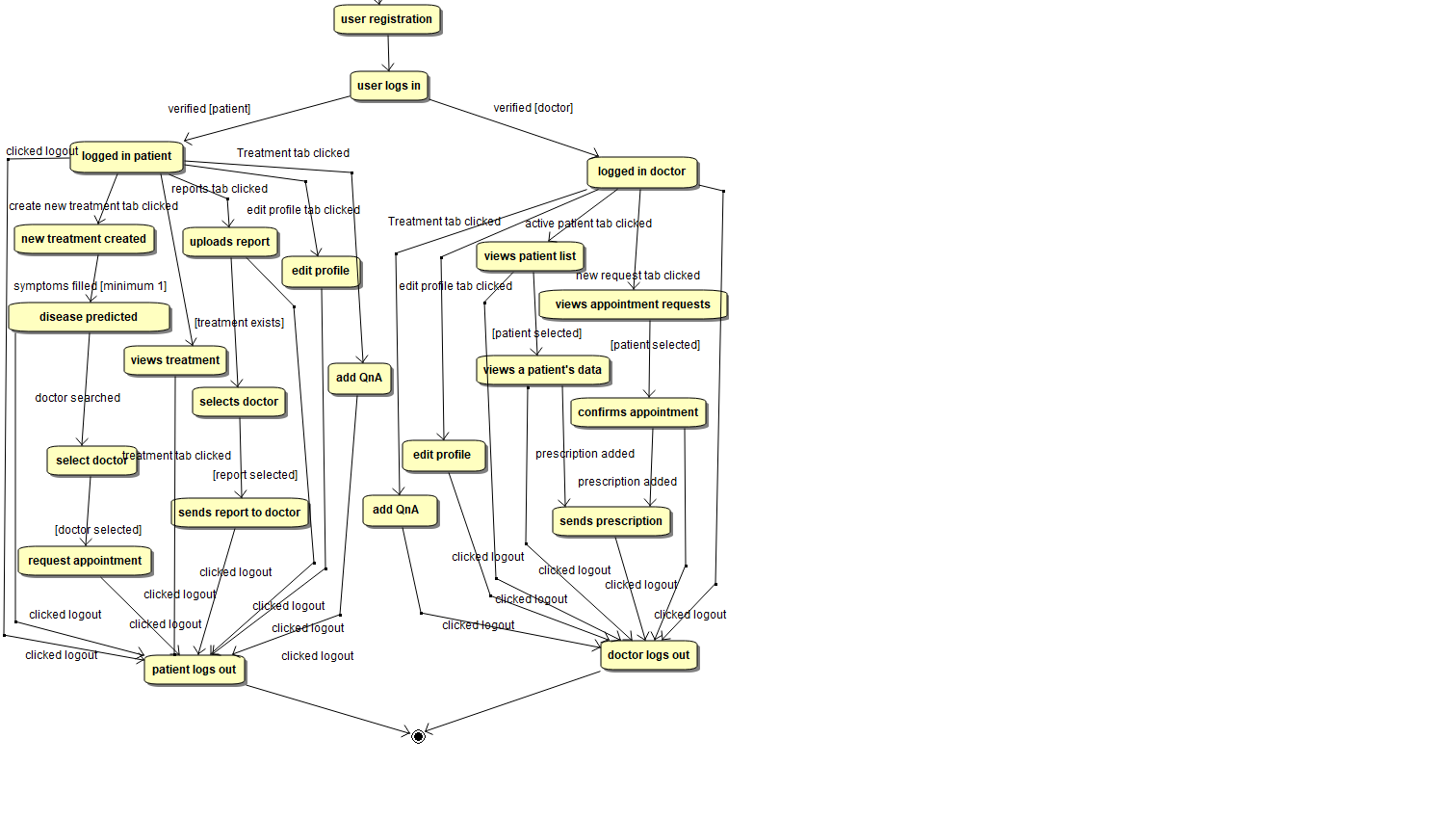


4.4 Collaboration Diagram



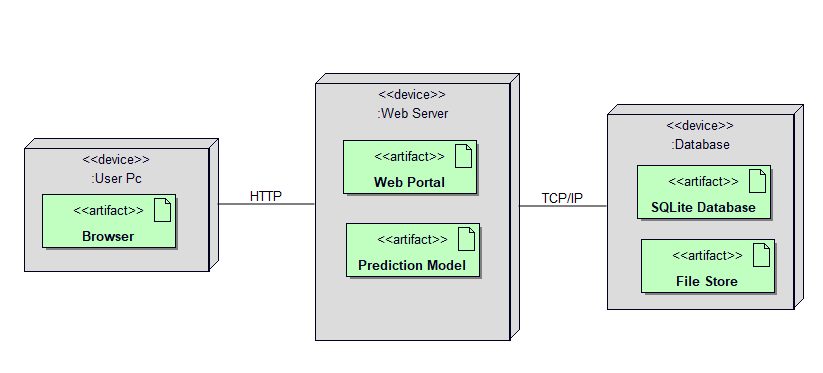
4.5 SEQUENCE DIAGRAM:  


4.6 STATE CHART DIAGRAM:



4.7 Component Diagram

4.8 Deployment Diagram



5. Testing

Unit testing is the first level of testing and is often performed by the developers themselves. It is the process of ensuring individual components of a piece of software at the code level are functional and work as they were designed to. Developers in a test-driven environment will typically write and run the tests prior to the software or feature being passed over to the test team. Unit testing can be conducted manually, but automating the process will speed up delivery cycles and expand test coverage. Unit testing will also make debugging easier because finding issues earlier means they take less time to fix than if they were discovered later in the testing process. TestLeft is a tool that allows advanced testers and developers to shift left with the fastest test automation tool embedded in any IDE.

5.1. Test Case Report

**Test Case Template** (Doc:T\_01)

**Pre-conditions-**

The user is a valid user- The user logged in with verified credentials.

The current password is hq#qwert1.

**Test Case #: 5.2**

**System: Web Portal**

**Designed by: Akriti**

**Executed by: Saumyaa**

**Short Description : Test system’s Change password service**

**Test Case Name : Change password**

**Subsystem: Profile**

**Design Date: 28/11/2020**

**Execution Date: 30/11/2020**

**Page: 1 of 1.**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Action** | **Expected System Response** | **Pass/ Fail** | **Comment** |
| 1 | Click the Change Password button | System displays a form containing a new password field. | Pass | Displayed correctly |
| 2 | Enter new password- 1234  Re-enter password- 1234 and click submit | System asks to enter a stronger password | Pass | Systems shows correct error |
| 3 | Enter new password- hbdsc#25  Re-enter password- hbdsc#25 and click submit | System shows success message | Pass | Systems shows correct message |
|  | Check post-condition 1 |  |  |  |
| 4 | Repeat steps 1,2,3 using another password ‘bab#34#’  Re-enter password ‘bbb#34#’ and click submit | System shows message- ‘Passwords do not match’ | Pass | Systems shows correct error |
| 5 | enter password ‘bbb#34#’ and click submit  Re-enter password ‘bab#34#’ and click submit | System shows success message | Pass | Systems shows correct message |

**Post-conditions**

Newpasswordhbdsc#25 saved in the database

