AIT Clinic Application

Stakeholders: Abhinav Singha (st120278@ait.asia), Gayatri Tirumalasetty(st119978@ait.asia)

Project Description: **AITClinicApp**is an application designed for doctors,nurses and patients who visit ait clinic. this app helps the users at different levels. In management level, the Admin,nurse and doctors can manage all patients and their prescriptions where as a patient can keep track of personal medical history.

Online Alternatives

There is no Online platform for ait clinic till date. So the head nurse at clinic was very excited about the application.

Distinct user class

1. Superusers: Admin, this person has the responsibility of creating authenticated accounts for nurses, doctors.

2.Nurse , Doctor: These users have the right to create new patients, manage their medical record.

2. Patients: they can login in to their personal medical account and keep track of their history. they cannot register themselves into the application.

Offline Alternatives

As of now, the clinic maintains the patient records physically by writing patient records on paper and also keeping prescriptions as a paper.

This application aims to give an online platform to the ait clinic so that the medical management becomes easy, efficient and paper free.

Our application main target is AIT community and if possible will link to ait Ldab.

Permanent Host & DNS Solution

We have a plan to host it on ait server as the clinic management is keen to have it.

Issue Tracker

https://trello.com/b/6AZv66ON/ps3.

Profile of User 1 (Student)

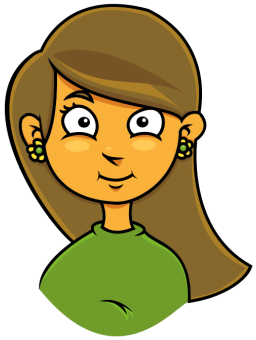
.

b. information

- name: Karan Raj Baruah
- age: 26
- occupation: Student
- housing situation: AIT dormitory

c. Short-term goals: To stay healthy in ait. And visit ait clinic for any health issue in case. I expect to get good services from the clinic

Using internet by: College Wi-fi



.

b. information

- name: Aniqua Nushrat Zereen
- age: 27
- occupation: nurse
- marital status: single
- housing situation: live in University’s Campus
- income: -

c. short-term goal: To serve for the ait clinic and keep good track of patient problems

