**SOFTWARE REQURIMENT SPECIFICATION (SRS) FOR**

**E HEALTH CARE ADVISOR (EHA)**

**1. Introduction**

**1.1 Purpose:**

The main objective of this project is to implement a computer based Healthcare Information System. This system will help the users to identify certain diseases by answering certain questions asked by the system. Based on the diagnose received the user will be getting some suggestion of medicines that are available at the local chemist without prescription with an advice to visit the doctor. The system once ready should be able to train itself with the feedback given to it (Artificial Intelligence). The database will be developed with open source software.

E – Healthcare Advisor is a web application which provides online medical services to everyone at their doorstep. The users living in metro or remote village can connect through internet or approach nearby kiosk to get these services. This web application is more effective, quick in providing medical help especially to people in villages where very few doctors are present. This helps the patients to maintain a neat health record and to lead a healthy life.

* 1. **Scope:**

There are three basic users - Patient, Admin, Doctors .

1.All users have their own profiles in VMH.

2.The web-cam interaction between doctor and patient.

3.Patients can search for doctor and make online appointments. They also can view their health record, lab reports, doctor’s prescription and medical expenses. Patient can also register complaint on any doctor.

4. Doctor’s can give appointments, e-prescription and can view patient’s health record.

5. Admin has the authority to add/delete users, grant permission to doctors and kiosk manager, to generate and view reports. He also views the complaints of patients and takes necessary actions.

**1.3 Definitions, Acronyms and Abbreviations:**

**EHA**

E – Healthcare Advisor. It’s a web application that provides online medical services for people.

**Admin**

Administrator: He has the authority to add/delete users, grant permission to doctors .

* 1. **References:**

The SRS document uses the following references

UHCL: Information Security Requirement

To provide security to the system based on the current system currently used by UHCL.

Billing System: To provide the interface between the system being developed and billing system in use of UHCL.

**1.5 Overview:**

**Existing System:**

**1.** Registration for users

**2.** Discussion forum

E – Healthcare Advisor is a web application which provides online medical services to everyone at their doorstep. The users living in metro or remote village can connect through internet or approach nearby kiosk to get these services. This web application is more effective, quick in providing medical help especially to people in villages where very few doctors are present. This helps the patients to maintain a neat health record and to lead a healthy life.

**2 . Over All Description**

**2.1 Product Perspective**

**Software Requirements**

1. Front end client on Internet

2. FF web Browser , Debian Os

3.Web server

4.Data Base Server

5.Development End

**Hardware requirements:**

1.**Client Side**

i processor with 800MHZ

ii RAM 128MHz

iii Disk space 900MB

iv Mozila firefox6

**2.Server Side**

i processor Intel Pentium

ii Ram 1500MB

**3. Communication Interface**

i using HTTP protocol

ii using HTTPS protocol

**2.2 Product Functions:**

|  |  |  |
| --- | --- | --- |
| Class of use cases | Use cases | Description of use cases |
| Use case related to  Installation | Installation | EHA installation |
| Use case related to  System Authorization | Login | Login into the account |
| Change password | Change the password |
| Use case related to  Registration | Registration | Register new patients |
| Use case related to  Search | Search | Search the patient is already present in the database or not |
| Use case related to create profile | View profile | Viewing details in profile. |
| Create profile | To create profile by entering details. |
| Use case related to E health record | Create health record | Containing health record |
| Update health record | Updating health record |
| Use case related to securities | Create security | Create a new security. |
| Delete security | Delete the existing security |
| Use case related to make an appointment | Seek an appointment | Get an appointment to interact with the doctor. |
| Use case related to delete  Users | Delete Users | Delete the users. |
| Use case related to disease database |  | Details of disease database |
| Use case related to prescription |  | Details of prescription of generic and non-generic medicines |

**2.3 User Characteristics:**

**Users**

1. Patients

2. Nonmembers

3. Administrator

**1. Patient**:

i Search for doctors

ii Get online appointments

iii View their record, lab reports

**2. Doctors:**

i Give appointment

ii E-prescription

iii Updated and view patients health record

**3. Admin:**

i Add and delete the users

ii Grant permissions to generate and view reports

iii View the complaints of patients and take necessary actions

**2.4 Principal Actors:**

1 The two principal actors in EHM are “user” and “system”.

1. EHA is a multi user software.

**2.5 General Constraints:**

1. For full working EHM requires Internet connection.

2. EHM is single-user software.

3.Only registered patients are authorized to use the services.