Newpasswordbab#34#saved in the database

**Test Case Template** (Doc:T\_02)

**Test Case #: 1.1**

**System: Web Portal**

**Designed by: Amish**

**Executed by: Parth**

**Short Description: Test system’s User Registration service**

**Test Case Name : User Registration**

**Subsystem: Login**

**Design Date: 27/11/2020**

**Execution Date: 29/11/2020**

**Page: 1 of 1.**

**Pre-conditions-**

The user is not registered.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Action** | **Expected System Response** | **Pass/ Fail** | **Comment** |
| 1 | Click the Register button. | System displays a form containing different fields required for registration. | Pass | Form displayed  correctly |
| 2 | Enter email id as “[abc@gmail.com](mailto:abc@gmail.com)” Enter password as “daisy@1234”  After filling all other details press the Register button. | System will load a new page which will display a message asking users to validate their account by clicking on the link sent on the registered email id. After clicking on the link ,users will get registered. | Pass | Page redirected  successfully |
|  | **Check the post condition** |  |  |  |
| 3 | Repeat step 1 and then enter email as “abcd” and then fill all other details and then click the register button. | System asks to enter valid email id. | Pass | Systems shows  correct error |
| 4 | Repeat step 1 and then enter email id as “[abcd@gmail.com](mailto:abc@gmail.com)”. and then fill other details and then click the Register button. | System will load a new page which will display a message asking users to validate their account by clicking on the link sent on the registered email id. After clicking on the link ,users will get registered. | Pass | Page redirected  successfully |
|  | **Check the post condition** |  |  |  |
| 5 | Repeat step 1 and then enter email id as “[abcd@gmial.com](mailto:abc@gmial.com)” and password as “ram”and after filling all the other details, click the register button | System pops an error stating that password is too common and short.The minimum length should be of 8 characters. | Pass | Systems shows  correct error |
| 6 | Repeat step 1 and then enter email id as “[abcd@gmial.com](mailto:abc@gmial.com)” and password as “daisy@1234”and after filling all the other details, click the register button. | System will load a new page which will display a message asking users to validate their account by clicking on the link sent on the registered email id. After clicking on the link ,users will get registered. | Pass | Page redirected  successfully |