e. devices used: laptop & smartphone

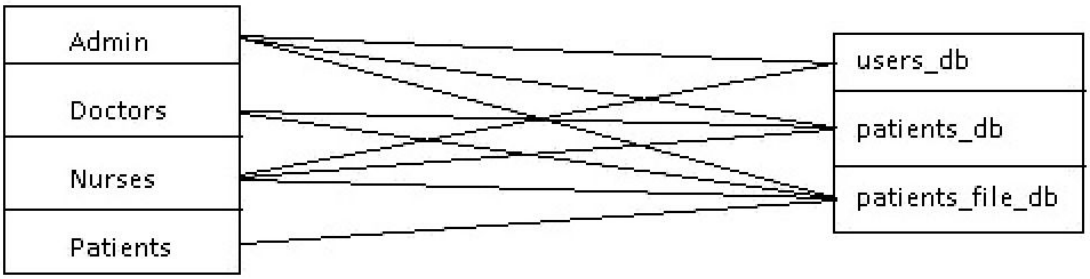
using internet by: Wifi

PS4

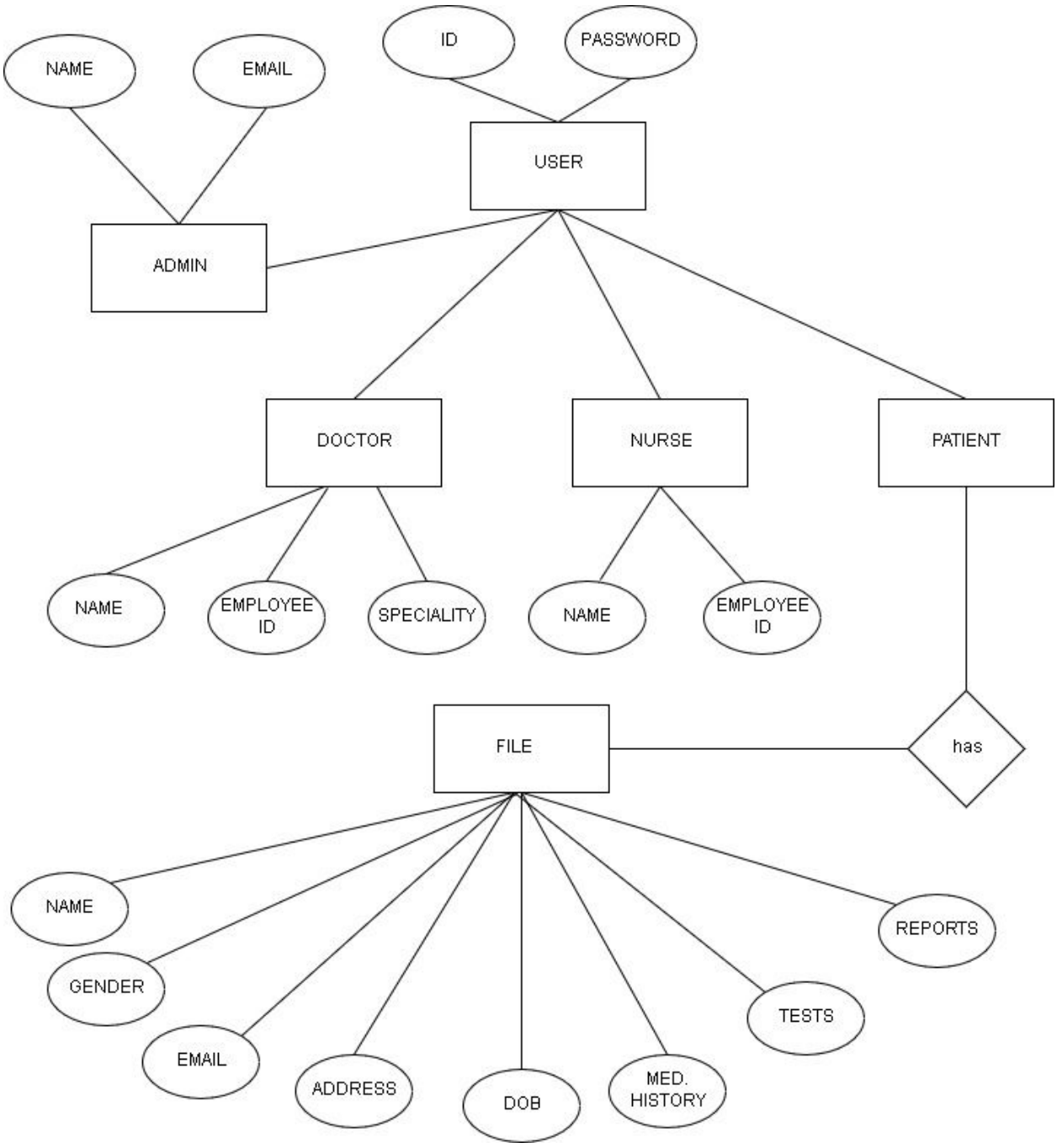
Documentation:

USER DATA MODEL

In our AIT Clinic application we basically have 4 kinds of users (Admin,Doctor,Nurse,Patient) and the content includes different databases which will hold user information and patient medical files. The admin has access to all the information and can read write and delete data from it, the doctor and nurse will have access to their personal information and the patient files(Patient list, patient files) whereas the patients will only be able to access their own personal information and read their medical history. The basic model is as
shown below.