4. Limited to HTTP/HTTPS.

**2.6 Assumptions and Dependencies:**

1 .Full Working of E-health advisor dependent on the availability of internet connection.

2. The people should be aware of English language.

3. Should be aware of medical terms.

**3 Specific Requirements:**

**3.1 Functional Requirements:**

We describe the functional requirements by giving various use cases.

**3.1.1 Functional Requirement 1:**

Introduction: Installation

The E-health care advisor software can be worked through installation.

Input :

User name and home directory are given.

Processing :

User initiates the EHA installation program. System asks the user for the home directory in which all the working files will be created. User is also asked for initial login and password. User specifies the home directory and login, password .System creates the working files in the specified home directory.

Output:

The details of patient , doctor , administrator can be accessed.

**3.1.2 Functional Requirement 2:**

Introduction: Login

The login page is used for the user to access the details of doctor , patient , administrator.

Input :

Username , Password .

Processing :

Start the application .User prompted for the login and password. User gives login and password .System does the authentication .Main screen is displayed.

If authentication fails then the user is again asked to fill the details.

Output :

Main screen is displayed.

**3.1.3 Functional Requirement 3:**

Introduction: Change Password

This is required for the users the lost their password then the user can change the password with the help of this page.

Input:

Username , Password he remembers.

Processing :

User initiates the password change command. User is prompted for the old password ,New password and confirm password. User gives the all the passwords asked then the system does authentication .New Password is registered with the system .User gets a message to his mail that password was changed.

Output:

Password changed successfully.

**3.1.4 Functional Requirement 4:**

Introduction :Profile

Input :

First name, Last name , Occupation ,age ,email-id , phone number , height ,weight , gender , blood-group , dob , address are received.

Processing :

After inputting data validation checks on various fields is performed .On submission of the information the profile is created and stored in the database .On the basis of information supplied.

Output :

Profile for the database user is created and stored in database.

**3.1.5 Functional Requirement 5:**

Introduction: Search

Input :

Patient /Doctors must be logged in using username and password.

Processing :

Search the patient /doctor is already present in the database or not. After login , the user clicks on the search box. System asks the user for the patient/doctor and clicks on the search button.

Output :

If the searched patient/doctor is present in the database , information related to the patient /doctor will be displayed .If not present in the database it displays on error message called “not found” on the screen.

**3.1.6 Functional Requirement 6:**

Introduction : Seek an appointment

Intput :

Patient must be logged in , by using username and password .

Processing :

User clicks on to appointment page. System asks the name of the hospital, name of the doctor , timings and phone number. Patient enter the details and click on the submit button.

Output :

An appointment is given by sending a mail to patient that appointment is given.

**3.1.7 Functional Requirement 7:**

Introduction :Medicines

Input : Doctor must be logged in , by using username and password .Patient age , name , and phone number , are also the inputs.

Processing :

Patients have to fill the questions given doctor. Based on the answers given by patient is matched with the any one of the symptoms stored in database. Then the particular disease name is displayed . Then the doctor prescribes the medicines to the patient .

Output :

Medicines are prescribed.

**3.1.8 Functional Requirement 8:**

Introduction : Hospitals

Input : Patients are logged in to select the hospital and the other details to get an session.

Processing :

Patient enters into the hospital website and enter the desired hospital name , and selected doctor and enters the doctor-id , name and qualification and the other details and also selects the specialization. If he clicks on submit button the an interactive session is established between patient and doctor in different browsers. There is an rapport between patient and doctor .Then he prescribes the medicines.

Output :

Connection established and exchange of views by the patient and doctor.

