



Aloha Social Network

Design Document

Document/Version Number: **Version 1.1**
Creation Date: **March 28, 2015**
Account/Client: **SOFTA Inc.**
Author: **Milind Gokhale;Renuka Deshmukh;Mrunal Pagnis;Vivek Supe**
Editor: **Milind Gokhale;Renuka Deshmukh;Mrunal Pagnis;Vivek Supe**
Last Edit Date: **March 28, 2015**
File Name: **Aloha_Design_Document.docx**

Revision History

Date	Description	Author	Comments
03/16/2015	Initial Draft	Vivek Supe, Mrunal Pagnis	Initial draft of the document
03/20/2015	Version 0.1	Milind Gokhale, Renuka Deshmukh	Inserted the basic headings and framework
03/27/2015	Version 1.0	Renuka Deshmukh, Milind Gokhale	First Revision

Document Approval

This section lists all key people in the architecture and design of project Aloha:

Module Name	Printed Name	Title	Date
Friends, Mutual Friends and Suggestions	Milind Gokhale	Project Lead and Developer	March 27, 2015
Posts	Renuka Deshmukh	Database Architect and developer	March 27, 2015
Registration and Profile	Mrunal Pagnis	System Architect and developer	March 27, 2015
Chat	Vivek Supe	Interface Design and developer	March 27, 2015
	Balaprasath Rajan	Professor	March 27, 2015

Table of Contents

Aloha Social Network	0
Design Document	0
Revision History	0
Document Approval	0
Table of Contents	0
1 Introduction	2
1.1 Purpose	2
1.2 Scope	2
1.3 Definitions & acronyms	2
1.4 Intended Audience	2
1.5 Overview	2
2 General Description	3
2.1 Product Perspective	3
2.2 Tools Used	3
2.3 Operating Environment	3
2.4 General Constraints	3
2.5 Assumptions	4
3 Architecture Details	5
3.1 Top level architecture	5
3.2 Application Architecture	5
3.2.1 Web Application Architecture	6
3.2.2 Database Architecture	6
3.3 Use Cases	7
3.3.1 Use Case UC1	7
3.3.2 Use Case UC2	8
3.3.3 Use Case UC3	8
3.3.4 Use Case UC4	8
3.3.5 Use Case UC5	8
3.3.6 Use Case UC6	9
3.3.7 Use Case UC7	9
3.3.8 Use Case UC8	9
3.3.9 Use Case UC9	9

3.3.10	Use Case UC10	10
3.3.11	Use Case UC11	10
3.3.12	Use Case UC12	10
3.4	Class Diagram	11
3.5	Sequence Diagrams.....	12
3.5.1	Registration	12
3.5.2	Manage Account	13
3.5.3	Search Friend	14
3.5.4	<i>Scribble</i>	15
3.5.5	Comment.....	16
3.5.6	<i>ChitChat</i>	17
3.6	User Interface	18
3.7	Deployment Diagram	22
3.8	Coding standards and UI Standards	22
3.9	Error-Handling.....	23
4	Data Model	23
4.1	Overview of Aloha DB	23
4.2	Overview of Aloha Logging DB.....	24
5	Risks and Mitigation Plans	26
5.1	Hardware failure	26
5.2	Software failure.....	26
6	Appendices.....	27
6.1	Appendix 1: Glossary.....	27

1 Introduction

1.1 Purpose

This design document is intended to give the overview of implementation of Aloha at a high level. It also identifies the framework and technologies used for the development and tries to define the system architecture. This document will also be used for identifying contradictions, if any, prior to coding phase. The main goal of this document is to make design level information easily understandable

1.2 Scope

This design document is meant to provide an overview of the structure of the system. This document also includes the database architecture of Aloha along with database diagram for reference. This document also serves as a mandate for the design standards, data structures and design patterns to be implemented. UML diagrams are included to show how they different components interact with each other.

1.3 Definitions & acronyms

The following table explains terms and acronyms specific to this SRS.

Term/Acronym	Description/Definition
Aloha	<i>Aloha is a social networking website.</i>
UC	<i>Use Case</i>
Scribble	<i>Scribble is a post shared by the user on his slate.</i>
ChitChat	<i>Chatting activity between two users registered on Aloha.</i>
Slate	<i>Wall where users scribble.</i>

1.4 Intended Audience

This document is intended for a varied set of audiences including Product manager, team lead, stakeholders, developers, QA, system architects, DBA, deployment engineers.

1.5 Overview

The remaining document has 5 more sections. The second section gives general description of the project. Section 3 gives architectural view of the system. Section 4 consists of the data model. Section 5 gives the consideration of any risks if occurred and their mitigations. Section 6 is a list of appendix that helps better understand the document.

2 General Description

2.1 Product Perspective

Aloha is a free online social networking website which allows users to connect with their friends and family. The product will consist of different modules namely User registration and setting module, *ChitChat* module, Friends and suggestions module and *Scribbles* module. The user interface will comprise of JSP pages. User Registration module requires user to fill-in his data. Re-Captcha and e-mail verification are included as an additional security measure.

ChitChat will be implemented using peer-to-peer architecture. *ChitChat* will also support File-sharing. Friends and Suggestions module will enable persistent storage of user connections while suggestions will be implemented using an efficient data structure. Database will be implemented using MySQL Community Server 5.x.

2.2 Tools Used

1. Tool for drawing Diagrams – Dia is used to create all the UML diagrams of Aloha during the design phase.
2. Spring Tool Suite – The Spring Tool Suite™ (STS) will be the development environment for implementing Aloha. Aloha will be developed using Spring MVC framework.
3. [<http://stackoverflow.com/tags/spring-tool-suite/info>]
4. MySQL Community Server 5.x – MySQL community server will be used as a relational database backend.
5. Apache Tomcat 7 – Apache Tomcat server will be the server environment to host the java code and JSP pages.
6. Java – JDK 1.8 will be the java development kit used for implementing Aloha.
7. Git (<https://git.iu.edu>) – GIT repository will be used for version control and tracking defect.
8. Secure Assist – Cigital SecureAssist Eclipse plugin will be used to find security vulnerabilities in Aloha. [<http://marketplace.eclipse.org/content/cigital-secureassist>]
9. Dreamweaver – HTML editor to design JSP pages.
10. CoffeeCup - HTML editor to design JSP pages.
11. Windows 7/8 OS –The development platform will be Windows 7/8 OS.

2.3 Operating Environment

- Minimum: 2 CPU Cores (Recommended: 4 CPU Cores or more)
- Recommended memory: 2 GB for Windows platforms, 1 GB for Linux
- Minimum disk space: 500 MB
- Recommended disk space: 1024 MB
- Java Platform Enterprise Edition (Java EE)
- Apache 8.x
- JDK 6+ for Spring Framework
- MySQL 5.x.x

2.4 General Constraints

- Aloha will support the following browsers – IE 8+, Chrome 27.0+, and Firefox 30+.

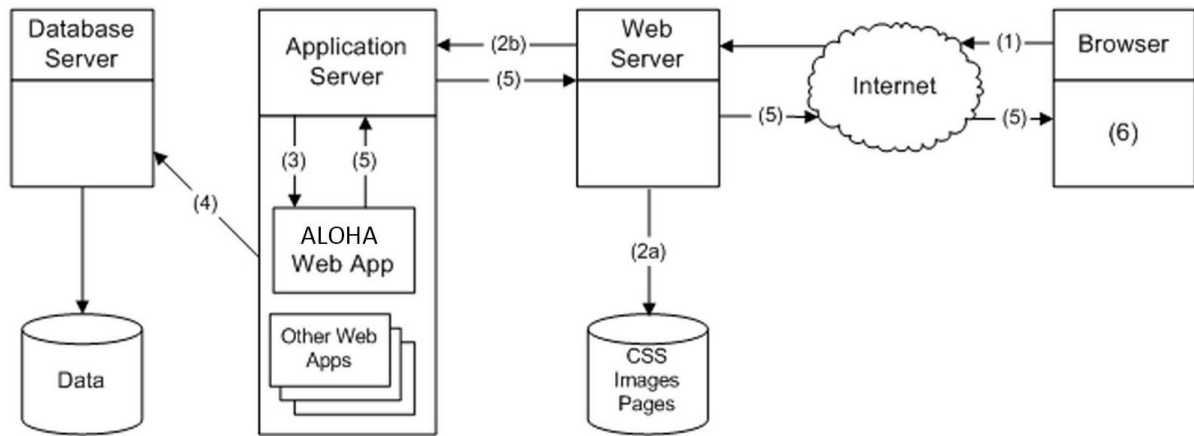
- Aloha supports English language only.
- Aloha requires certificates issued by CA to use HTTPS.
- The first version of the website is intended to be viewed best on a desktop browser.