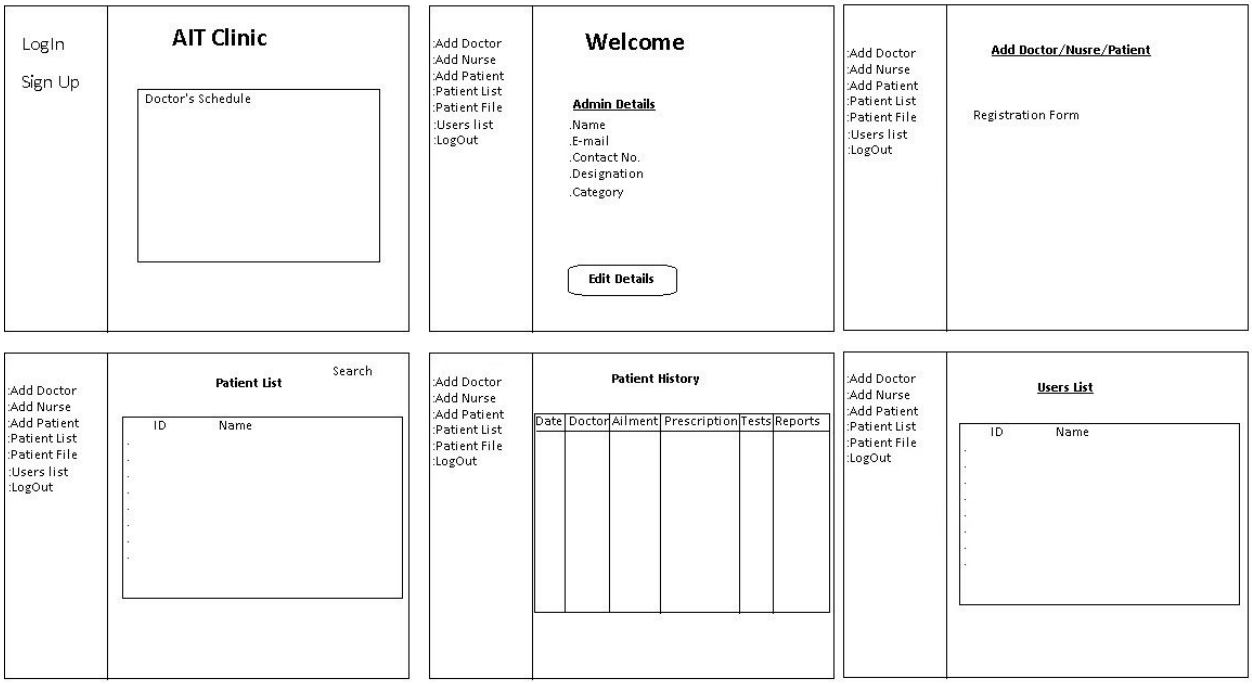
E-R Diagram



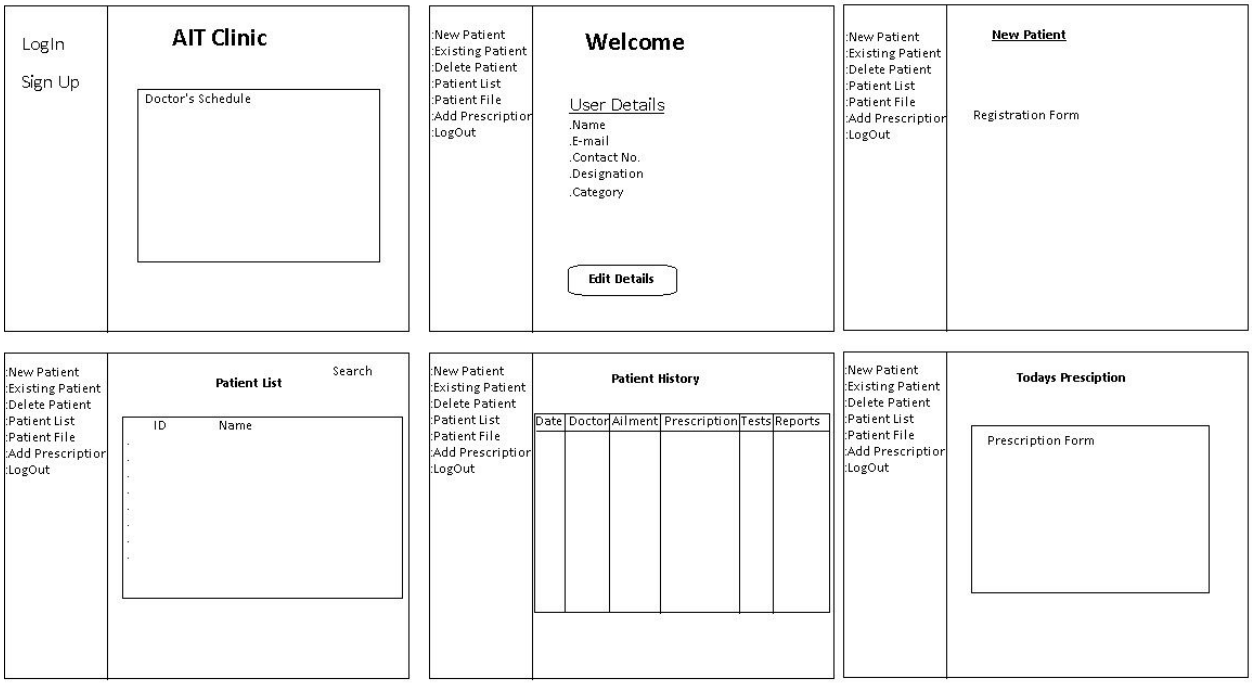
Sketch your user registration and management page flow.

