CSE 360 Open-Source Journey

Introduction

Software engineering is the science of developing the softwares. This is one f the classic cources in the computer science field that, it is being instructed in amost all universities. The software engineering is closely related to the software making industry which always change its practices to drive itself to the effiiency in making softwares. But the software engineering cources fails to embrace thses changes in the same pace as the industry does. For instance, when the world is moving more into agile, scrum and open development and all new bleeding edge techniques in software making, most of the software engineering cources in undergradute degrees are more concentrated on the waterfall and other practices, which usually stays in the text. We observed that most of these models doesn't give much experience on software making process to the students. This was the main reason why we wanted to update the course structure and incorporate more practical aspect to the course.

From the previous works[1], we have learned that most of the leading software makers in the industry encourages the use of existing softwares for rapidly scaling the architecture. The paper(Wei-Tek Tsai et al, 2014[1]) shows that the industry embraces the use of well nurtured open-source softwares. This lead us to reconstructing the entire course with prime focus on open-source software development. We created a series of lectures and assignments that included both theorectical aspects and industry oriented practical tasks on software engineering.

We implemented this reconstructed course plan at software engineering cource of fall 2014 at Arizona State University, Tempe, USA. The chapters that follows tells the journey that we had during the software engineering course of Fall 2014.

Assignment 1

The objective of this assignment was to introduce the students to the OSS environment and software development. In the initial classes of this course, we gave presentations and materials to the students on how the open-source-software developemnt works and the importance of the OSS in the current software industry. Soon we realized the fact that there is no better way to introduce OSS to students than ask them to contribute something to an active OSS community.

In the first assignment, we asked the students to pick an OSS software that they were passionate about. Their challenge in this assignment was to contribute some work back to that community. We gave them proper guidelines on how to get introduced to an OSS community and the ettiquetts to be followed. This included things like contacting community members through mailing lists, IRC channels, reading the code base, fixing bugs and writing a patch or documentation. We gave them three and half weeks to get to know the community and to make the contribution, at the end of which they had to submit a report on all the interactions happened with the community and the experience from their point of view, with a breifing of their contribution to the OSS community.

Most of the students didn't have much trouble finiding an OSS that they were impressed with. The list was dominated by the daily used doftwares like Mozilla Firefox, Wordpress, Notepad ++, Twitter Bootstrap. This pattern stresses over the important role of OSS in the current daily used softwares. At the end of the assignment, most of the students were successful in making a contribution to the community. A few went over the open source code and wrote new functionalities, some fixed bugs from bug tracker lists and the rest did documentation related contributions to the community.

The assignment was perfect introduction to OSS devlopment and software engineering. Not only the student understood how the OSS communities work, they got a glimpse of team work and procedures in the making of software and engineering behind it.

Assignment 2

In the first assignment we aimed at introducing the students to the way OSS are made. The next step was to introduce them to them to latest trends and practices happening in the OSS space and its impact on software development industry. The second assignment was tailor made with that objective in mind.

The assignment instructed the students to research the open internet and come up with a latest trend in software engineering or industry that has some influence or assosiation with open-source-software development. Based on the theme they picked for the assignment, they had to make a wiki page explaining all their findings and analysis. Later we asked students to make a short presentation on their analysis of the findindings and present it. This was a short assignment compared to the first assignment. We received wide veriety of topics from stdents ranging from open-source Version Controlling, Github, Continuous Integration frameworks, Scalability with open source softwares, Cloud solutions with open-source-softwares, testing with open-source-softwares and requirement analysis of OSS etc.

We provided infrastructure for the students to build their wiki page at course's wiki space. They created the wiki page filled the findings respective to their topic and the analysis or conclusion of the specific trend in the software making process.

The OSS Project

The key objective of the new course structure is to introduce OSS and its benefits to the students. So we designed the term project with this key goal in mind. The project requires students to form a team of three and create a web application using some given requirements, using open-source powered technologies.

At the start of the project, we gave them two requirement documents, which explained them the requirements of the website in simple terms. Based on their interest they created teams of three people and selected one of the requirement from two available options.

Once the requirement is chosen from either of the two sets, the student team had to go through each of the following step to deliver the complete project.

1. Requirement analysis.
2. Use case design.
3. Architecture design.
4. Deciding on the open-source softwares to be used.
5. User Interface mock up design.
6. Coding.
7. Unit Testing.
8. Functional Testing.
9. Load Testing.

In the following sections, we will explain each of this steps and what students were expected deliver.

1. Requirements

We gave students two set of requirements, which they could choose from to make the web application.

1. Online image storage and edit application
2. Online ticket selling application.

The requirements are specified below.

Requirements specification for `Online Image storage & editing` web application.

Purpose: Develop a web application, which can store user's images privately via uploading. The web application should also be capable of editing the image and save them.

Scope: A web application, which can be accessed from a set of browsers specified in the system constraints.

Functional specification:

1. The user should see one of the following on accessing the website URL.
    a. If the user is not already logged into the application, then redirect the user to the Login/Register page that shows form for logging in and registering to the web application.
    b. If the user has already logged into the web application, show the home page for the logged in user.
2. From the login/register page, the user should be able to register by filling email, password and name fields in the register form. The successful /unsuccessful registration should alert user about the respective registration status.
3. From login/register page the user should be able to fill the existing and valid email and password to login to the web application. A successful login takes the user to user's home page. An unsuccessful login status should be alerted to the user.
4. The user's home page should contain all images that the user had uploaded earlier, with options to upload new photos and access the profile settings.
5. On clicking the "upload new photo" option, the user should be able to upload a new image to user's space, which should be displayed on user's home page.
6. On accessing the profile settings page, user should be able to see user's profile picture. Also user should be able to upload a new profile picture, which should replace the old profile picture.
7. In the home page when user clicks on an image, user should be able to edit the brightness of the image and save it permanently.
8. The user should have an option to log out at anytime the user wishes to.
9. System should be able handle multiple users and one user should only be allowed to view the content uploaded by the same user.