**Post-conditions**

User is registered successfully after clicking the link recieved on the registered email id.

**Test Case Template** (Doc:T\_03)

**Test Case #: 1.2**

**System: Web Portal**

**Designed by: Saumyaa**

**Executed by: Akriti**

**Short Description: Test system’s login functionality**

**Test Case Name : Login**

**Subsystem: Login**

**Design Date: 26/11/2020**

**Execution Date: 29/11/2020**

**Page: 1 of 2.**

**Pre-conditions-**

1.The user has registered on the portal and verified his/her credentials through a validation mail sent on the registered account.

2.The email for the current user is harry@gmail.com .

3. The password is potter123.

4. Harry is a patient.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Action** | **Expected System Response** | **Pass/ Fail** | **Comment** |
| 1 | Click the login tab | System displays a form with email and password fields. | Pass | Form displayed  correctly |
| 2 | Enter the email ‘[harry@gmail.com](mailto:harry@gmail.com)’  Enter the password ‘potter123’  Click the login button | System directs the user to the home page for the patient. | Pass | Page redirected  successfully |
|  | **Check the post condition** |  |  |  |
| 3 | Repeat steps 1,2,3 using email ‘[harry1@gmail.com](mailto:harry1@gmail.com)’ and password ‘potter123’ and click on the login button. | System displays an error message prompting to enter valid credentials. | Pass | Systems shows  correct error |
| 4 | Re-enter the email-id ‘[harry@gmail.com](mailto:harry@gmail.com)’ and click on the login button. | System directs the user to the home page for the patient. | Pass | Page redirected  successfully |
|  | **Check the post condition** |  |  |  |
| 5 | Repeat steps 1,2,3 using email ‘[harry@gmail.com](mailto:harry1@gmail.com)’ and password ‘potter1234’ and click on the login button. | System displays an error message prompting to enter valid credentials. | Pass | Systems shows  correct error |
| 6 | Re-enter the password ‘potter123’ and click on the login button. | System directs the user to the home page for the patient. | Pass | Page redirected  successfully |
|  | **Check the post condition** |  |  |  |
| 7 | Repeat steps 1,2 with email ‘harrygmail.com’. | A prompt appears telling the user to enter an email address | Pass | Systems shows  correct error |
| 8 | Re-enter the email id ‘[harry@gmail.com](mailto:harry@gmail.com)’ and repeat steps 3 and 4. | System directs the user to the home page for the patient. | Pass | Page redirected  successfully |
|  | **Check the post condition** |  |  |  |

**Post-conditions**

The user is logged into the portal and thus, he/she can access the other functionalities depending on whether he/she is logged in as patient or doctor.

**Test Case Template** (Doc:T\_01)

**Test Case #: 2.1**

**System: Web Portal**

**Designed by: Amish**

**Executed by: Parth**

**Short Description: Test system’s upload report functionality**

**Test Case Name : Upload Report**

**Subsystem: Report**

**Design Date: 25/11/2020**

**Execution Date: 27/11/2020**

**Page: 1 of 1**

**Pre-conditions-**

1.The user is logged in with verified credentials.

2. The title of the report is ‘Diabetes’.

3. The description is ‘Report containing treatment details for diabetes’.

**Post-conditions**

The recently uploaded report is added to the reports section and saved in the database.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Action** | **Expected System Response** | **Pass/ Fail** | **Comment** |
| 1 | Click the ‘Reports’ tab. | System displays a form with title, description and upload fields. | Pass | System displays  For correctly |
| 2 | Enter the title ‘Diabetes’.  Fill All details  Select Report  Click on the upload button. | System redirects the patient to the same page with the updated reports. | Pass | System loads  Page correctly |
|  | **Check the post condition** |  |  |  |

**Test Case Template** (Doc:T\_01)

**Test Case #: 3.1**

**System: Web Portal**

**Designed by: Saumyaa**

**Executed by: Akriti**

**Short Description: Test system’s Prediction of Disease service**

**Test Case Name : Prediction of Disease**

**Subsystem: Prediction**

**Design Date: 27/11/2020**

**Execution Date: 30/11/2020**

**Page: 1 of 2.**

**Pre-conditions-**

The user is a valid user- The user logged in with verified credentials.