Different users will have different privileges hence different pageflow for each user is as shown

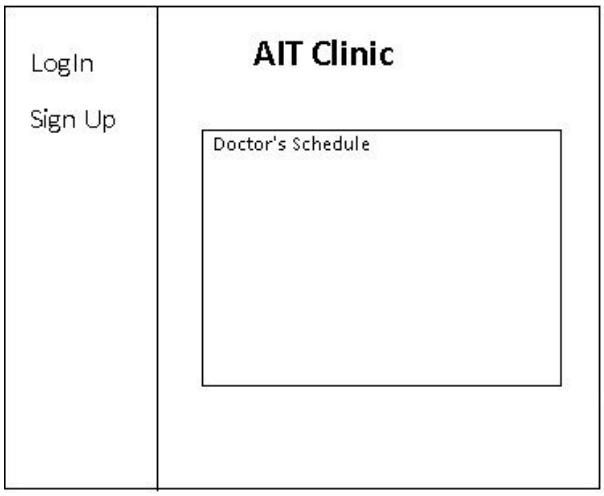
ADMIN

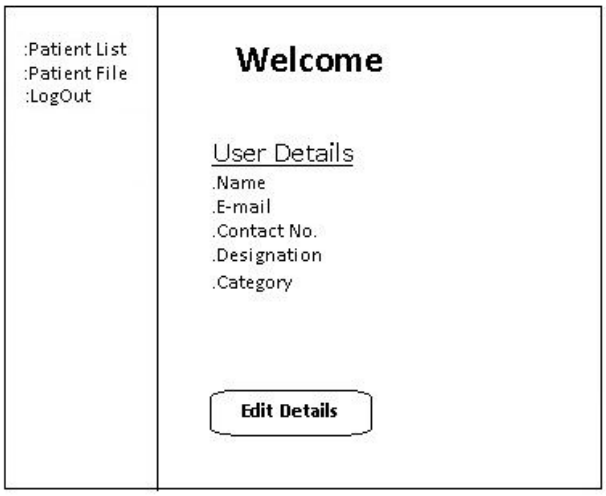

LogIn
Sign Up
AIT Clinic
Doctor's Schedule
:Add Doctor
:Add Nurse
:Add Patient
:Patient List
:Patient File
:Users list
:LogOut
Welcome
Admin Details
.Name
.E-mail
.Contact No.
.Designation
.Category
Edit Details
:Add Doctor
:Add Nurse
:Add Patient
:Patient List
:Patient File
:Users list
:LogOut
Add Doctor/Nusre/Patient
Registration Form
:Add Doctor
:Add Nurse
:Add Patient
:Patient List
:Patient File
:Users list
:LogOut
Search
Patient List
ID
Name
:Add Doctor
:Add Nurse
:Add Patient
:Patient List
:Patient File
:Users list
:LogOut
Patient History
Date
Doctor
Ailment
Prescription
Tests
Reports
:Add Doctor
:Add Nurse
:Add Patient
:Patient List
:Patient File
:LogOut
Users List
ID
Name