System constraints:

- The web application should work completely in most modern web browsers. Specifically on Google chrome v32+ and Firefox v24+.
- The system should be able to handle high volume of traffic at the end of the project.

Requirements specification for `Online ticket selling` web application.

Purpose: Develop a web application, which can sell tickets for various events to its users.

Scope: A fully functional web application, which can be accessed from a set of browsers specified in the system constraints.

Functional specification:

10. The user should see one of the following on accessing the website URL.
    a. If the user is not already logged into the application, then redirect the user to the Login/Register page that shows form for logging in and registering to the web application.
    b. If the user has already logged into the web application, show the home page for the logged in user.
11. From the login/register page, the user should be able to register by filling email, password and name fields in the register form. The successful /unsuccessful registration should alert user about the respective registration status.
12. From login/register page the user should be able to fill the existing and valid email and password to login to the web application. A successful login takes the user to user's home page. An unsuccessful login status should be alerted to the user.
13. The user's home page should list all events that the user can buy ticket for.
14. When user clicks on an event, all the event details should be shown like (Title, type of event, Venue, date, time, tickets available, total tickets). If tickets are available for that event, user should be able to click on the book button in event details page. By clicking book button user should be asked to select the number of tickets needed. If the number given is smaller or equal to the number of tickets available, then user should be able to book tickets successfully.

15. User should have an option at the home page to see all the tickets booked by the user with event details ordered by event time.

16. User should also access a profile settings page from homepage. On accessing the profile settings page, user should be able to see user's profile picture. Also user should be able to upload a new profile picture, which should replace the old profile picture.

17. The user should have an option to log out at anytime the user wishes to.
18. System should be able handle multiple users and one user should only be allowed to view the content uploaded by the same user.

System constraints:

- The web application should work completely in most modern web browsers. Specifically on Google chrome v32+ and Firefox v24+.
- The system should be able to handle high volume of traffic at the end of the project.

2. User case digrams/User stories.

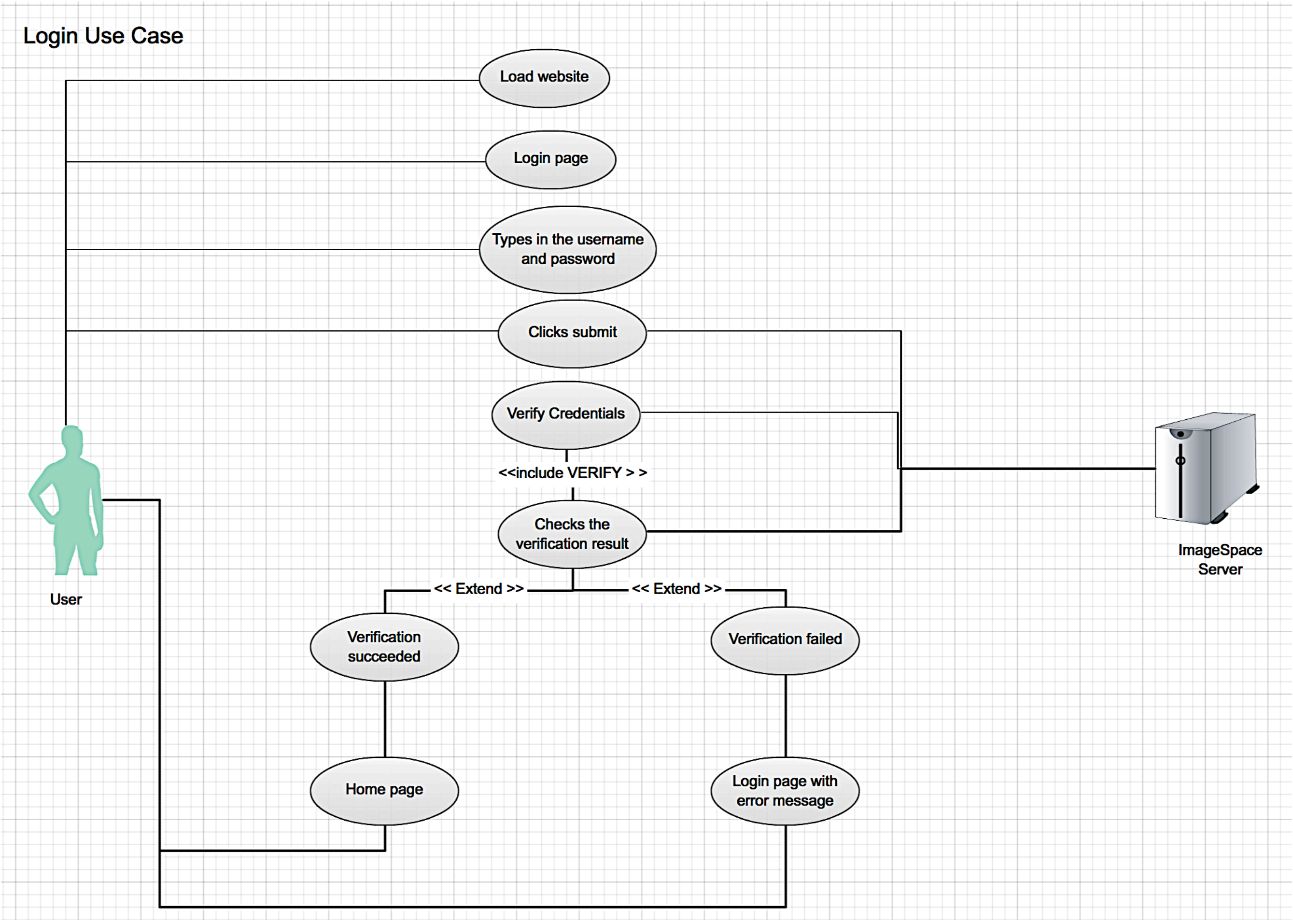
Onece the team selected the requirement that they want to work with, the first deliverable in the project was user case diagram or user story.