**Post-conditions**

Saved the predicted disease in the database

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Action** | **Expected System Response** | **Pass/ Fail** | **Comment** |
| 1 | Click the Create New tab | System displays a list of symptoms | Pass | System  Displays list  correctly |
| 2 | Select symptoms | System asks to select minimum one symptom | Pass | System |
| 3 | Select atleast one symptom | System shows the selected symptoms | Pass | System displays  Symptoms  correctly |
| 4 | Click on predict button | System shows the predicted disease | Pass | Systems loads  Page correctly |

**Test Case Template** (Doc:T\_01)

**Test Case #: 2.2**

**System: Web Portal**

**Designed by: Parth**

**Executed by: Amish**

**Short Description: Test system’s Send Report service**

**Test Case Name : Send Report**

**Subsystem: Reports**

**Design Date: 26/11/2020**

**Execution Date: 30/11/2020**

**Page: 1 of 2**

**Pre-conditions-**

The User is logged in as patient.

The User has Uploaded Report

The User has Active Treatments

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Action** | **Expected System Response** | **Pass/ Fail** | **Comment** |
| 1 | Click the Reports Tab. | System displays form and below that name of file with a checkbox is displayed. | Pass | System displays  Form correctly |
| 2 | Select the checkboxes of doctor | System displays a check on the corresponding name | Pass | System displays  check correctly |
| 3 | Click on already selected checkboxes to unsend file | System unticks those checkboxes. | Pass | System  Unchecked  correctly |
| 4 | Click the Send Button | System refreshes the page with the updates checkbox state. | Pass | System loads  Page correctly |

**Post-conditions**

The File is shared with selected doctors

**Test Case Template** (Doc:T\_01)

**Test Case #: 4.1**

**System: Web Portal**

**Designed by: Parth**

**Executed by: Amish**

**Short Description: Test system’s View active treatments service**

**Test Case Name : View active treatments**

**Subsystem: Treatments**

**Design Date: 24/11/2020**

**Execution Date: 26/11/2020**

**Page: 1 of 2.**

**Pre-conditions-**

The User is logged in..

The User has Active Treatments

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Action** | **Expected System Response** | **Pass/ Fail** | **Comment** |
| 1 | Click the Active Patients Tab. | System displays a list of all the active treatments currently with the doctor | Pass | System displays  List correctly |
| 2 | Select one treatment from list | System loads a new page with treatment details. | Pass | Systems loads  Page correctly |

**Post-conditions**

The User can view Treatment Details.

**Test Case Template** (Doc:T\_01)

**Test Case #: 5.1**

**System: Web Portal**

**Designed by: Akriti**

**Executed by: Saumyaa**

**Short Description: Test system’s Edit Profile service**

**Test Case Name : Edit Profile**

**Subsystem: Profile**

**Design Date: 27/11/2020**

**Execution Date: 30/11/2020**

**Page: 1 of 2.**

**Pre-conditions-**

The user is logged in.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Action** | **Expected System Response** | **Pass/ Fail** | **Comment** |
| 1 | Click the Edit Profile tab from the navigation bar. | System displays a form containing different fields to edit information. | Pass | Form displayed  correctly |
| 2 | Edit the details like the name as “olof”.  Click on the Save Changes button. | System will pop an alert showing “Profile Updated Successfully”. | Pass | System displays  Error correctly |
|  | **Check the post condition** |  |  |  |
|  | Repeat steps 1,2 and enter age as “abcdef” | System will not allow input of any type except numbers. | Pass | System displays  Error correctly |
|  | Enter age as “12”. | System will ask to enter all other details that the user wants to edit. | Pass | System displays  For correctly |
|  | Click the Save changes button | System will pop an alert showing “Profile Updated Successfully”. | Pass | System displays  Alert correctly |

**Post-conditions**

User’s profile is now updated with the new name as “olof” and age as 12.