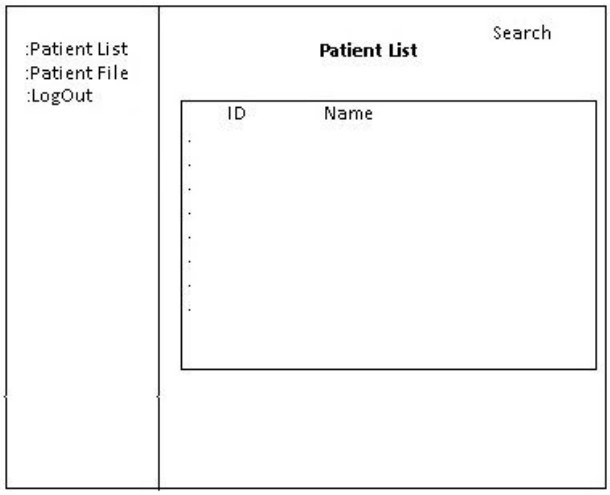
NURSE

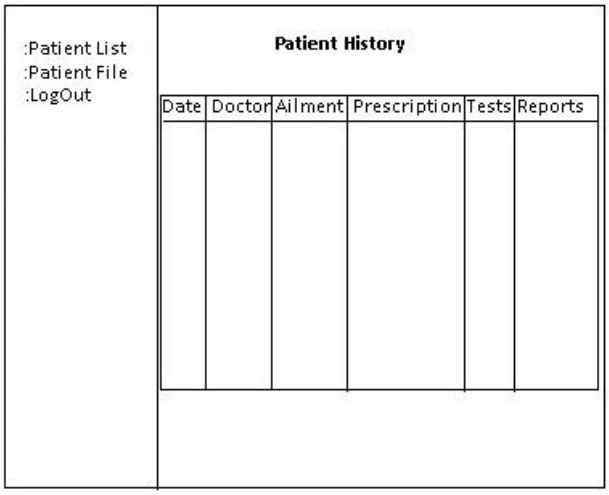

LogIn
Sign Up
AIT Clinic
Doctor's Schedule
:New Patient
:Existing Patient
:Delete Patient
:Patient List
:Patient File
:Add Prescription
:LogOut
Welcome
User Details
.Name
.E-mail
.Contact No.
.Designation
.Category
Edit Details
:New Patient
:Existing Patient
:Delete Patient
:Patient List
:Patient File
:Add Prescription
:LogOut
New Patient
Registration Form
:New Patient
:Existing Patient
:Delete Patient
:Patient List
:Patient File
:Add Prescription
:LogOut
Patient List
Search
ID
Name
:New Patient
:Existing Patient
:Delete Patient
:Patient List
:Patient File
:Add Prescription
:LogOut
Patient History
Date
Doctor
Ailment
Prescription
Tests
Reports
:New Patient
:Existing Patient
:Delete Patient
:Patient List
:Patient File
:Add Prescription
:LogOut
Todays Presciption
Prescription Form