PROBLEM:
Identify features and functionalities in the web application & build user case digrams for each functionality. The feature needs to be analysed to first to understand the steps involved in a functionality. The analysed user case should be converted to a diagram with standard UML. Instead of UML based diagrams teams can opt for agile/scrum style user stories too.

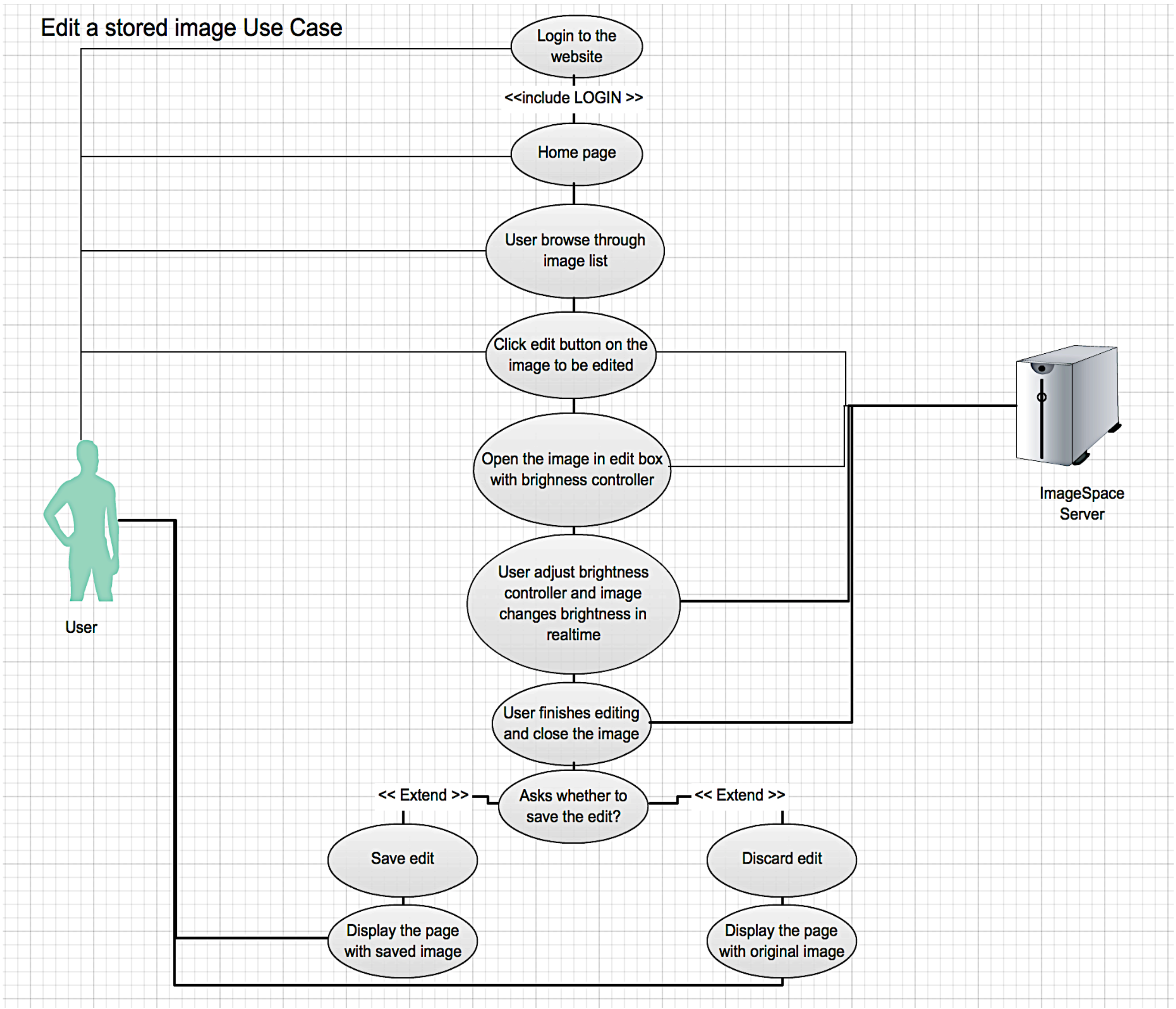
To identify the functionalities and user cases, the requirement document given in the previous section is to be used.

Sample Solution(User diagrams)