2.5 Assumptions

1. There will be availability of Internet via 3G, 4G or Wi-Fi.
2. Central server of the system must be able to handle all the incoming requests simultaneously.
3. The developer machines will be up and available till the end of the project.
4. All the necessary product licenses will have been acquired.
5. Requirement engineering has been carried out correctly and thoroughly.
6. On implementation the modules of the system will work coherently.

3 Architecture Details

3.1 Top level architecture



ALOHA

Figure 1: Aloha Top Level Architecture Diagram

The above diagram gives a block view of Aloha system showing the interaction of different modules with each other. The numbered arrows show the flow of control between these modules.

3.2 Application Architecture

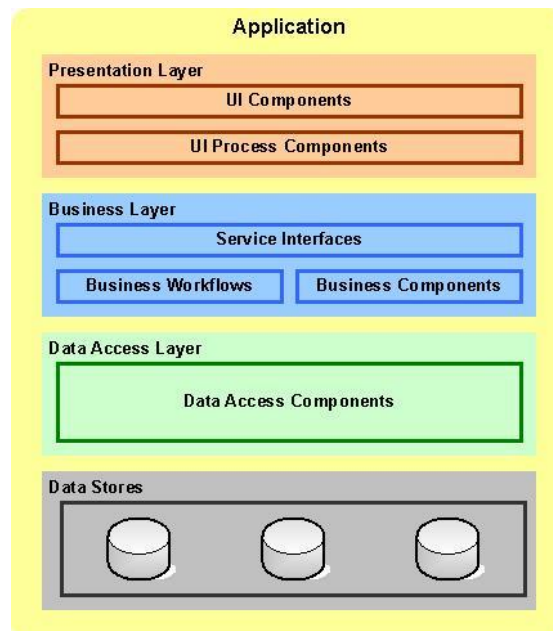


Figure 2: Application Architecture Stack

3.2.1 Web Application Architecture

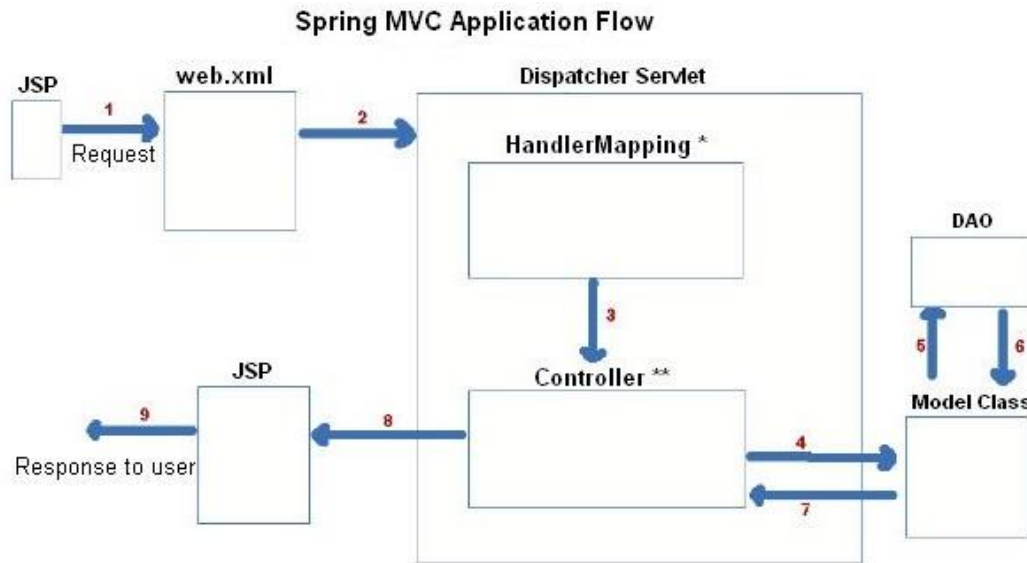


Figure 3: Aloha Web Application Architecture Diagram

Spring MVC framework will be used for developing the Aloha web application. It divides the application into three interconnected parts. Thus the internal representations of the information are kept separate from the information that is presented to the end-user or accepted from end-user.

1. Presentation Layer

The presentation layer of the application will be JSP pages displayed to the user. The JSP pages will be both to display information and to take inputs from user.

2. Controller Layer

The controller initiates the commands to the model to make changes to database. It also sends commands to the presentation layer views associated to the models.

3. Data Access Layer

The data access layer is used to access the database and make changes to the data.

3.2.2 Database Architecture

This layer consists of relational database implemented on MySQL server. For the detailed database architecture refer section for Data Model (section 4).

3.3 Use Cases

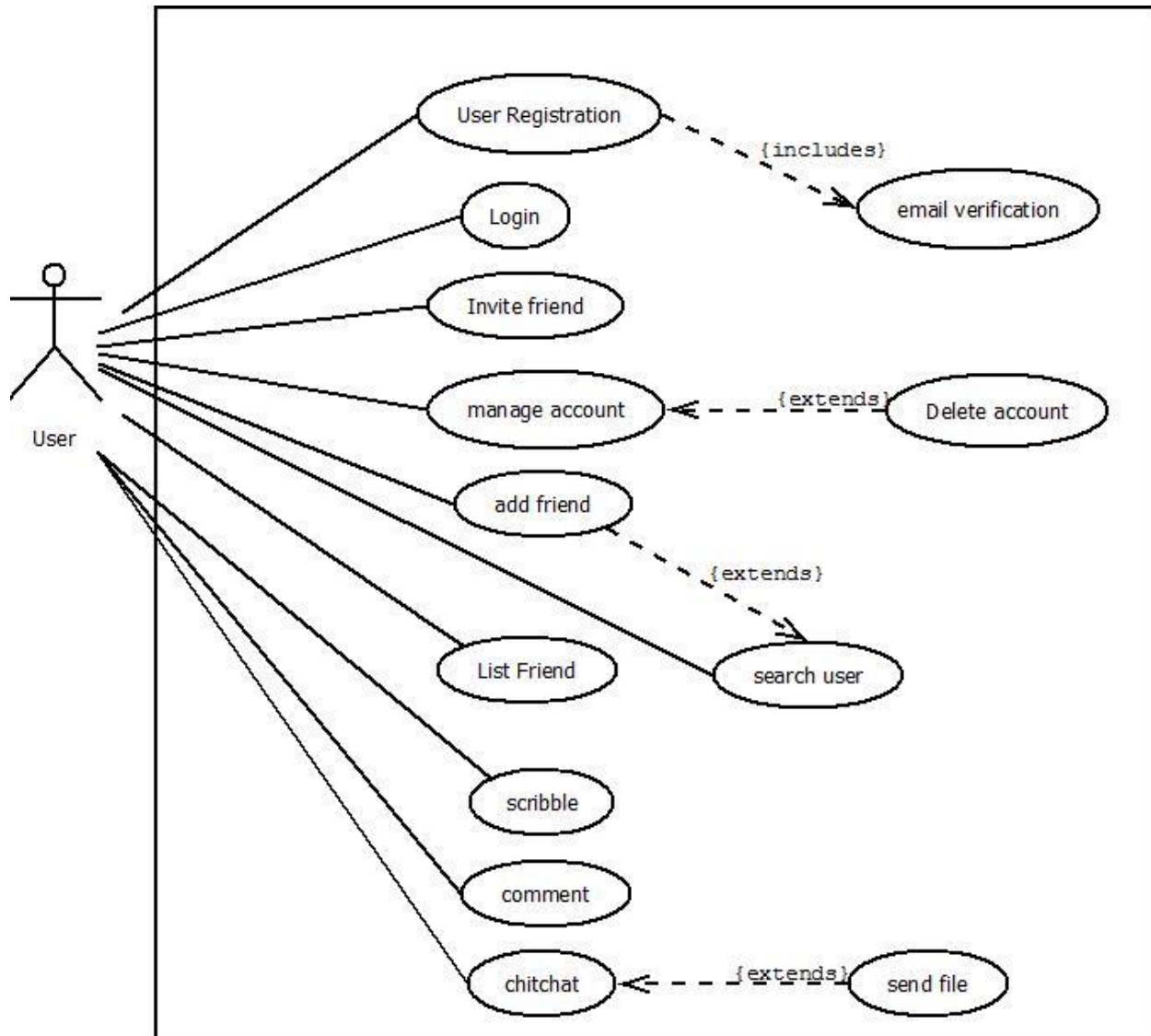


Figure 4: Use Case Diagram

3.3.1 Use Case UC1

3.3.1.1 Objective

Registration

3.3.1.2 Priority

High

3.3.1.3 Actors

End-User

3.3.1.4 Pre-conditions

The user has internet and opens the website on his browser.