DOCTOR

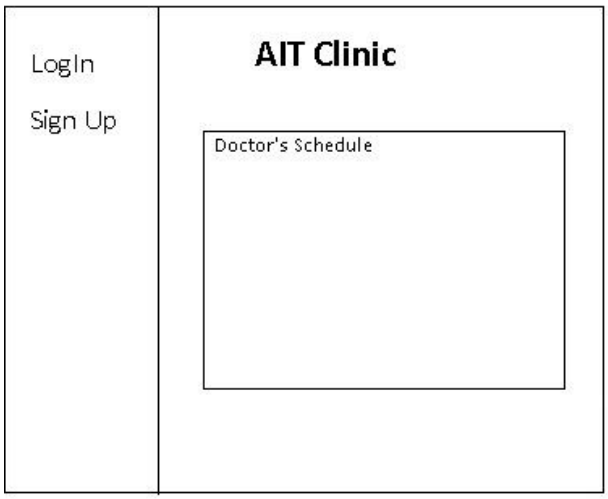

LogIn
Sign Up
AIT Clinic
Doctor's Schedule

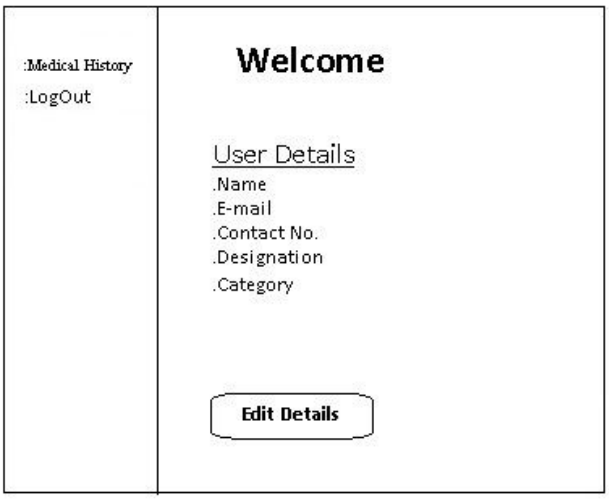

:Patient List
:Patient File
:LogOut
Welcome
User Details
.Name
.E-mail
.Contact No.
.Designation
.Category
Edit Details

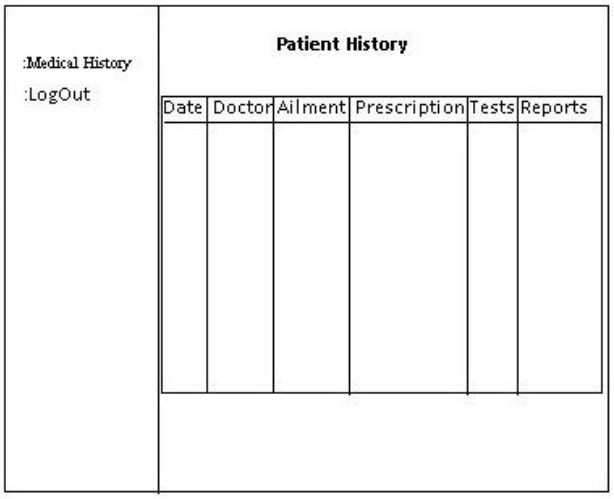

:Patient List
:Patient File
:LogOut
Search
Patient List
ID
Name


:Patient List
:Patient File
:LogOut
Patient History
Date
Doctor
Ailment
Prescription
Tests
Reports

PATIENT


LogIn
Sign Up
AIT Clinic
Doctor's Schedule


:Medical History
:LogOut
Welcome
User Details
.Name
.E-mail
.Contact No.
.Designation
.Category
Edit Details


:Medical History
:LogOut
Patient History
Date
Doctor
Ailment
Prescription
Tests
Reports

Learn how to avoid SQL injection and XSS attacks in your development framework and document and follow your plan for the rest of the project.

SQL Injection

Rails provides finder methods for all ActiveRecord (database) models. For example, to lookup a user using a primary key that was provided through the “id” request parameter, one would usually write:
User.find(params[:id])

Rails also provides so-called “dynamic finder methods”. It generates a “find_by_*” method for all database columns in model. If your “users” table have the “id”, “name” and “phone” columns, then it will generate methods so you can write things like this:
User.find_by_id(params[:id])
User.find_by_name(params[:name])
User.find_by_phone(params[:name])

The vulnerability is in these dynamic finder methods, not in the normal and often-used find method.
ActiveRecord protects you against SQL injection by escaping input for you. For example the following works as expected, with no vulnerability:
User.find_by_name("kotori'; DROP TABLE USERS; --")
# => SELECT * FROM users WHERE name = 'kotori\'; DROP TABLE USERS; --' LIMIT 1

But ActiveRecord also defines ways for the programmer to inject SQL fragments into the query so that the programmer can customize the query when necessary. The injection interfaces are documented and the programmer is not supposed to pass user input to those interfaces.

To prevent any scope of sql injection we are not using any SQL fragments and accessing all the data through Active Records only.

XSS attack prevention

Here are some general guidelines which we found would be useful on sessions.

