Logical ArchitecturE Document

Version 1.1

**Project Title:** **CHAT APPLICATION**

**Team Members:** Christopher Cargile

Gayathri Parthasarathy

Nicklaus Rhodes

Jonathon Ng

**VERSION HISTORY**

|  |  |
| --- | --- |
| **Version#** | **Revision Date** |
| 1.0 | 03/12/2014 |
| 1.1 | 04/19/2014 |

**Table of Contents Page No**

Introduction 3

Component Diagram 3

Class Diagrams 4

Sequence Diagrams 6

State Transition Diagrams 10

Tools Used to Create Diagrams 12

**Introduction:**

* 1. Purpose of Logical Architecture

This Logical Architecture is designed to identify the Chat Application Components (and their dependencies) that provide software services to meet the goal of the project.

Models will include:

1. Component and Connector Diagrams
2. Class Diagrams
3. Sequence Diagrams
4. State Transition Diagrams

This document will provide models of different types to provide a visual output of the software services to the stakeholders of the project.

The intended audience of the Chat Application Logical Architecture is all project stakeholders including the project team members, CSCI 656 classmates and Professor Rudolph.

For information regarding the Interface Controls, please refer to the Interface Control Document of the Chat Application Project documentation.

**Component and Connector Diagrams**

This section will show the component and connectors being used in the Chat Application program.

The components in Figure 1 include:

1. Client1
2. Client 2
3. Server
4. MySQL Database

The connectors in Figure 1 include:

1. JDBC
2. REST
3. Java Socket API

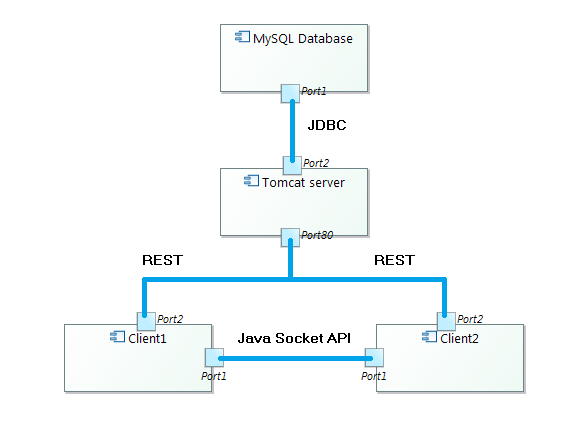


Figure 1.Chat Application – Components and Connectors

**Class Diagrams**

This section will provide Unified Modeling Language (UML) class diagrams to provide a static structure diagram of the Chat Application program.

Diagrams provided include:

1. The peer to peer connector class diagram shown in Figure 2 – details complete peer to peer implementation of the system
2. Client Authentication Class Diagram shown in Figure 3 – Details the authentication system which involves server and database transaction.

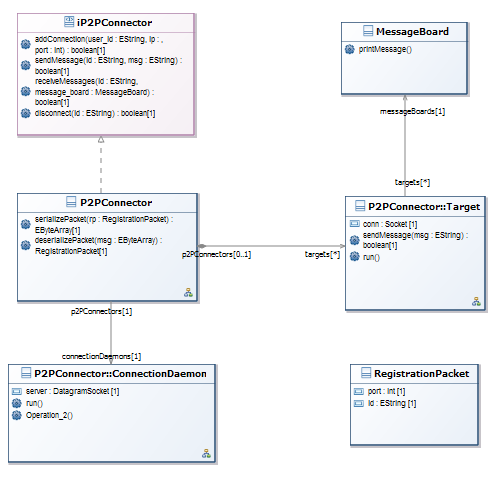


Figure 2.Peer to Peer Connector: Class Diagram

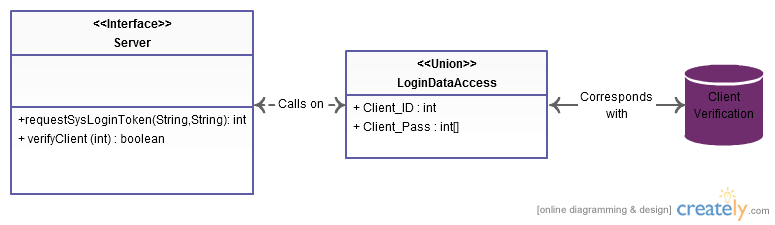


Figure 3.Client Authentication Class Diagram

**Sequence Diagrams**

This section will provide the interactions of processes which operate with one another and in what order.

Sequence diagrams will include:

1. Add new connection in a peer to peer connector shown in Figure 4.
2. Login Sequence Diagram shown in Figure 5.
3. Registration Sequence diagram shown in Figure 6.
4. Log Off Sequence Diagram shown in Figure 7.
5. Send Chat Message shown in Figure 8.
6. Refresh User List shown in Figure 9.