3.3.1.5 Post-conditions

The user is now a member of Aloha website.

3.3.2 Use Case UC2

3.3.2.1 *Objective*

Email Verification

3.3.2.2 *Priority*

High

3.3.2.3 *Actors*

End-User

3.3.2.4 *Pre-conditions*

The user is registering to aloha website as a new user.

3.3.2.5 *Post-conditions*

The user is now a registered user on Aloha and can update account, post scribbles, add friends and chat with other registered friends.

3.3.3 Use Case UC3

3.3.3.1 *Objective*

Login

3.3.3.2 *Priority*

High

3.3.3.3 *Actors*

End-User

3.3.3.4 *Pre-conditions*

User is a registered member of Aloha.

3.3.3.5 *Post-conditions*

The user logs in successfully and can now view his profile, friends and scribbles.

3.3.4 Use Case UC4

3.3.4.1 *Objective*

Manage Account

3.3.4.2 *Priority*

Medium

3.3.4.3 *Actors*

End-User

3.3.4.4 *Pre-conditions*

A user is logged into Aloha.

3.3.4.5 *Post-conditions*

The user account has been successfully updated as per his requirements, like, update account, visibility setting, etc.

3.3.5 Use Case UC5

3.3.5.1 *Objective*

Delete Account

3.3.5.2 *Priority*

Medium

3.3.5.3 *Actors*

End-User

3.3.5.4 *Pre-conditions*

User is a registered member of Aloha.

3.3.5.5 *Post-conditions*

Account is successfully deleted.

3.3.6 Use Case UC6

3.3.6.1 *Objective*

Add Friend

3.3.6.2 *Priority*

High

3.3.6.3 *Actors*

End-User

3.3.6.4 *Pre-conditions*

User is a registered member of Aloha.

3.3.6.5 *Post-conditions*

Friend is added successfully.

3.3.7 Use Case UC7

3.3.7.1 *Objective*

List Friends

3.3.7.2 *Priority*

High

3.3.7.3 *Actors*

End-User

3.3.7.4 *Pre-conditions*

User is a registered member of Aloha.

3.3.7.5 *Post-conditions*

User sees a list of his friends.

3.3.8 Use Case UC8

3.3.8.1 *Objective*

Invite Friend

3.3.8.2 *Priority*

Low

3.3.8.3 *Actors*

End-User

3.3.8.4 *Pre-conditions*

User is a registered member of Aloha.

3.3.8.5 *Post-conditions*

Friend invite is sent successfully on his email allowing him to register as a new user.

3.3.9 Use Case UC9

3.3.9.1 *Objective*

Scribble

3.3.9.2 *Priority*

High

3.3.9.3 *Actors*

End-User

3.3.9.4 *Pre-conditions*

User is a registered member of Aloha.

3.3.9.5 *Post-conditions*

Posted scribbles are visible to the user and his friends.

3.3.10 Use Case UC10

3.3.10.1 *Objective*

Comment

3.3.10.2 *Priority*

High

3.3.10.3 *Actors*

End-User

3.3.10.4 *Pre-conditions*

User is a registered member of Aloha.

3.3.10.5 *Post-conditions*

Posted comments are visible to the user and his friends.

3.3.11 Use Case UC11

3.3.11.1 *Objective*

ChitChat

3.3.11.2 *Priority*

High

3.3.11.3 *Actors*

End-User

3.3.11.4 *Pre-conditions*

User is a registered member of Aloha.

3.3.11.5 *Post-conditions*

The user successfully performs direct chitchat with his friend.

3.3.12 Use Case UC12

3.3.12.1 *Objective*

Send File

3.3.12.2 *Priority*

High

3.3.12.3 *Actors*

End-User

3.3.12.4 *Pre-conditions*

User is a registered member of Aloha.

3.3.12.5 *Post-conditions*

The second user successfully receives the send file.