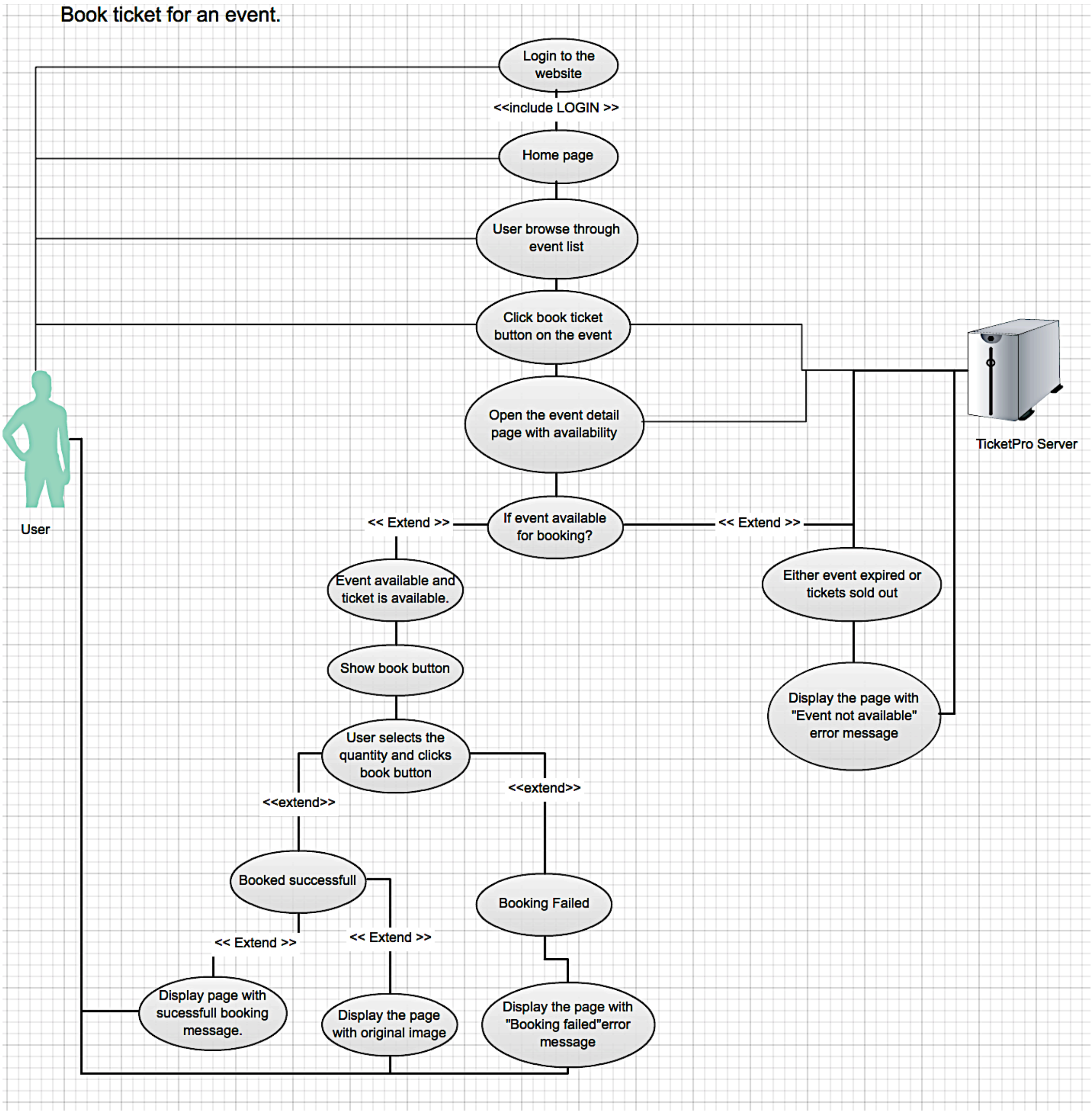
1. Login user case.(ImageSpace application)



2. Editing an image(ImageSpace app)


Edit a stored image Use Case
Login to the website
<<include LOGIN >>
Home page
User browse through image list
Click edit button on the image to be edited
Open the image in edit box with brighness controller
User adjust brightness controller and image changes brightness in realtime
User finishes editing and close the image
<< Extend >>
Asks whether to save the edit?
<< Extend >>
Save edit
Discard edit
Display the page with saved image
Display the page with original image
User
ImageSpace Server

3. Book ticket for an event (Eventpro app)


Book ticket for an event.
Login to the website
<<include LOGIN >>
Home page
User browse through event list
Click book ticket button on the event
Open the event detail page with availability
If event available for booking?
<< Extend >>
<< Extend >>
Event available and ticket is available.
Either event expired or tickets sold out
Show book button
Display the page with "Event not available" error message
User selects the quantity and clicks book button
<<extend>>
<<extend>>
Booked successfull
Booking Failed
<< Extend >>
<< Extend >>
Display page with sucessfull booking message.
Display the page with original image
Display the page with "Booking failed"error message
User
TicketPro Server

Sample Solution(User story)

1. Login user story.(ImageSpace application)

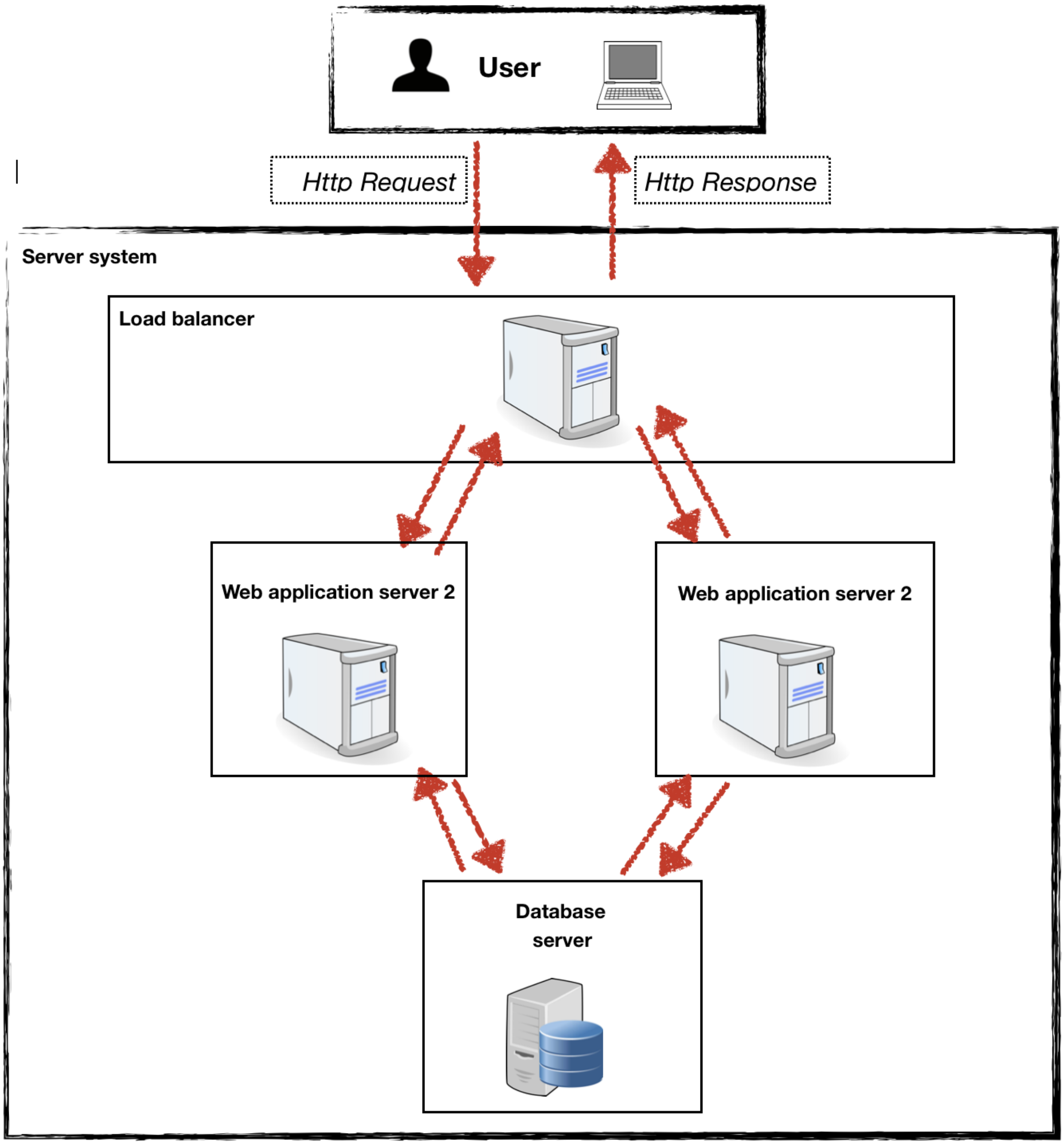
Architecture design.