**3.2 External Interface Requirements:**



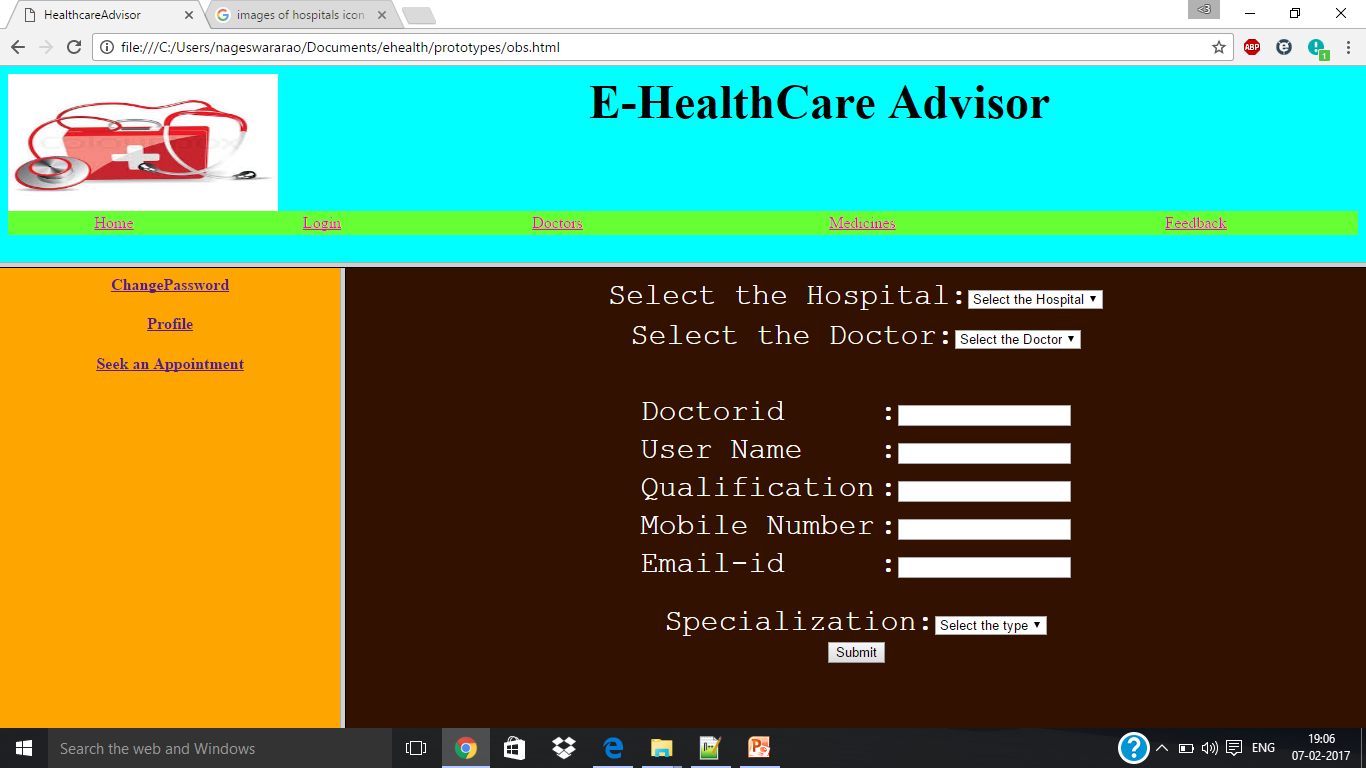
The above screen describes the login page which takes the username and password as input and there is a login button. The user after entering the username and password clicks on the login button .The user is allowed for the next page only if the username and password are correct i,e they should be matched with data available in the data base. If the username and password are correct then the user is displayed with message login successful and redirecting to the home page.



The above screen allows the user to change the password if the user has forgotten his/her password or if the user feels that it is unsecured. There are three text boxes named as current password , new password ,and confirm password .The user is initially prompted to give the current password and then new password and he again asked to confirm new password , if new password and confirm new password are some and if user initiates , save password button the password must be updated into database. Otherwise user should be displayed with the message re-enter the password.



The above screen helps the doctor to prescribe the medicines to patients .The form contains the text boxes age, name ,phone number and medicines to be prescribed based on age and gender. After filling these columns of form he clicks on submit button. Then the medicine form are sent to database and then sent from database to particular patient.



The above screen helps the patient to select the hospital , doctor in Online Consultation .This form contains the two selection boxes , selecting the hospital , doctor , and remaining are text boxes. The user has to know the doctor details .They are the unique-id called doctor-id which acts a primary key which is set as auto increment .They are other details of doctor such as qualification , mobile number , email-id , specialization . If he clicks on submit button the input data is taken to data base and stored in the database.