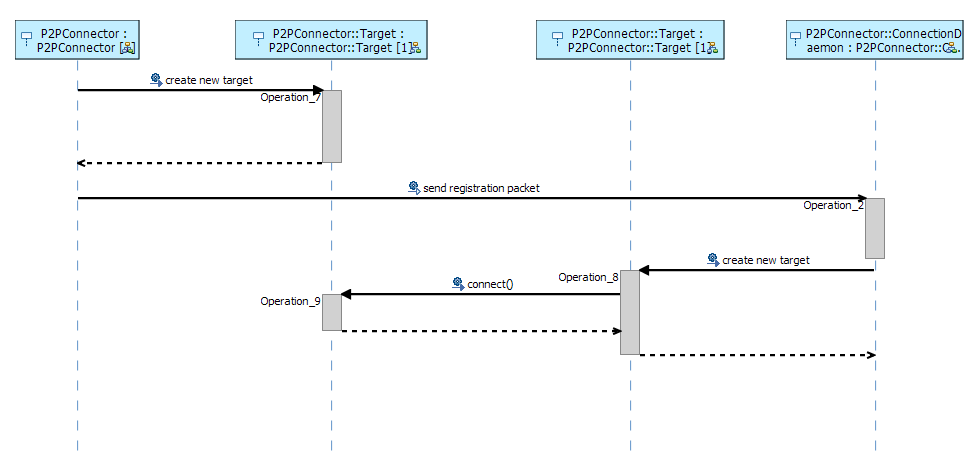


Figure 4.Peer to Peer Connector: Add New Connection Sequence Diagram

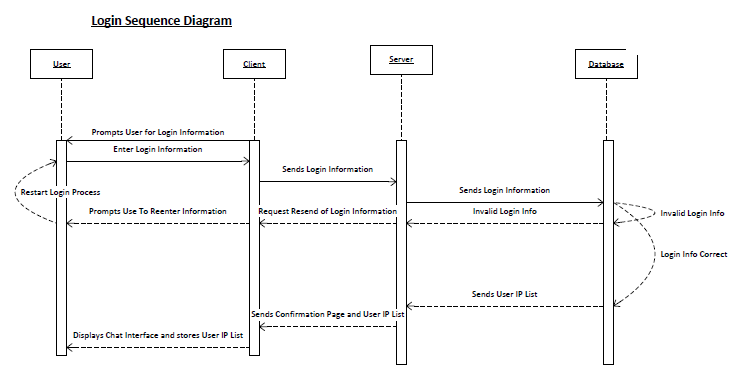


Figure 5.Client Login: Sequence Diagram

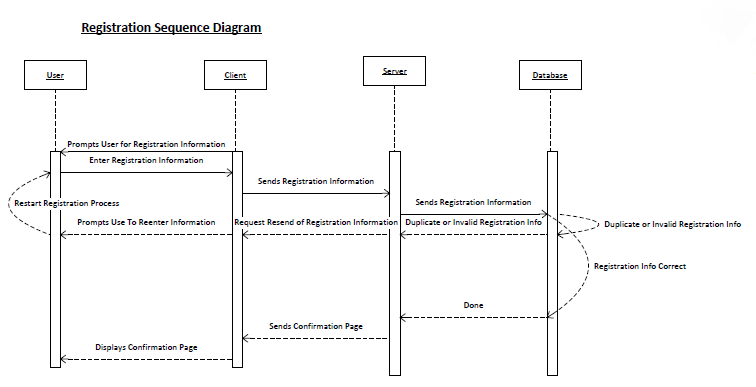


Figure 6. Registration: Sequence Diagram

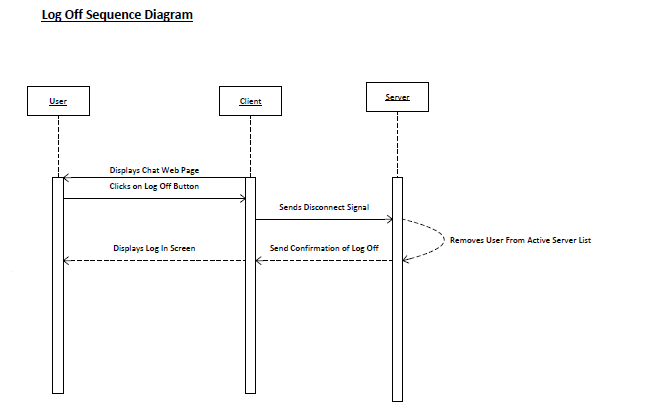


Figure 7.Log Off Sequence Diagram

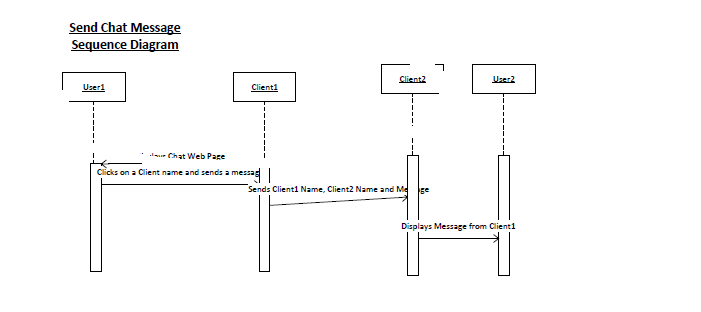


Figure 8.Send Chat Message Sequence Diagram

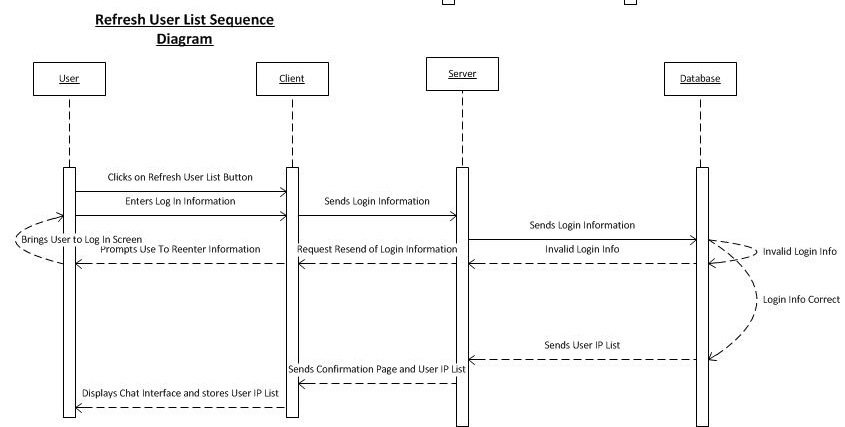


Figure 9.Refresh User List Sequence Diagram.

**State Transition Diagrams**

This section will show the behavior of the systems composed of a finite number of states.

The State Transition Diagrams include:

1. Client State Transition Diagram shown in Figure 10 – Details the various states chat client enters from login to logout.
2. Client Login Validation shown in Figure 11- Details the state transition of the system when client enters chat application.

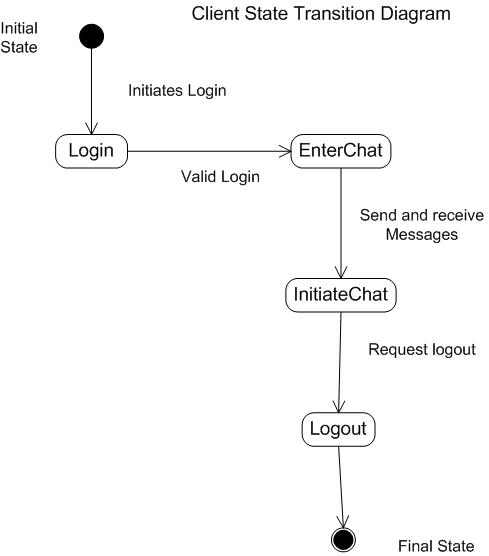


Figure 10.Client State transition with Valid Login credentials.

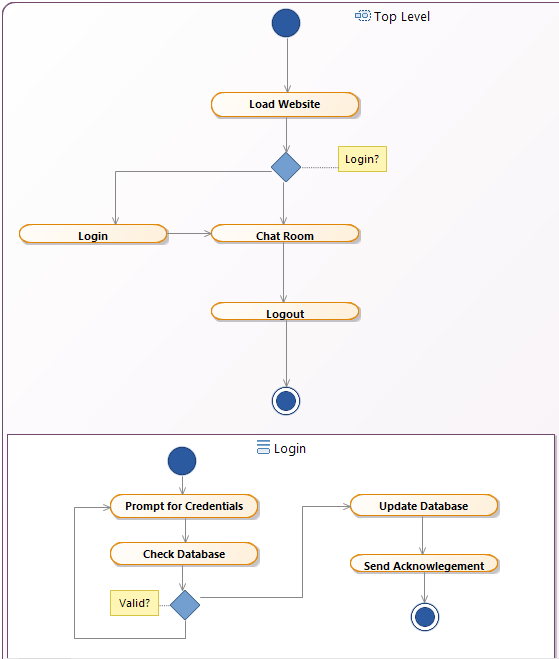


Figure 11.Client Login Validation State Transition

**Entity Relationship Diagram:**

Figure 12 details the database information of the chat application.

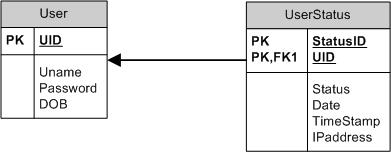
****

Figure 12: Entity Relationship diagram

**Tools Used**:

1. www.Creately.com
2. Microsoft Visio
3. Visual Paradigm UML