- *Do not store large objects in a session*. Instead you should store them in the database and save their id in the session. This will eliminate synchronization headaches and it won't fill up your session storage space (depending on what sessionstorage you chose, see below). This will also be a good idea, if you modify the structure of an object and old versions of it are still in some user's cookies. With server-side session storages you can clear out the sessions, but with client-side storages, this is hard to mitigate.

- *Critical data should not be stored in session*. If the user clears their cookies or closes the browser, they will be lost. And with a client-side session storage, the user can read the data.

The HTTP protocol basically provides two main types of requests - GET and POST (DELETE, PUT, and PATCH should be used like POST). The World Wide Web Consortium (W3C) provides a checklist for choosing HTTP GET or POST:

Use GET if:

- The interaction is more *like a question* (i.e., it is a safe operation such as a query, read operation, or lookup).

Use POST if:

- The interaction is more *like an order*, or
- The interaction *changes the state* of the resource in a way that the user would perceive (e.g., a subscription to a service), or
- The user is *held accountable for the results* of the interaction.

Apart from following these guidelines we plan to use SSL as any computer in between you and the server can see the sensitive information if it is not encrypted with an **SSL** certificate. When an **SSL** certificate is used, the information becomes unreadable to everyone except for the server you are sending the information to. Therefore for our application we are using open SSL certificate.

Using the techniques of test-first and behavior-driven development we've studied in class and lab, build up the basic user registration and login pages. Set up SSL for the login page to prevent password sniffing, and set up SSL for all authenticated interaction if session hijacking is a concern.

SSL has been set up using a self signed certificate in OpenSSL as mentioned above. Please check the link to see: https://web10.cs.ait.ac.th

Database Security

Our database is handled by postgresql and our application is able to access the database through peer security protocol of postgresql as the credentials of the linux account and postgresql match, and also because it is deployed on the same machine as database thus if someone else wants to access the database they should know the credentials of the linux account or the database passwords which are in capistrano, the application has no saved database passwords

Again using BDD, build the user administration pages. Make sure admin can 1) see recently registered users, 2) ban a user, and 3) see statistics on user registrations.

Feature: adminScenarios

Scenario: Admin Login
Admin should be able to login
Given I am an admin
And I visit home page
When I should click "**Log in as admin**" link
Then I should see a "**Log in**" form
When I click Log in button
Then I should go see Admin Dashboard

Scenario: Add users
inorder to add ordinary users to
admin should register them.

When I visit admin dashboard
Then I should see "**Add User**"
When I click "**Add User**"
Then I see "**New User**" form
When I fill in the form and click "**Create User**"
Then I should see "**User was successfully created**"

Scenario: view stastics
Admin should be able to see user statistics

When I visit admin dashboard
Then I should see "**statistics**" link
When I click on "**statistics**" link
Then I should see "**users statistics**"

Scenario: Ban users
inorder to prohibit users from using the
application admin should ban.

When I visit admin dashboard
Then I should see "**user list**"
When I click "**user list**"
Then I see "**Users**" list with Ban option
When I click "**ban**" button
Then I should see a message saying "**User was successfully ban.**"

Scenario: User Login
User should be able to login
Given I am a user
And I visit home page
When I should click "**Log in as member**" link
Then I should see a "**Log in**" form
When I fill in the form and click "**Log in**"
Then I should see "**Logged in**"

Use your issue tracker and CI server to their full potential. Start an issue in the issue tracker before you begin any work. Get email notification of broken builds running, and get the CI server to report test coverage and code quality metrics

Issue tracker- trello is being used to keep track of our work progress. You can check it out at : https://trello.com/b/j0a3yEPu/webapplication-project

Also jenkins is used for CI.

PS 5:

Think about how to classify each of the types of content on your site. Articles, comments on articles, news items, questions and answers, documents to be approved or rejected with comments, and so on.
Each type of content will require a different flavor of data model and moderation/approval/versioning Process

Our application is about clinic management, with four kinds of users:

1. Admin
2. Doctor
3. Nurse
4. Patient

We do not have need for articles/comments/news. But we are planning to include question & answer : patients/ait students can ask questions on health and Doctor/nurse will be answering the questions.
document approval: patients will submit their health insurance document to be approved by the Nurse.