3.4 Class Diagram

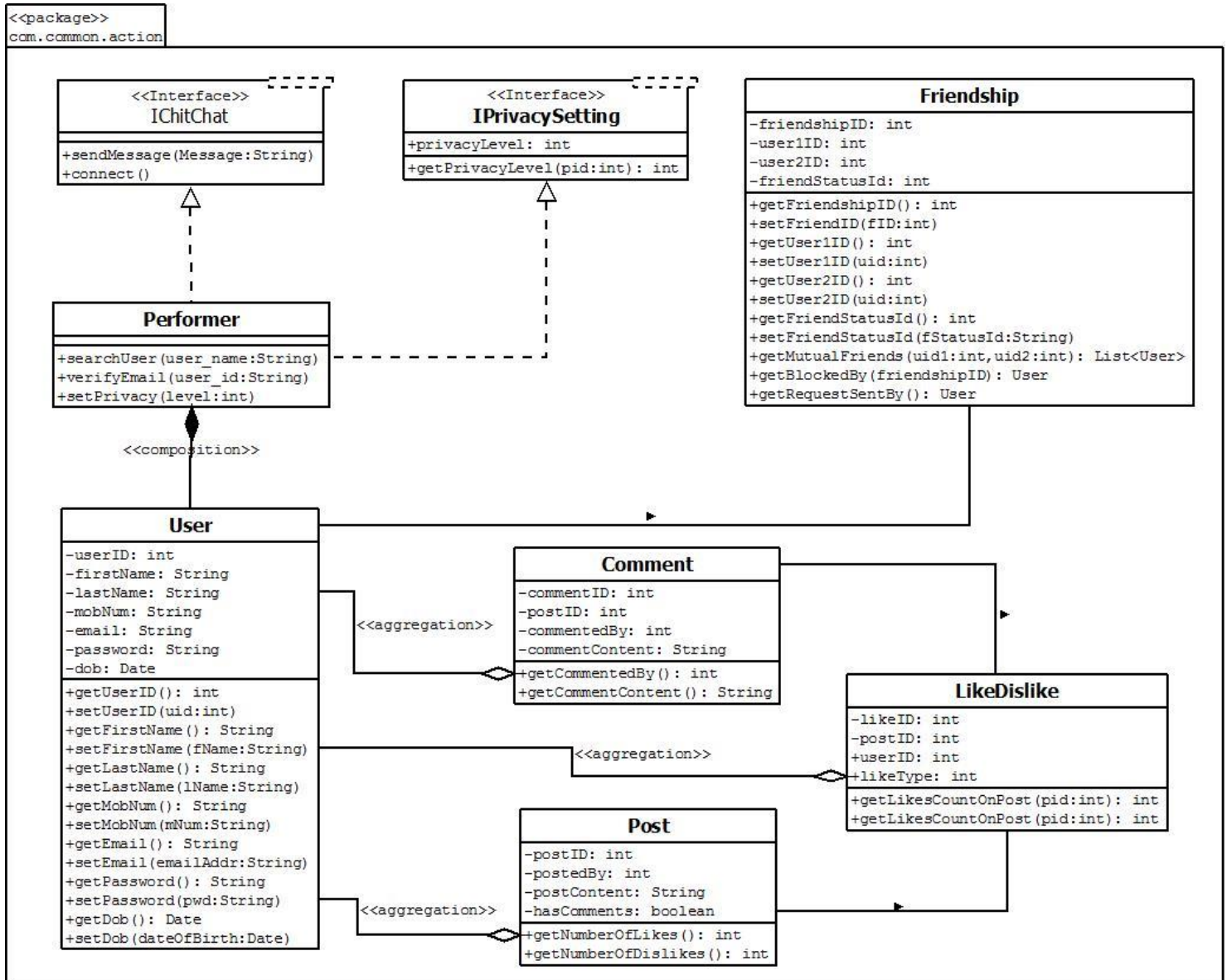


Figure 5: Class Diagram for Aloha

3.5 Sequence Diagrams

3.5.1 Registration

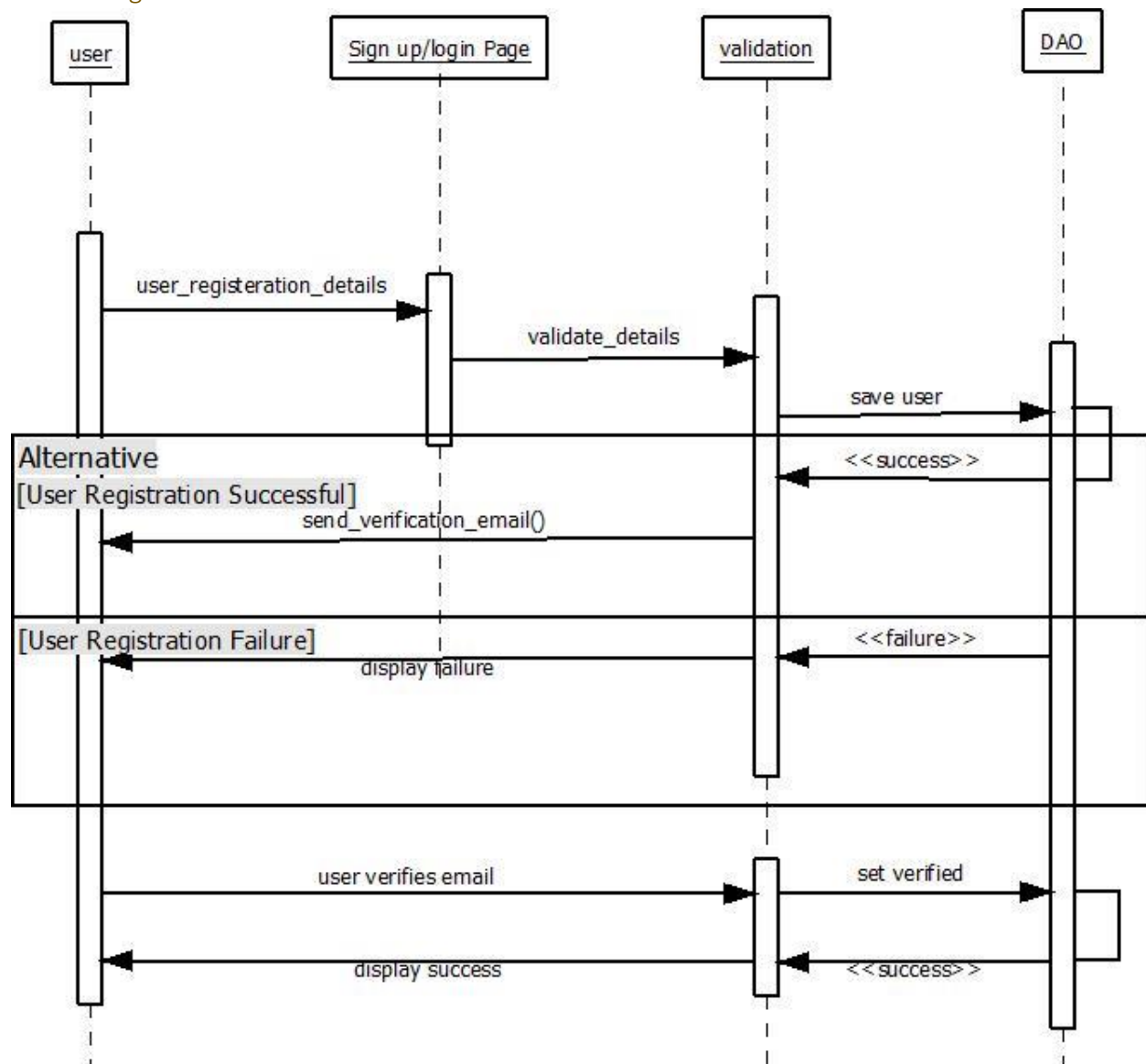


Figure 6: Registration Sequence Diagram

3.5.2 Manage Account

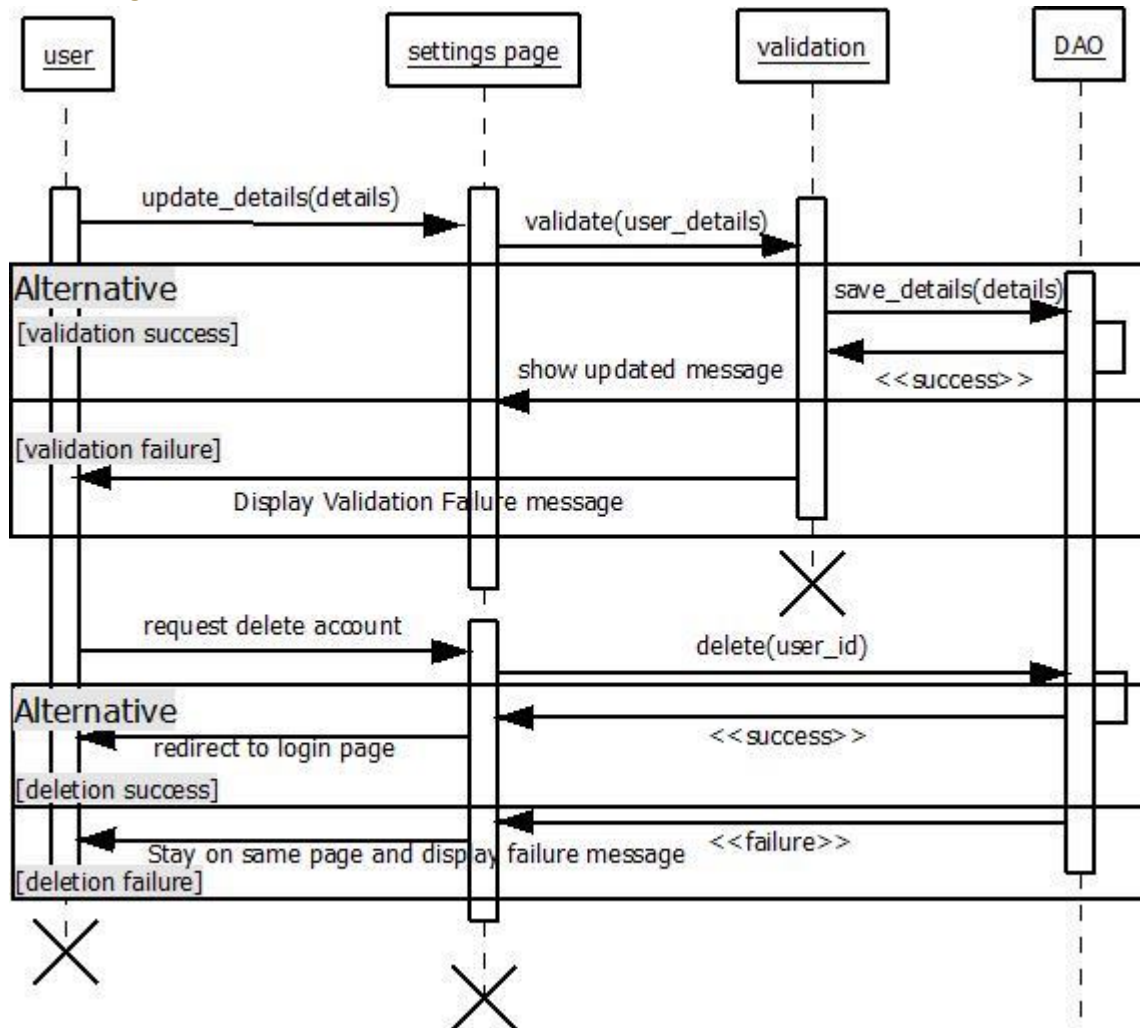


Figure 7: Manage Account Sequence Diagram

3.5.3 Search Friend

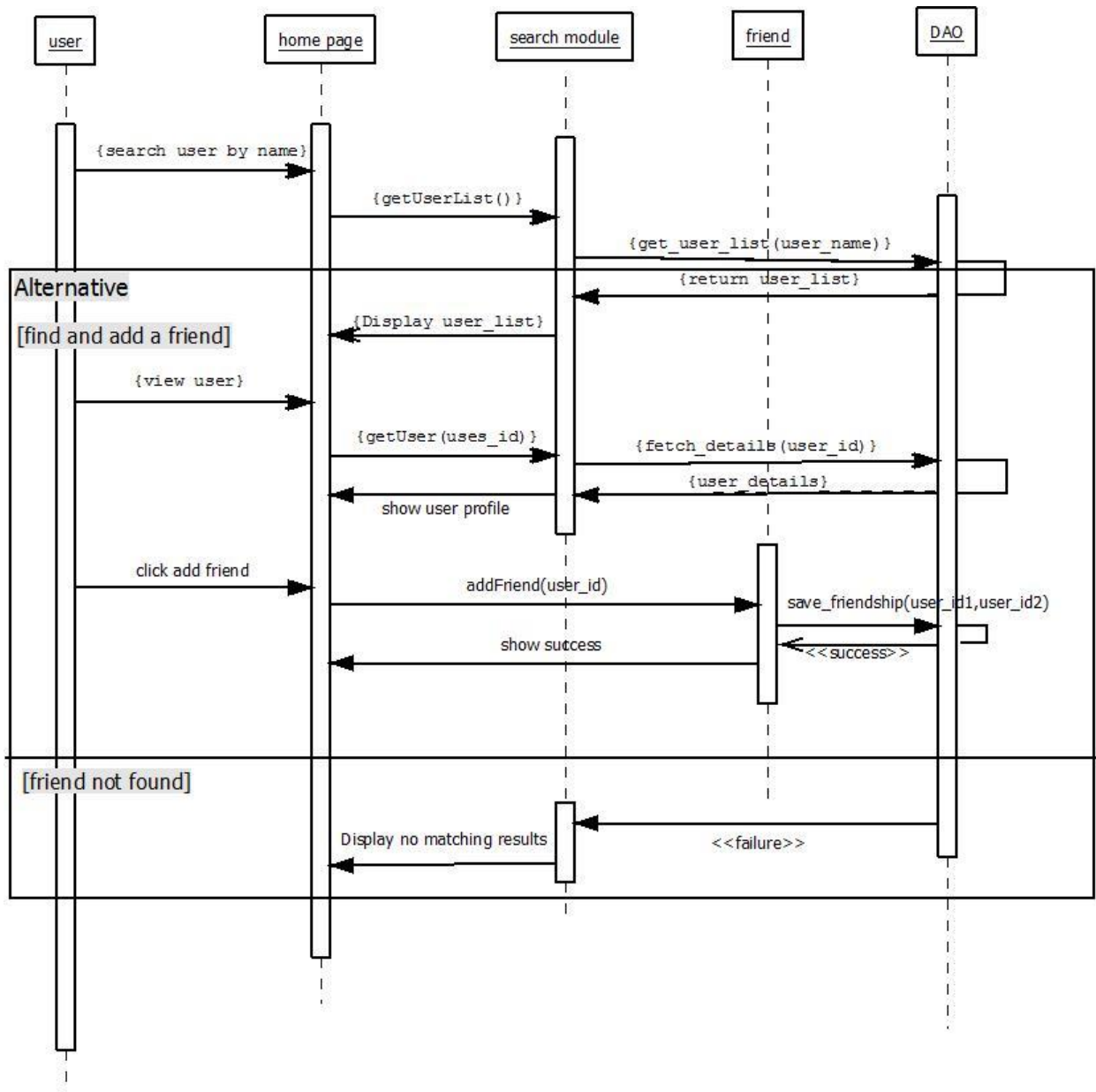


Figure 8: Search Friend Sequence Diagram

3.5.4 Scribble

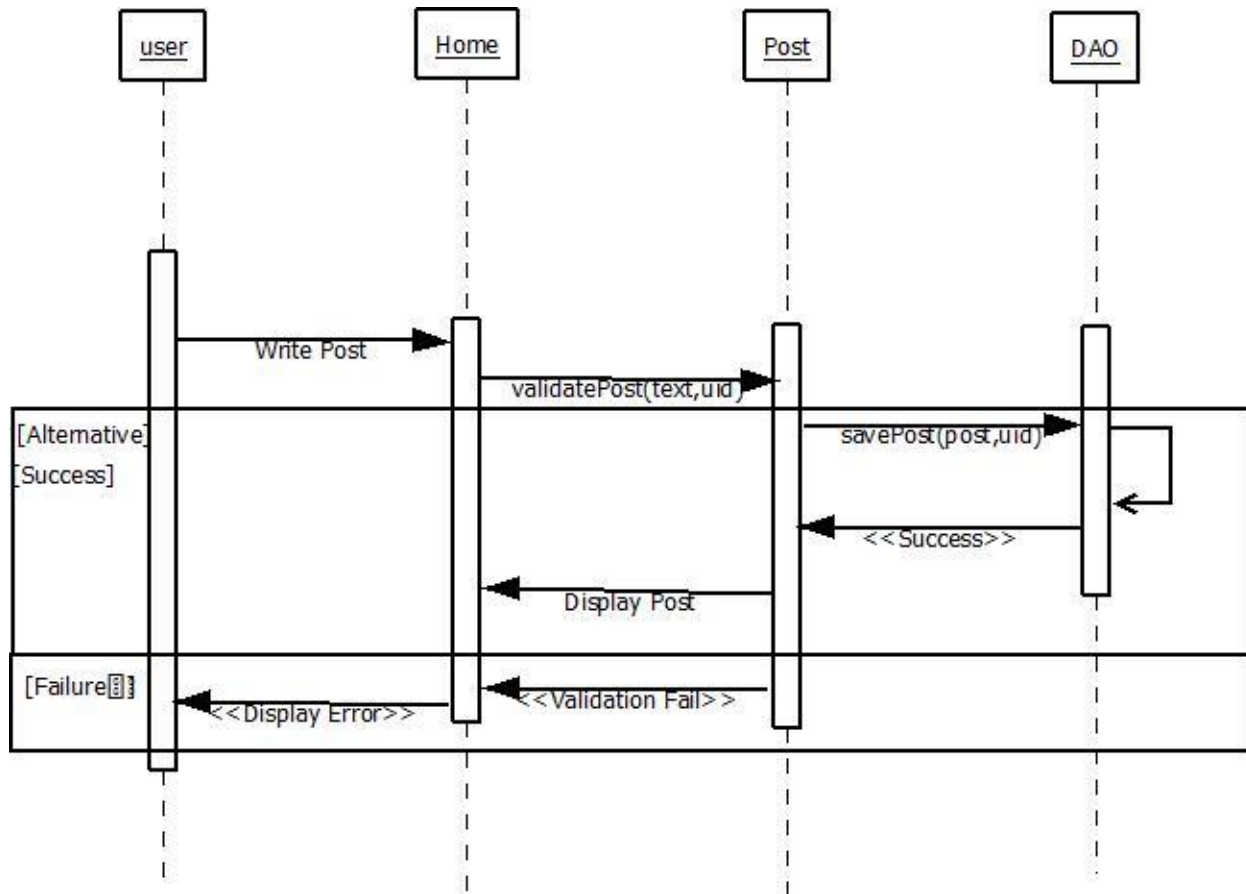


Figure 9: Scribble Sequence Diagram

3.5.5 Comment