The above screen prototype screen helps the patient to seek an appointment from the doctor. This seek an appointment from contains name of the hospital, doctor ,timings , and phone number .The patient after entering the above text boxes and clicks on submit button .The an email is used to the patient that appointment is registered to you , and he is directed to the home page.



The above screen describes the profile of a particular patient .This profile forms takes the input fields like his/her first name, last name , occupation , age , email-id, phone number , height, weight gender, blood –group and address so on. The patient clicks on submit button the input is taken to the database and stored in the database and stored in the database with patient–id as the primary key is set as auto increment.

**3.3 Performance Requirements :**

**Response:**  
   The system shall give responses in 1 second after checking the patient’s information.

**Capacity**

  The System must support 1000 people at a time

**User-interface**   
The interface shall respond within 5 seconds.

**Conformity**  
  The systems must conform to the Microsoft Accessibility guidelines.

**3.4.Design Constraints:**

**1.Fully e-enabled scheduling and check in:**

i Ability to schedule appointments, check in ,health status.

ii Triage , easements ,treatments.

**3.5 Security Requirements:**

**1. Attacks at data collection level:**  These attacks may cause several threats to data collection level such as altering information, dropping some important data, or resending data messages.

**2. Spoofing Attack:** Where the attacker targets the routing information to perform several disruptions such as spoofing, altering, or replay the routing information, leading to complicate the network by creating routing loops.

**3.6 Maintainability Requirements:**

**Back Up**: The system shall provide the capability to back up the data.

**Error**  : The system shall keep a log of all the errors.

**3.7 Reliablity Requirements:**

1**.** The system shall provide be recovered with in 10 min if it is down .

2.The system will generate error messages when the user to enter invalid data.

**3.8 Availability Requirements**:

1.They would possess a centralized ,up to date and easily accessible health database while retain ownership and ensuring appropriate use of their data the report reads.

2. Regarding health care benefits, the report s suggest that they will have a comprehensive view of patients medical background ,enabling them to offer the most appropriate treatment.

**3.9 Database Requirements**:

**Hospital Information System:**

It is safe in e- document.

**Electronic Patient Records:**

It provide communication between providers , patients and payers in good manner

**Patient Information system:**

It is used to enter and update the data.

The Information stores gives a list of information like symptoms which improves the quality of services.

**3.10 Documentation Requirements**:

By storing the data electronically it provides proper documentation for furthur analysis.

**3.11 Operational Requirements:**

**Patient registration:**

Registration of patient is very important, which helps to identify the patient with an identification number.

**Consultation:**

1.It holds the details of the patients history.

2. Contain the details of medical reports.

3.Contain medical diet of patient.

**Doctor:**

It contain the patient problems and the related doctors who have taken care of particular care of patient .

**3.12 Site Adaption:**

**Internet:**

1. A provision of accessing the internet directly for this software to be facilitated.

2. The user with login only this facility can connect to the internet.

**Structured Analysis :**

A DFD is a graphical tool that describes the flow of data through a system and the functions performed by the system. It shows the processes the that receive input , perform a series of transformations , and produce the desired outcomes.

A DFD has four symbols:

1 .Process

2. Data Flow

3 .Data Store

4.Actor

**Process:**

A process is represented by a circle and it denotes transformations of the input data to produce the output data.

Process

**Data Flow:**

Data flows are data in motion. Data flows are represented by arrows , connecting one data transformation to another.