The performance and scalability of a system has huge connection with the underlying architecture of a system. After the usercase/user story we asked students to come with an architecture with scalability in mind.

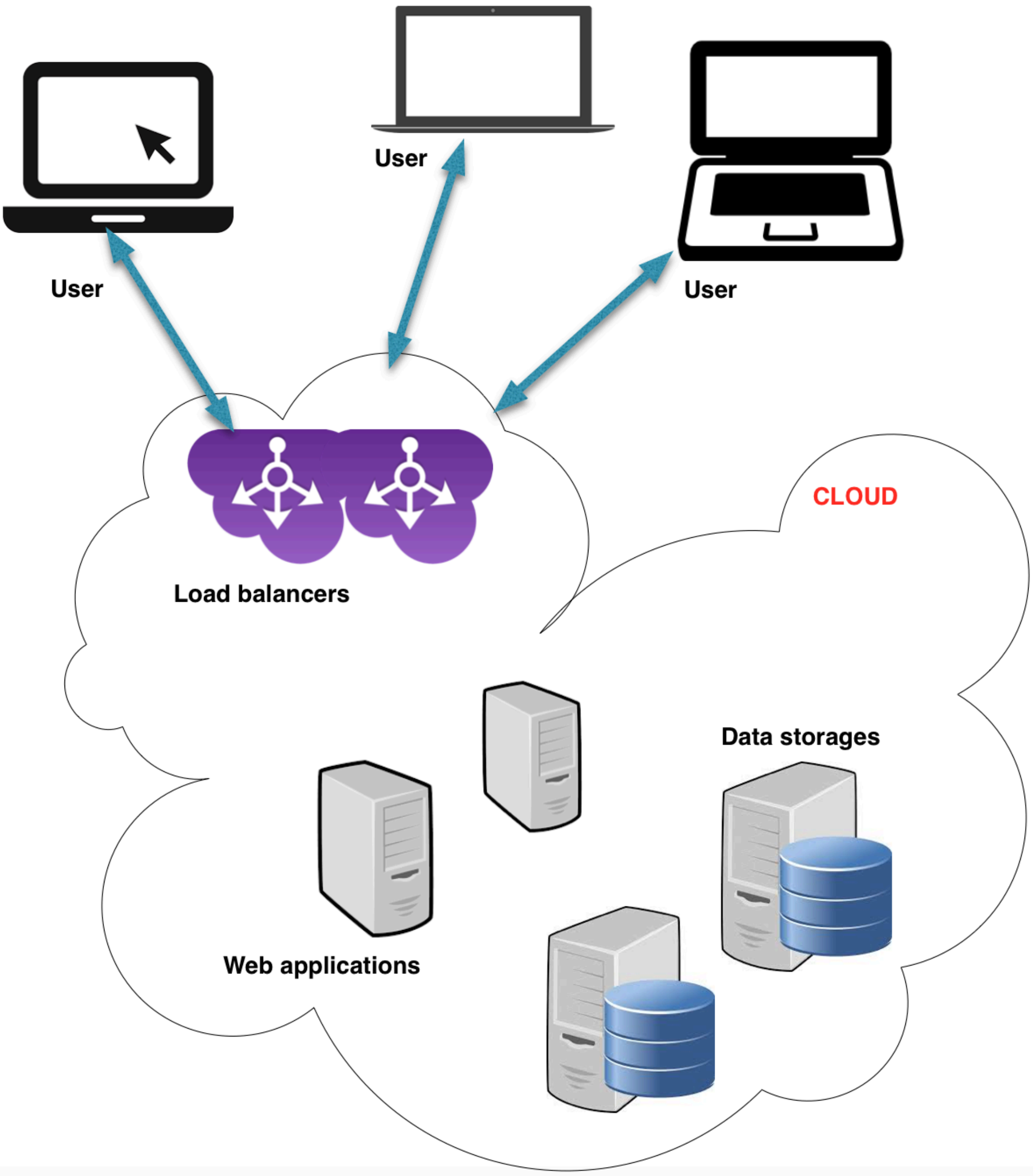
PROBLEM:
Design an architecture for your web application. As the requierments say, scalability should be an important parameter in your decision.

SOLUSION:
We received two categories of architectures as solution. One set of architectures used load balancers and multiple application servers. The other set made use of the cloud architecture and all the free benefits in the cloud like load balancing, clustering and data replication. Both solutions are presented here.