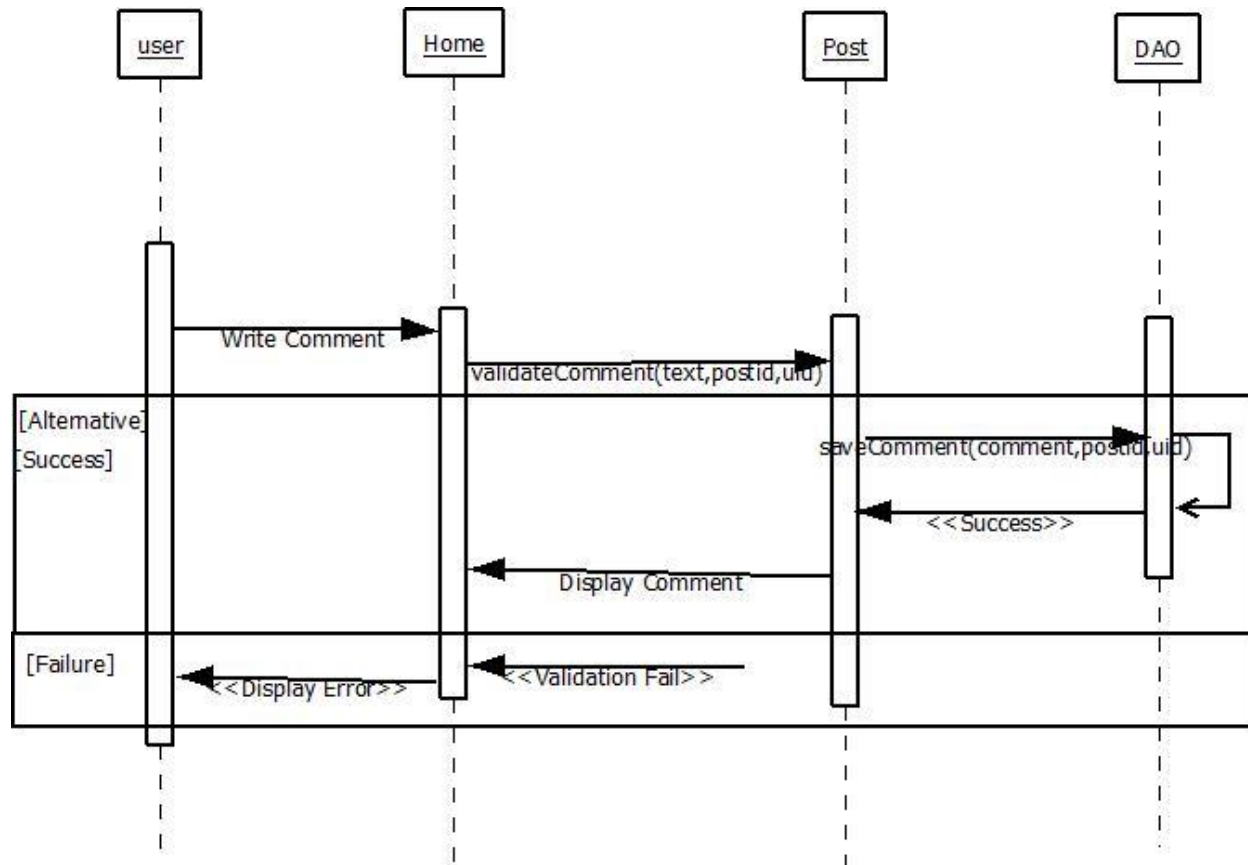


Figure 10: Comment Sequence Diagram

3.5.6 ChitChat

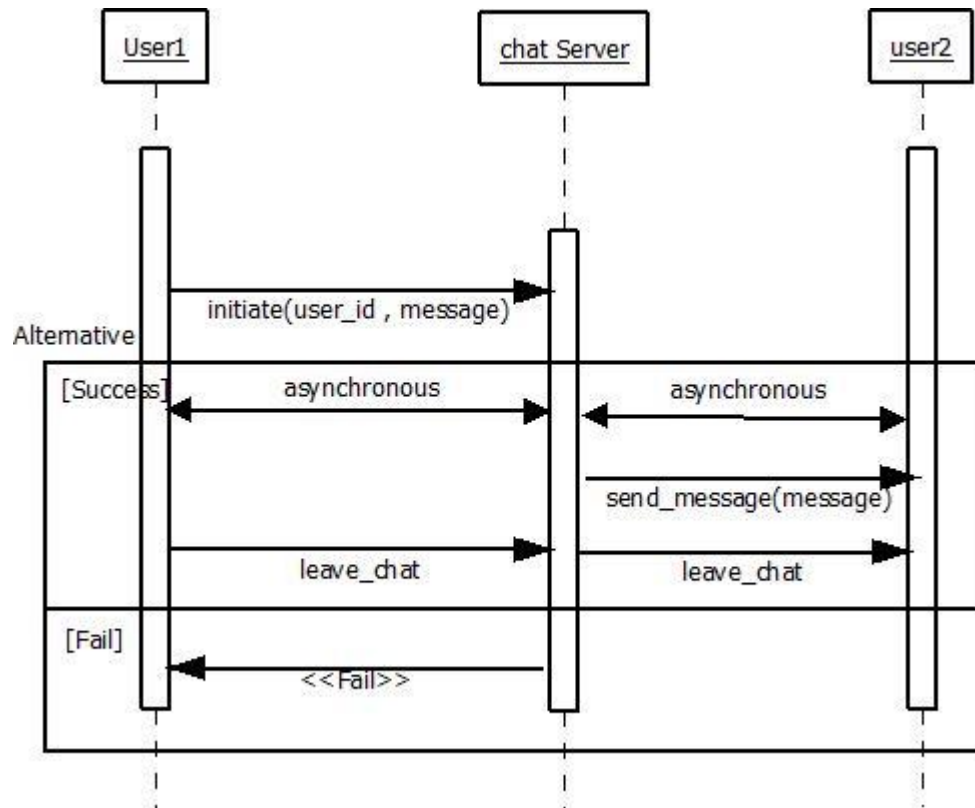


Figure 11: ChitChat Sequence Diagram

3.6 User Interface

This section shows some of the UI interfaces visible to the user.

The screenshot shows a web browser window titled "Aloha...get connected" with the address bar displaying "https://aloha.com". The page has a "Home" link in the top navigation bar. The main content area is divided into two columns. The left column is titled "Sign Up" and contains the following fields and controls: "Name:" with the value "Mark Dane", "Email:" with the value "MarkD@cinergix.com", "Password:" with masked characters "*****", "Confirm:" with masked characters "*****", a "Remember me" checkbox, a CAPTCHA image showing "eP3 text my" with a "replay" button, a "captcha text" input field, a "Term and Conditions" checkbox, and a "Sign Up" button. The right column is titled "Sign In" and contains the following fields and controls: "User Name:" with the value "email", "Password:" with masked characters "*****", a "Sign In" button, and a "New User" section with a "Sign Up" button.