Data Flow

**Data Store:**

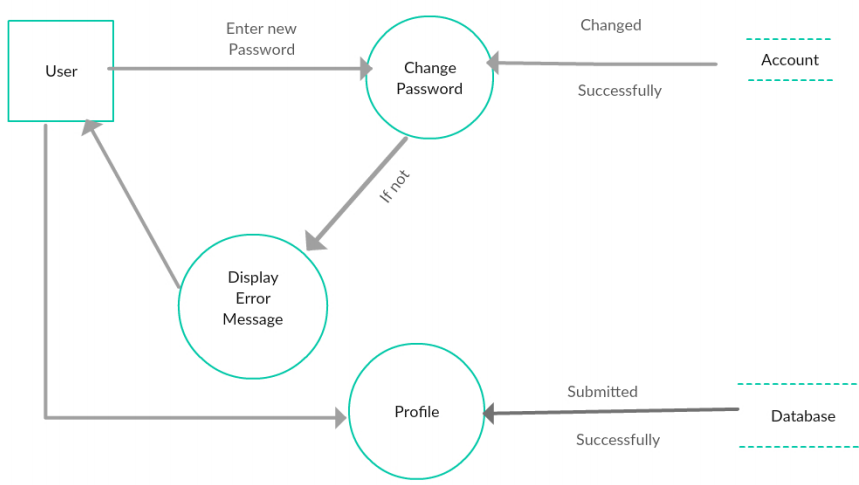
Data store is the data at rest. It is represented by an open ended rectangle or parallel lines. It shows the disk , data structures , files or database.

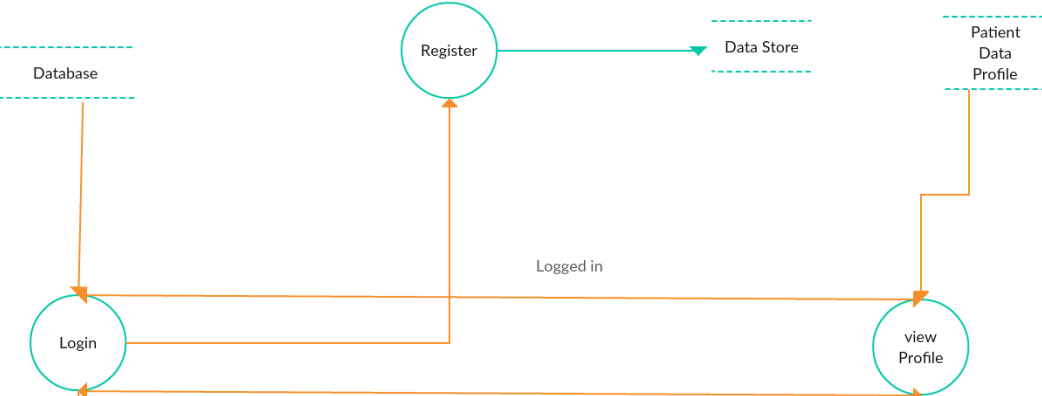
Data store

**Actor:**

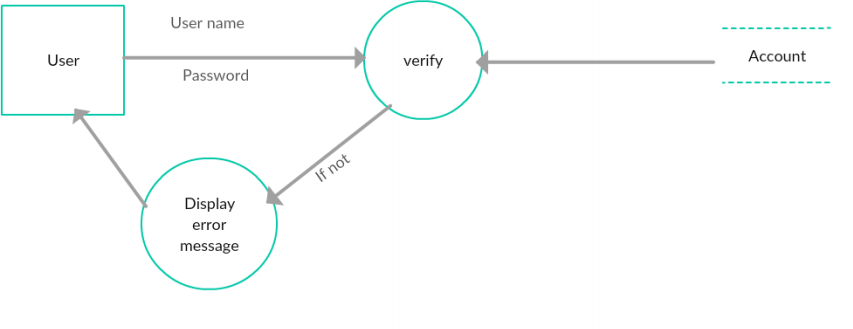
It is the external entity that represents the source or sink.It is represented by an rectangle.