SOLUTION 1


User
Http Request
Http Response
Server system
Load balancer
Web application server 2
Web application server 2
Database
server

Solution 2


User
User
User
CLOUD
Load balancers
Data storages
Web applications

3. Adoption of open-source softwares

After a architecture is decided each team needed to find appropriate open-source components to fullfil their architecture. So the next delivarable was a document with details of each open-source software that could fit in the architecture build the system.

PROBLEM:

Identfy the software components you need to fullfill your architecture and try to find open source softwares which could be used used to implement that component. For example, the you could choose the web application framework to be either Django framwork or Rails framework or any other open-source frameworks.

- **Load balancer - Nginx**

NGINX

What does the software do?

Nginx is an open source reverse proxy server, which supports HTTP and HTTPS protocols. It can be used as an excellent load balancer, cache and application server. It uses FastCGI, WSGI or Passenger interface specification to communicate with other web servers.

Why is that particular OSS chosen?

Nginx is easy to integrate with other web servers. Also it is very lightweight. Other solutions like Apache HTTP server is heavy and consumes lot of memory, CPU resources. The whole NGINX loads in leads than 10MB memory. It has high availability and performance. Also it includes pluggable architecture which helps to add lot of additional features. There are lot of usable plugins like performance monitor, analysis plugin etc.

Licensing information of that OSS

It is licensed under the 2-clause BSD-like license and it runs on Linux, BSD variants, Mac OS X, Solaris, AIX, HP-UX, as well as on other *nix flavors.

Link to the OSS.

http://nginx.org

http://hg.nginx.org/nginx

- **Application Server**

Ruby On Rails

What does the software do?

Ruby on Rails(ROR), or simply Rails, is an open source web application framework written in Ruby. Rails is a full-stack framework that emphasizes the use of well-known software engineering patterns and paradigms, including convention over configuration (CoC), don't repeat yourself (DRY), the active record pattern, and model–view–controller (MVC).

Why is that particular OSS chosen?

ROR is made following the best practices in the industry. SO it does the best for an application server without much tweaking. Also its easy to scale ROR application because of the design of the framework is exclusive for scalability. More importantly, it is extremely simple to develop application using ROR and ruby. The learning curve is pretty small.

Licensing information of that OSS

It is licensed under the MIT license and available under all *nix environment and windows.

Link to the OSS.

http://rubyonrails.org

https://github.com/rails/rails

- **Database server**

PostgreSQL

What does the software do?

PostgreSQL, often simply "Postgres" is an object-relational database management system (ORDBMS) with an emphasis on extensibility and standards-compliance. As a database server, its primary function is to store data, securely and supporting best practices, and retrieve it later, as requested by other software applications, be it those on the same computer or those running on another computer across a network (including the Internet).

Why is that particular OSS chosen?

PostgreSQL is completely open source and free. The another option is MySql. But it is paid software if want to scale beyond a point to use in an enterprise environment. PostgreSQL can handle workloads ranging from small single-machine applications to large Internet-facing applications with many concurrent users. Recent versions also provide replication of the database itself for security and scalability

Licensing information of that OSS

PostgreSQL is developed by the PostgreSQL Global Development Group, a diverse group of many companies and individual contributors.[13] It is free and open source software, released under the terms of the PostgreSQL License, a permissive free software license. It runs on all *nix environments and windows.

Link to the OSS.

http://www.postgresql.org

http://www.postgresql.org/ftp/source/

- **Functional Test**

Selenium

What does the software do?

Selenium is a portable software testing framework for web applications. Selenium provides a record/playback tool for authoring tests without learning a test scripting language (Selenium IDE). It also provides a test domain-specific language (Selenese) to write tests in a number of popular programming languages, including Java, C#, Groovy, Perl, PHP, Python and Ruby.

Why is that particular OSS chosen?

The tests can then be run against most modern web browsers. Selenium deploys on Windows, Linux, and Macintosh platforms.

Licensing information of that OSS

It is open-source software, released under the Apache 2.0 license, and can be downloaded and used without charge.

Link to the OSS.

http://docs.seleniumhq.org

- **Load Test**

Jmeter

What does the software do?

Apache JMeter is an Apache project that can be used as a load testing tool for analyzing and measuring the performance of a variety of services, with a focus on web applications.

JMeter can be used as a unit test tool for JDBC database connections, Web services,HTTP, generic TCP connections and OS Native processes. JMeter can also be configured as a monitor, although this is typically considered an ad hoc solution in lieu of advanced monitoring solutions. It can be used for some functional testing as well.

Why is that particular OSS chosen?

JMeter has an awesome GUI client which makes testing lot easier. Also, it generates graphs which are easy to understand.

Licensing information of that OSS

It is open-source software, released under the Apache 2.0 license, and can be downloaded and used without charge.

Link to the OSS.

http://jmeter.apache.org

- **Cloud Hosting (Optional)**

It is best to host the web application on cloud to make sure it scales for any requirement. The cloud best suited here is Heroku(http://heroku.com). It is free and it has full support for Ruby on Rails and PostgreSQL. But deploying application to cloud is optional.