Figure 12: Sign In

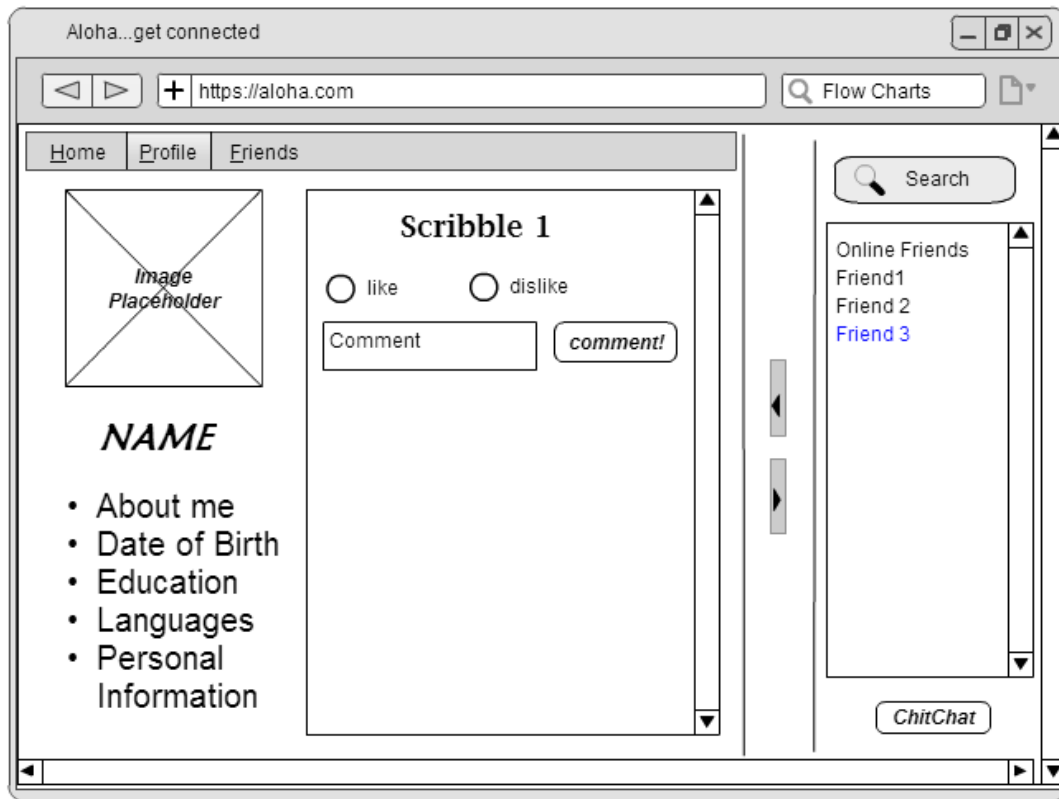


Figure 13: Profile

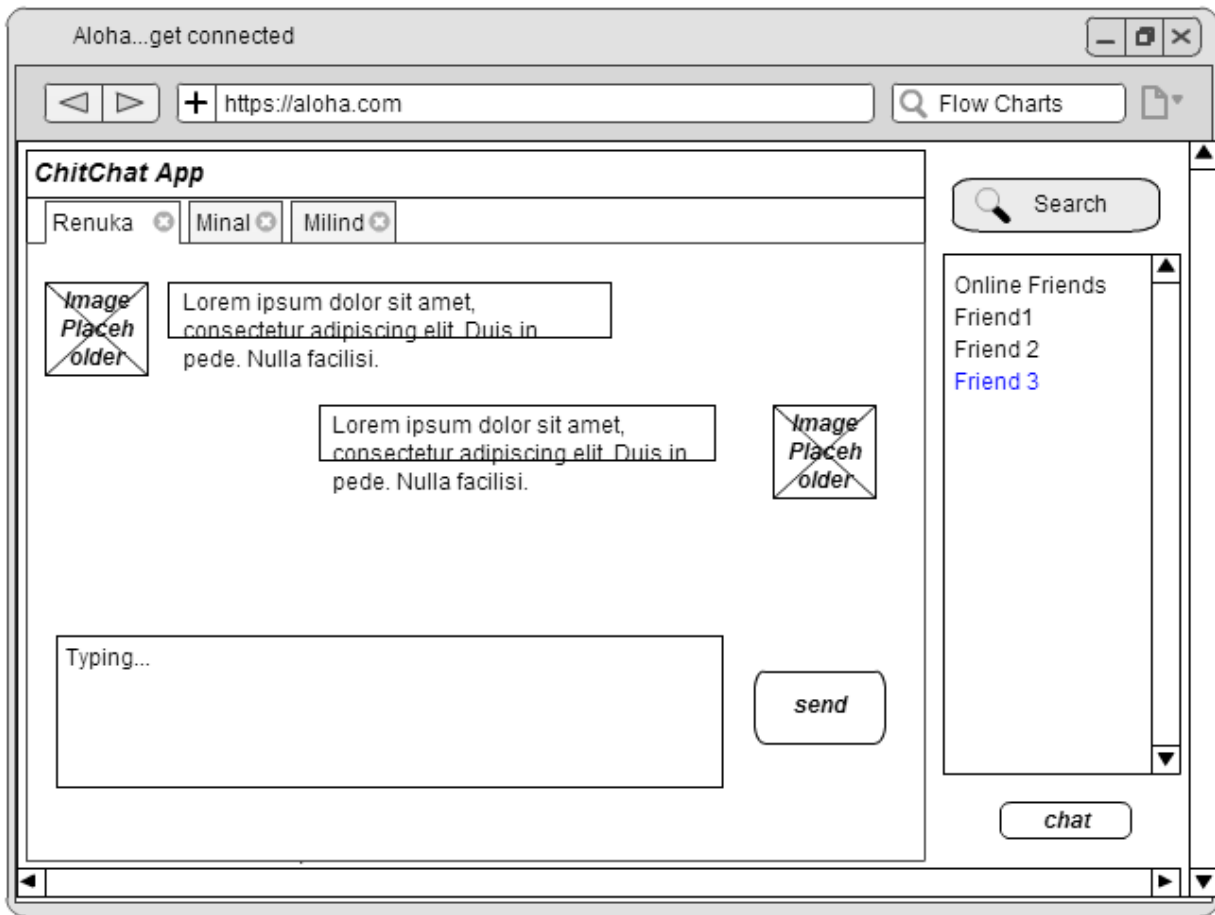


Figure 14: Chat

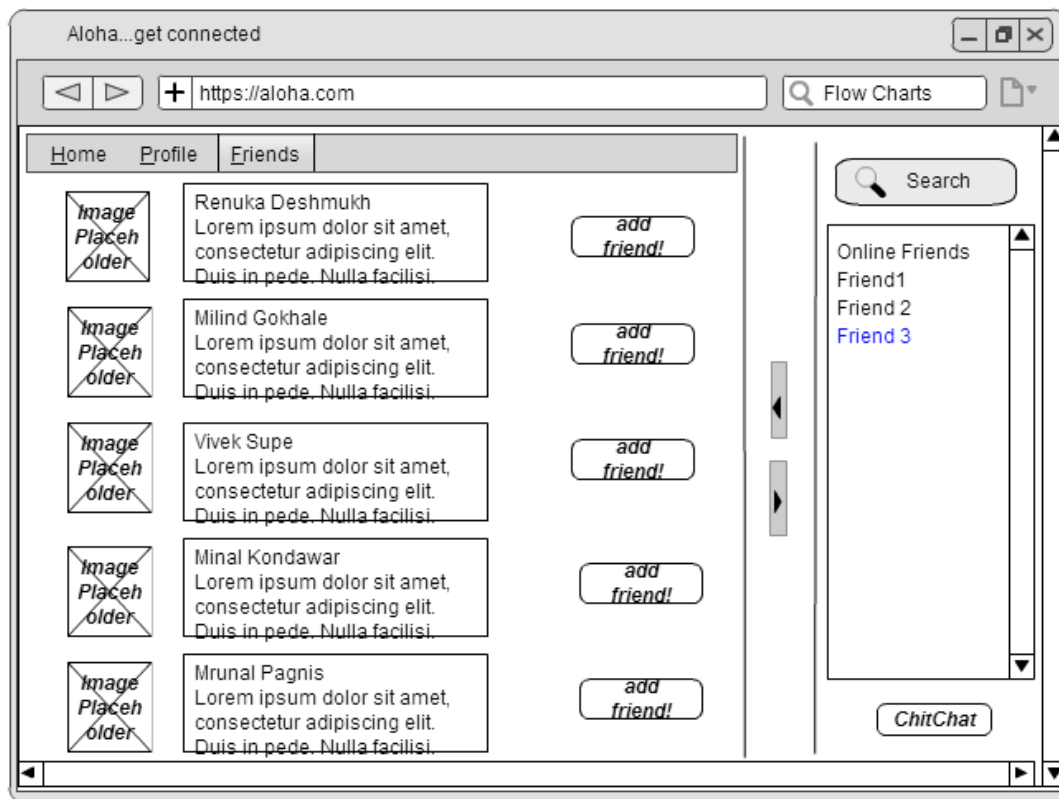


Figure 15: Friends

3.7 Deployment Diagram

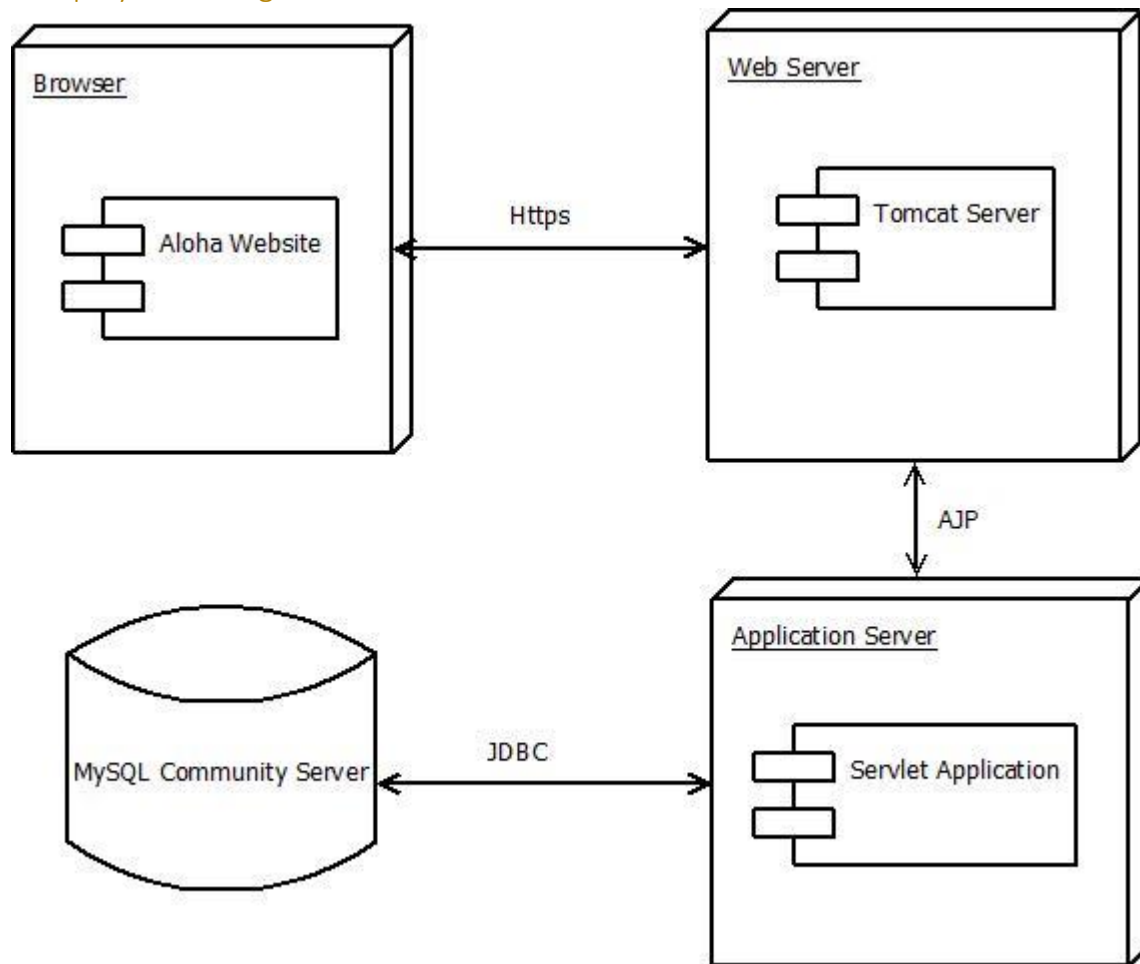


Figure 16: Aloha Deployment Diagram

3.8 Coding standards and UI Standards

Following standards must be strictly followed during the development and implementation phase of project Aloha:

1. Follow HTML5, CSS3, and JavaScript coding guidelines.
2. Markup should be well formed, semantically correct and generally valid.
3. All markups should be delivered as UTF8. It should be mentioned in the document head and the HTTP header.
4. JavaScript file should be well commented.
5. Use proper validation of user input on client and server-side.
6. Sun's JavaBeans coding and naming conventions should be followed.
7. Java code should be well commented.
8. Proper logging and exception handling should be implemented.