Actor

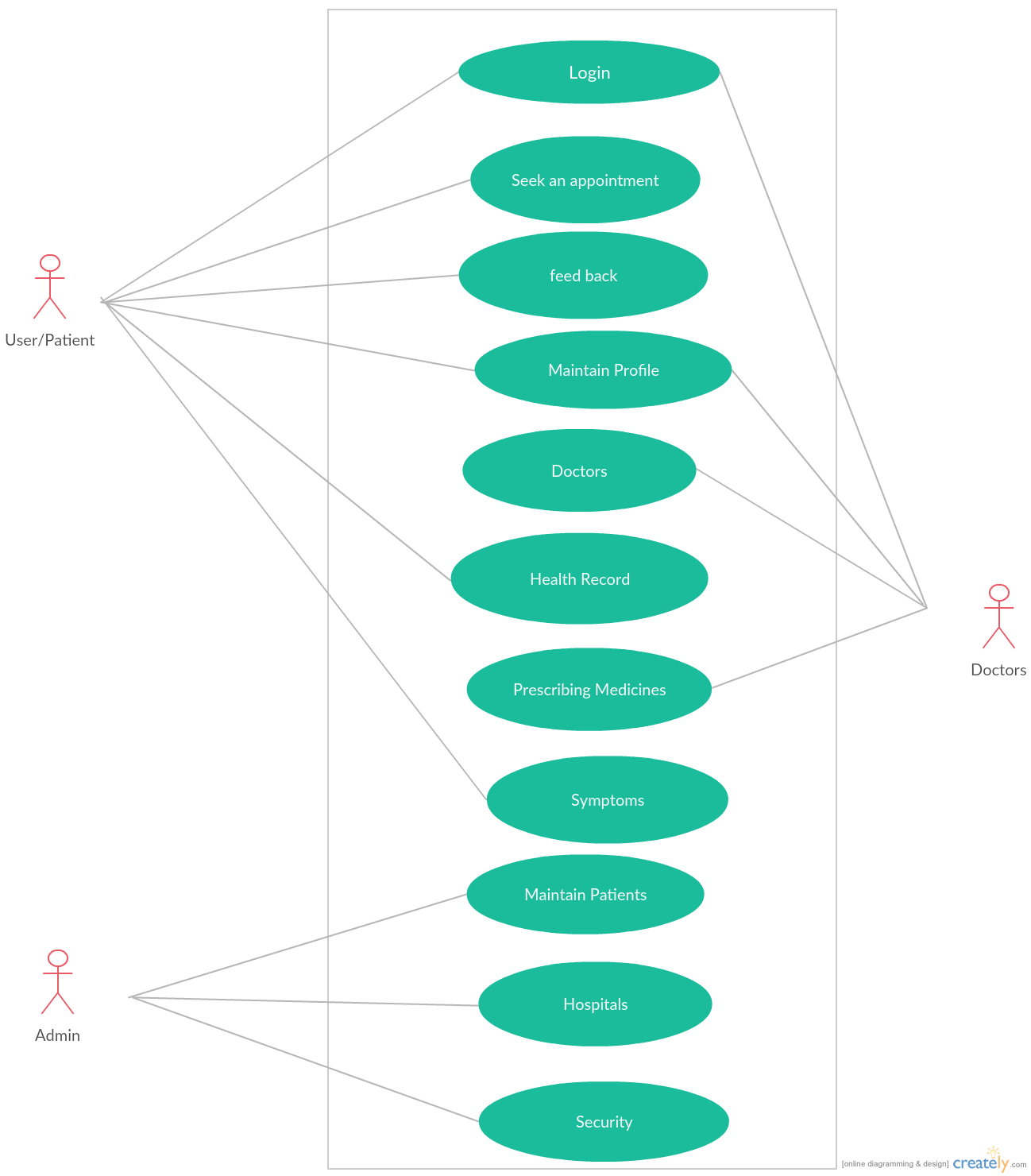
DFD 





****

USE CASE DIAGRAM :-



**USE-CASE DIAGRAM**

**Use Case 1: Installation**

Primary Actor: Administrator

Pre-Condition: Internet connection available.

*Main Scenario:*

1. User initiates EHM installation program.

2. System asks the user for the home directory in which all the working

Files will be created. User is also asked for the initial login and password.

3. User specifies the home directory and login/password.

4. System creates the working files in the specified home directory.

Working files contain:

a. Information of Patient.

b. Current Location of Doctor.

c. Backup & Restore of data.

*Alternate Scenario:*

a. Network failure.

b. Installation aborted.

Use cases related to system authorization:

**Use Case 2**: **Login**

Primary Actor: Doctor ,Patient, Administrator Pre-Condition: Internet connection available

*Main Scenario:*

1. Start the application. User prompted for login and password.

2. User gives the login and password.

3. System does authentication.

4. Main screen is displayed.

*Alternate Scenario:*

1. Authorization fails.

2. Prompt the user that he typed the wrong password.

3. Allow him to re-enter the password. Give him 3 chances.

**Use Case 3: Change Password**

Primary Actor: Doctor , Patient, Administrator

Pre-Condition: Certain User logged in

*Main Scenario:*

1. User initiates the password change command.

2. User is prompted for old password, new password and confirm new

Password.