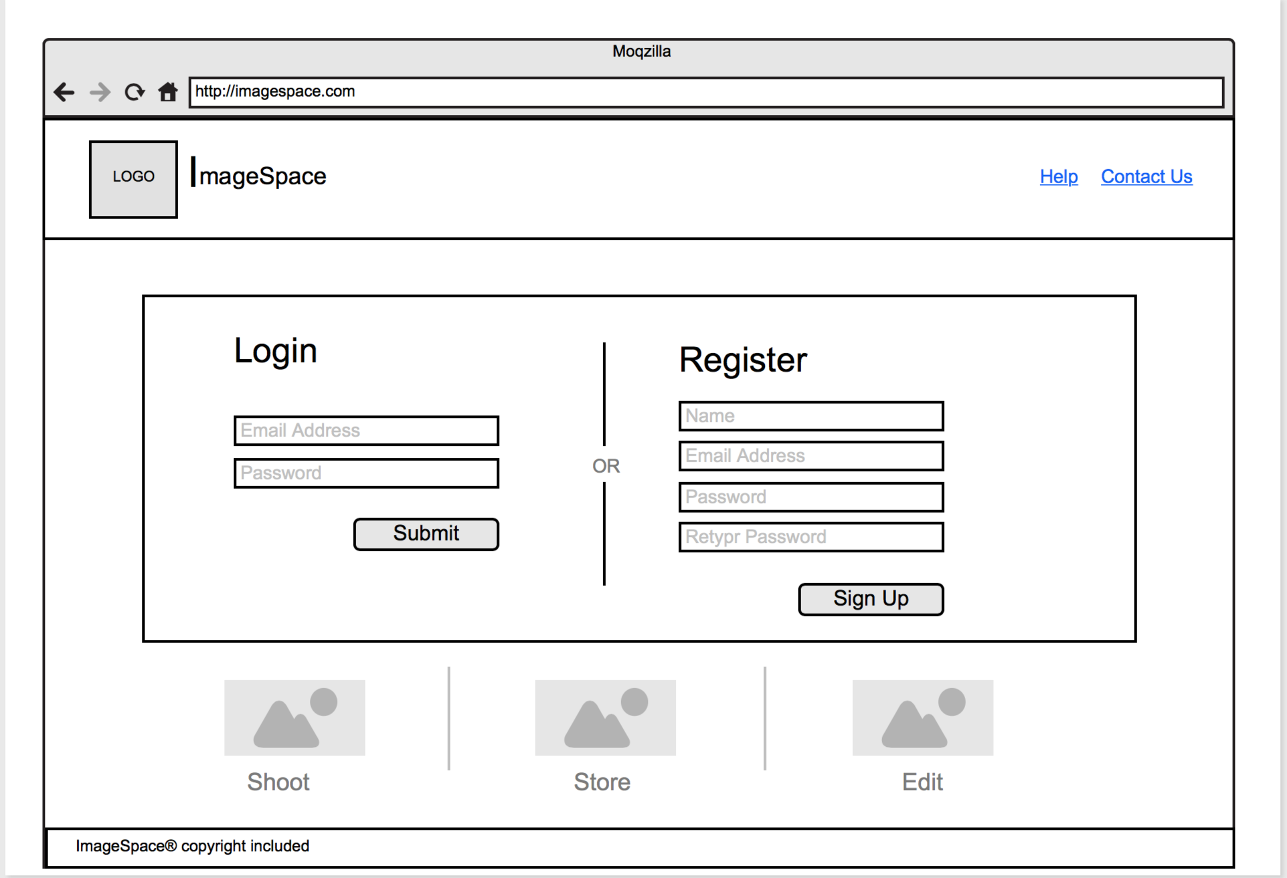
User Interface design & mock ups

Since the final product is a web application for the general public, the usability and user interface intuitiveness has a major role deciding the effectiveness of the solution. As any industrial solution or an open source solution would do, we wanted students to visualize these features in form user interface and web pages. So we asked teams to prepare UI mockups using free or open-sourced mockup tools.

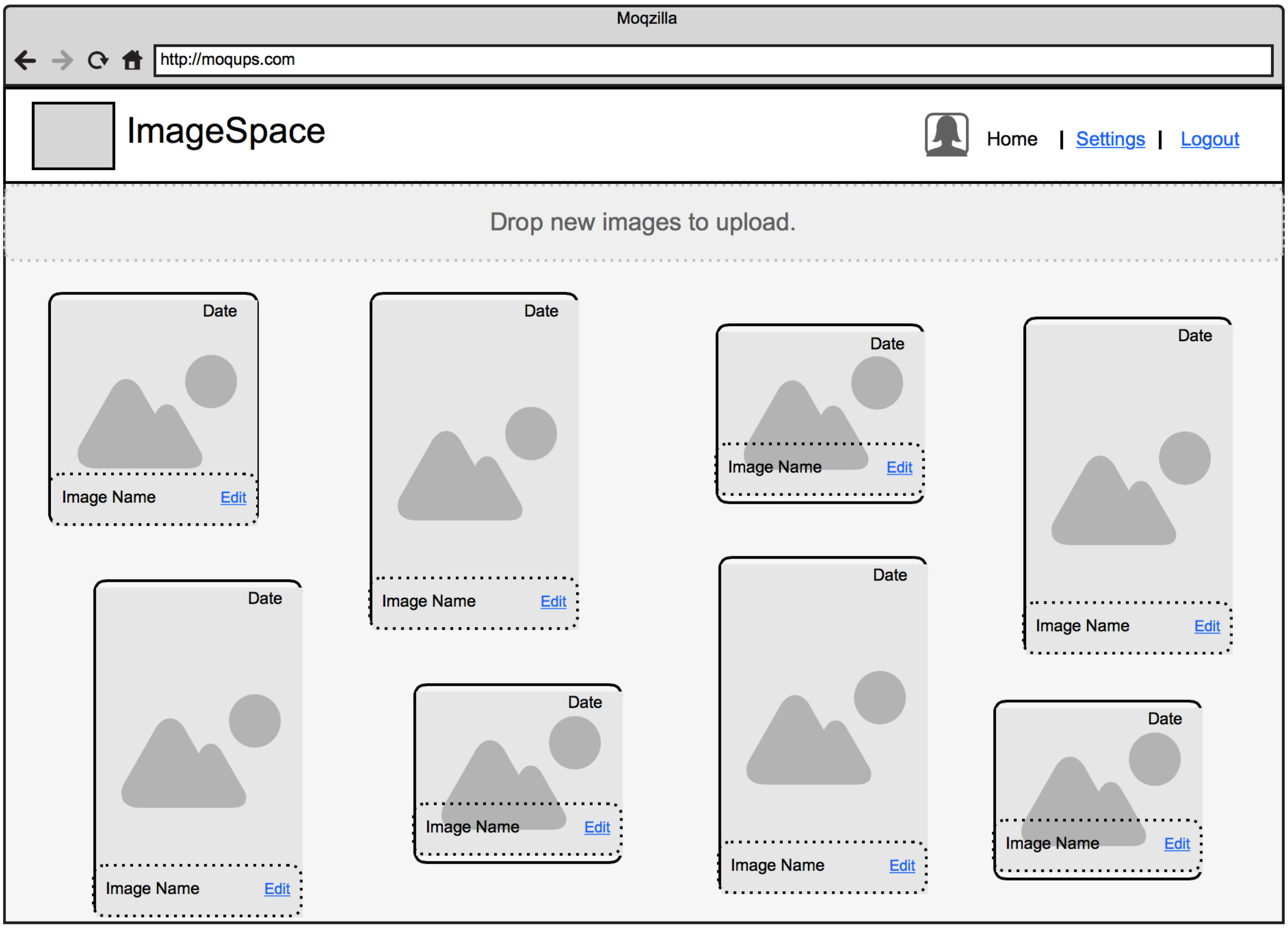
PROBLEM:

Create user interface mockups for proposed web application using the free or open-source mockup tools like Balsamiq and moqups etc. Make sure the UI follows standard web practices and use maximum existing UI elements. Also, keep the usability and intuitiveness of the UI in prime focus, while you design the UI.

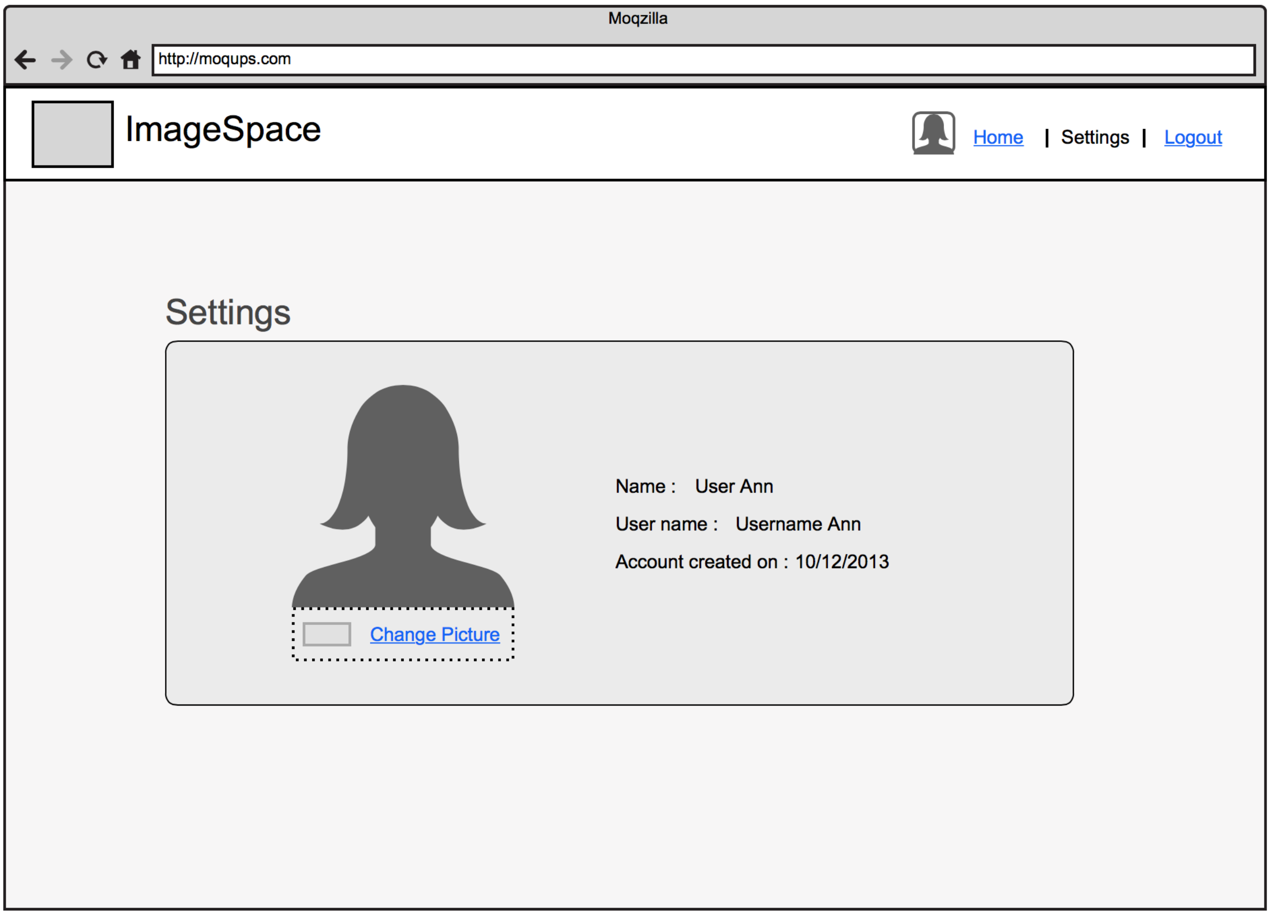
- **Login/Register page**



- **Home page**


Moqzilla
http://moqups.com
ImageSpace
Home
| Settings |
Logout
Drop new images to upload.
Date
Image Name
Edit

- **Settings page**



Settings

Name : User Ann

User name : Username Ann

Account created on : 10/12/2013

REFERENCES

[1] Peng, R., Sun, D., & Tsai, W. (2014). *Understanding Requirements Driven Architecture Evolution in Social Networking SaaS: An Industrial Case Study* (p. 234). IEEE 8th International Symposium on Service Oriented System Engineering.