3.9 Error-Handling

To help the developer understand the different operations being done by the application. The application will have a database logging which will have crucial logs written to the database and file logging will have all events logs which may be required in the future. We will be using log4j which will log entries in 5 levels of priority so that we can fetch logs as per the business requirement.

4 Data Model

4.1 Overview of Aloha DB

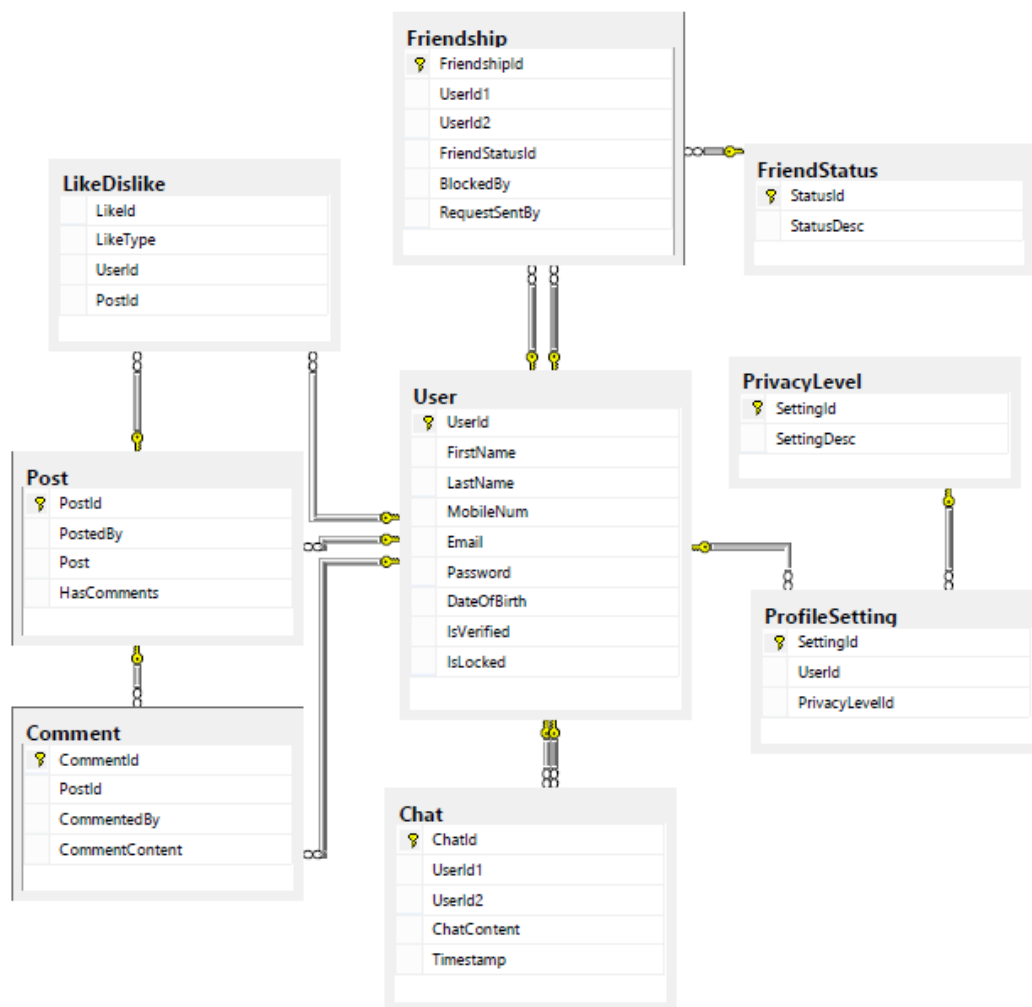


Figure 17: Aloha DB Model

Table Name	Description
User	This table consists of the user profile data
Friendship	This table consists of all the relations between all the users.

FriendStatus	This is a reference table to track the status between users in friendship table. It can have values like "Accept/Pending", "Blocked", etc.
Chat	This table saves <i>chitchat</i> history between two users.
Post	This table contains all the <i>scribbles</i> posted by users of Aloha.
Comment	This table contains all the comments to all the <i>scribbles</i> posted on Aloha.
ProfileSetting	This table contains the privacy level setting for each user.
LikeDislike	This table manages the likes and dislikes of each <i>scribble</i> .
PrivacyLevel	This is a reference table which contains all the possible privacy levels of a user in Aloha.

4.2 Overview of Aloha Logging DB

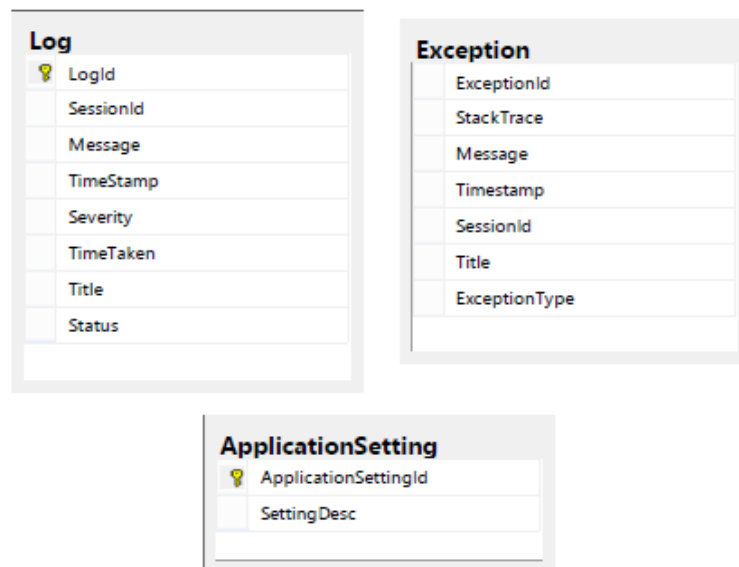


Figure 18: Aloha DB logging Model

Table Name	Description
Log	This table contains all the logs for all the activities in Aloha. It will be used for tracing bugs and auditing.
Exception	This table contains all the exceptions thrown by the application in the various flows in Aloha. It will be used for tracing bugs and auditing.
ApplicationSetting	This table contains settings to enable to disable logging in database.

5 Risks and Mitigation Plans

5.1 Hardware failure

- After deployment the development server fails
 - Use one of the operational servers if urgent until problem is fixed
- One of the operational servers fails
 - The two deployment servers will be the replacements of each other, use other.
- Any of the development workstations fail
 - There are 2 spare workstations in case of workstation failure.

5.2 Software failure

- Spring Tools Suite crashes or fails
 - Re-install a clean copy of STS on the development machine.
- Tomcat Server Installation is corrupted.
 - Re-install tomcat package and server.

6 Appendices

6.1 Appendix 1: Glossary

<i>Term/Acronym</i>	<i>Description/Definition</i>
MVC	<i>Model View Controller</i>
HTTP	<i>Hyper Text Transfer Protocol</i>
HTTPS	<i>Hyper Text Transfer Protocol over SSL</i>
CAPTCHA	<i>Completely Automated Public Turing test to tell Computers and Humans Apart</i>
DAO	<i>Data Access Layer</i>
Tomcat	<i>Apache Tomcat Web Server</i>
IE	<i>Internet Explorer</i>
OS	<i>Operating System</i>
JDK	<i>Java Development Kit</i>
CA	<i>Certificate Authority</i>
STS	<i>Spring Tool Suite</i>
QA	<i>Quality Assurance</i>
DBA	<i>Database Administrator</i>

References:

1. "Code Standards" Front-end. Accessed March 25, 2015.
<http://standards.findsubstance.com/frontend/>.
2. "Online Diagram Software to Draw Flowcharts, UML & More" Creately. Accessed March 20, 2015. <http://creately.com/>.
3. Gokhale, Milind, Renuka Deshmukh, Vivek Supe, and Mrunal Pagnis. "SRS_Aloha_My_Group" February 1, 2015. Accessed March 28, 2015.
https://iu.app.box.com/files/0/f/3167866963/1/f_26849929211.
4. "Spring MVC Application Flow" Accessed March 26, 2015. <http://www.goospoos.com/wp-content/uploads/2009/12/Spring24.jpg>.
5. "High Level Design Distributed Network Traffic Controller." Accessed March 28, 2015.
https://people.ok.ubc.ca/rlawrenc/research/Students/CJ_05_Design.pdf.