3. User gives the old password, new password and confirm new

Password.

4. System does authentication.

5. New password is registered with the system.

6. User gets a message to his mail-id and mobile that password was changed.

*Alternate Scenario:*

1. Authorization fails.

2. Prompt the user that he typed the wrong password.

3. Allow him to re-enter the password. Give him 3 chances.

4. New password and confirm new password do not match.

4.1 Allow him to re-enter the attributes. Give 3 chances.

**Use Case 4: Search**

Primary Actor: Doctor , Patient, Administrator.

Pre-Condition: Certain User logged in.

*Main Scenario:*

1. User clicks on search box

2. System asks the user for the name of Patient.

3. User enters name/blood group/Image.

4. Click on the search button

*Alternate Scenario:*

1. Patient name does not exists.

2. Patient name exists but finger print /blood group does not match.

**Use Case 5: Profile**

Primary Actor: Patient

Pre-Condition: Patient logged in

*Main Scenario:*

1. Maintain the database

2. Grant role to other user

3. Revoke role from other users.

4. Back and restore the data

*Alternate Scenario:*

1. Data missing of Patient

2. Backup and restore is done every day

**Use Case 6: Record**

Primary Actor: Patient

Pre-Condition: Patient logged in

*Main Scenario:*

1. Maintain the database

2. Grant role to other user

3. Revoke role from other users.

4. Back and restore the data

*Alternate Scenario:*

1. Data missing of Certain Patient

2. Backup and restore is done every day

**Use Case 7: Delete data.**

Primary Actor: Patient

Pre-Condition: Patient logged in.

*Main Scenario:*

1.Patient initiates the ”delete data” functionality.

2. System asks for the name of the data.

3. The data is deleted.

*Alternate Scenario:*

1. Data does not exist.

2. Deletion fails, error message is displayed.

**Use Case 8: Create a security.**

Primary Actor: Administrator.

Pre-Condition: User logged in.

*Main Scenario:*

1. User selects the data in which the security is to be created.

2. User initiates the “create security” functionality.

3. System asks the user to enter the attributes of the security.

4. User specifies the following fields:

a. Name of Patient or place

b. health record

c prescription

5. An empty security of specified attributes is created.

*Alternate Scenario:*

1. A security with the given name already exists.

2. Security creation fails, error message is displayed.

**Use Case 9: Delete security.**

Primary Actor: Administrator

Pre-Condition: User logged in.

*Main Scenario:*

1. User selects the portfolio.

2. User initiates the “delete security” functionality.

3. System asks for the security name.

4. Security is deleted.

*Alternate Scenario:*

1. Security does not exist.

2. Deletion fails, error message is displayed.

.

**Use Case 10: Seek an Appointment**

Primary actor: Administrator

Pre-Condition: User logged in.

*Main Scenario:*

1 .user clicks into appointment page.

2 .System asks the patient name ,age ,place, and other details.

3 . User enter the details.

4.Click on submit button.

*Alternate Scenario:*

1.Appointment is not given.

**Use Case 11: Disease**

Primary actor: Doctor.

Pre-Condition: User logged in.

*Main Scenario:*

1. Patients have to fill the questions given by the system.

2. Based on the answers given by patient is matched with any one of the symptoms .

3. Then particular disease is displayed.

4. The doctor prescribes the medicines to the patient.

*Alternate Scenario:*

1. If the answers given by the patient is not understood, there is no prescription of medicines.

Data Oriented Analysis :

|  |  |
| --- | --- |
| Entity | Attributes |
| Admin | Address, Name, Gender, Admin-id ,password, Qualification |
| patient | Patient-id, Name, Age, Gender, Occupation, contact-info |
| Health record | Record-id, prescription, Tests, Insurance ,medical condition |
| Doctor | Doctor-id, Name, gender, address, qualification |
| Appointment | App-id, time, and doctor details |
| Compliant | Content, comp-id |
| report | Report-id, content |