Design your content data model and discuss the implications of your design choices with potential users of the system. In general, there should be at least one category of user-submitted content with authoring workflow specific to your project and at least one question and answer forum. However, the specifics will be very dependent on your project's requirements. Document the design and your discussions on your site

The content data model that was discussed in PS4 includes doctors info, nurse info, patient details, patient-record, medicine details. This has been discussed with our client (nurses at ait clinic).
Our app consists of user submitted content, i.e, doctors, nurses, patients submit their personal details. Patients also submit their insurance documents for approval.

Though question and answer forum doesn't fit into the requirements, we'll talk to our client about this.

Specify the workflow for each kind of content on your site and discuss the implications of your design choices with potential users of the system. Document the workflow design and your discussions on your Site.

We have three kinds of data-

1. User details(Employee database) - The user details is used to authenticate users, to find the access rights of different types of employees
2. Patient details(Patient database) - The patient details is used to authenticate patients, is used to reference each patient's medical records and is available to employees for access.
3. Patient Records(Patients' medical records) - This is a more personal sort of data so is not accessible to everybody. This database can be accessed by attending doctor, the nurse and the particular patient whose file it is. There is one more table(i.e. medicines) but it is like a part of the patient file. It stores the medicines prescribed by the doctor on each visit for every patient and is referenced to patient record.

The design of database is pretty simple
The user detail is stand alone and is accessible by only the admin
The patient details is available to everyone except the patients and references patient records
The patient record is referenced by patient and references medicines

Thus we have a **one to many** relationship between patient and patient records
Thus we have a **one to many** relationship between patient records and medicines

Design your versioning system. What is the versioning system for each type of content? Do you need a complete history of every version, or just the current version with the author of the last version? Make sure you think carefully about the needs of the user community. Document the versioning design on your site.

Our versioning system has been built using a ruby gem called "paper-trail".
This gem helps in storing various versions of a content/model in a version table.
In our application, versioning is used for patient_record model, medicine model, because the users (doctors, nurses, patients) might want to revert back and look at a patient's health history/condition and also in case of medicine inventory, the nurse might need a way to keep track of medicine available in the inventory.

Write UATs for a “skeletal” implementation of the workflow for at least one type of content on your site then get the UATs to pass. Users should be able to get something up on your server and see it Published.

The UATs for User models (includes doctors and nurses) have been designed as:

Scenario: User Login
User should be able to login
Given I am a user
And I visit home page
When I should click "**Log in as member**" link
Then I should see a "**Log in**" form
When I fill in the form and click "**Log in**"
Then I should see "**Logged in**"

Scenario: Nurse has to view patient list
if the logged in user is nurse, then the user
should be able to carry out the various tasks.
Given I am a Nurse
Given I am logged in
When I click "**Patient list**" link
Then I should see "**Patients**" list
When I click the history link "**Patient history**" link
Then I should see all records "**Patient Records**"
When I click new patients "**New Patient**" link
Then I should see "**New Patient**" form
When I click "**Create Patient**" button
Then I should see content "**New Patient created successfully**"

Scenario: Doctor has to view the patients diagnosed by
if the logged in user is nurse, then the user
should be able to carry out the various tasks.
Given I am a Doctor
Given I am logged in
When I click "**Patient list**" link
Then I should see "**Patients**" diagnosed by me
When I click the history link "**Patient history**" link
Then I should see all records "**Patient Records**"

And the content modified by these users is patient details and patient_record.

The UATs for patient role are:

Scenario: patient Login
Patient should be able to login
Given I am a patient
And I visit home page
When I should click "**Log in as Patient**" link
Then I should see a "**Log in**" form
When I fill in the form and click "**Log in**"
Then I should see "**Logged in**"

Scenario: Patient functions
Inorder to perform patient tasks, patient
has to do these tasks
Given I am a Patient
Given I am logged in
When I click "**My history**" link
Then I should see "**Medical Record**"
When I click the "edit" link
Then I should be able to "edit" my content

Note that all the scenarios have been passed.