No Clue Crew

Chat User Manual

Version: 1.0

Authors:

Zachary Agness, Benedict Casasola, Jared Cruz, Jaeven Laron, Liyuan Zhao, Robert La

University of California, Irvine

Table of Contents

Glossary **2**

Instant Messaging **2**

1.1 Usage Scenario 2

1.1.1 Startup and Logging In 2

1.1.2 Creating an Account 3

1.1.3 Contact List and User Requests 4

1.1.4 Server Menu 5

1.2 Goals 6

1.3 Features 6

1.3.1 Client Application Features 6

1.3.2 Server Application Features 6

Installation **6**

2.1 System Requirements 6

2.2 Setup and Configuration 7

2.2.1 Configuring X11 Forwarding 7

2.2.2 Running the Application 7

2.3 Uninstalling 8

Chat Program Functions & Features **9**

3.1 Client and Server Communication 9

3.2 Log-in/Registering 10

Back Matter **11**

4.1 Copyright 11

4.2 Error Messages 11

4.2.1 Error Messages: Username is Taken 11

4.2.2 Error Messages: Username and/or Password is incorrect or does not exist 11

4.2.3 Error Messages: Username Requested does not exist 11

4.3 Index 12

4.4 References 13

Glossary

Client: a device or computer that requests data or information from a host.
GUI: Graphical User Interface
Hash Table: data structure used to cycle through account info
Host: a device or computer that provides data or information requested by a client.
IP Address: Internet Protocol Address
Peer to Peer (P2P): a term used to refer to the communication/interaction of one user with another.
Server: a device or computer that acts as a hub for information or resources in a network.
TCP Address: Transmission Control Protocol Address
Txt file: A file format that contains text information.

Instant Messaging

1.1 Usage Scenario

1.1.1 Startup and Logging In

Fig 1: Login window when user opens the application.

The user opens the application and would be greeted by a login in screen. The login in screen would prompt the user to enter their login and password. If the user is new to the application, they have the option to create an account.

1.1.2 Creating an Account

Fig 2: Creating a New Account.

When creating an account, the user will be prompted to enter the user name of their choice (as long as it is not taken) and their password, and nickname.

1.1.3 Contact List and User Requests

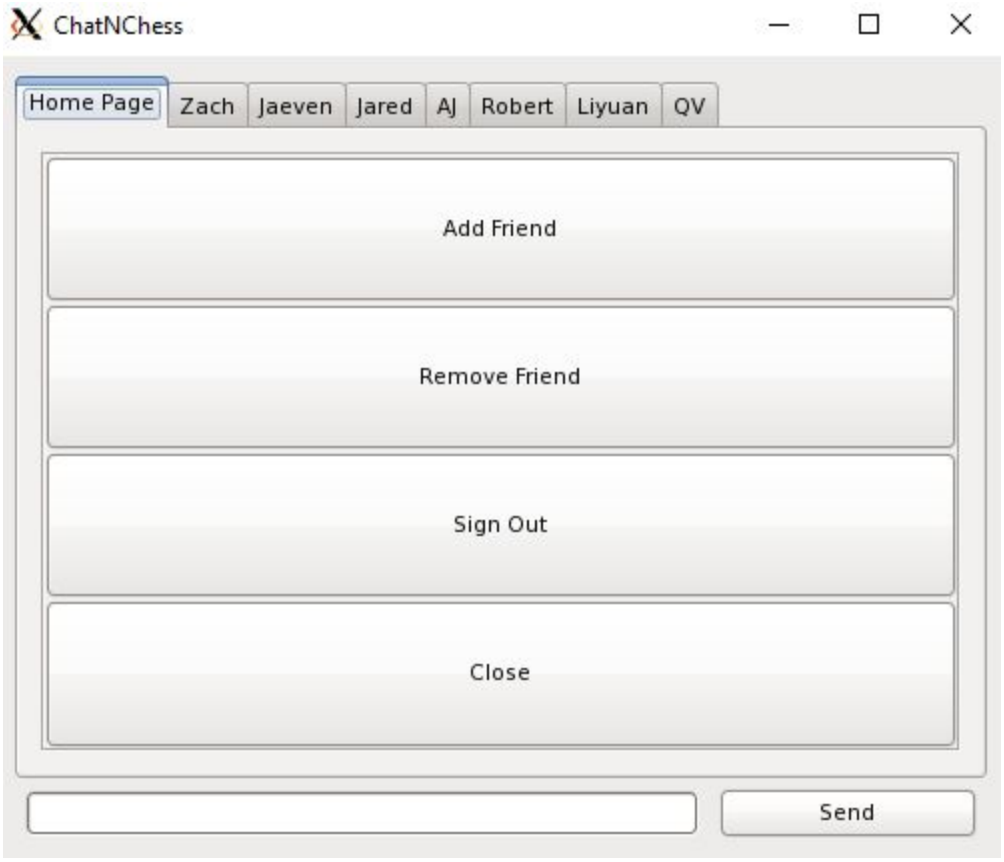

Fig 3: Contacts Menu.

Once the user has logged in, the user can send a request to others to join a chat room and chess game.
Note: Only friends can send and accept requests to a chess game. If a user wishes to play with someone, they must first make that user their friend.

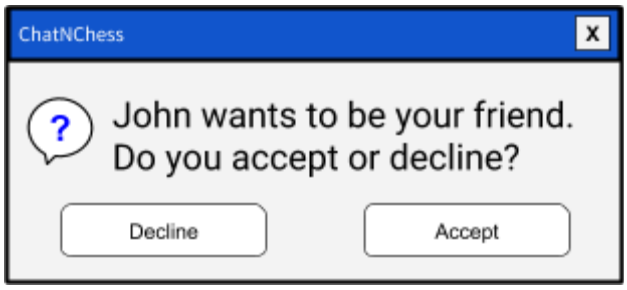

Fig 4: When a user receives a friend request, they can accept or decline.

A user can receive friend requests and can either choose to accept or decline. After they accept, the user that sent the friend request will be added to their friends list.

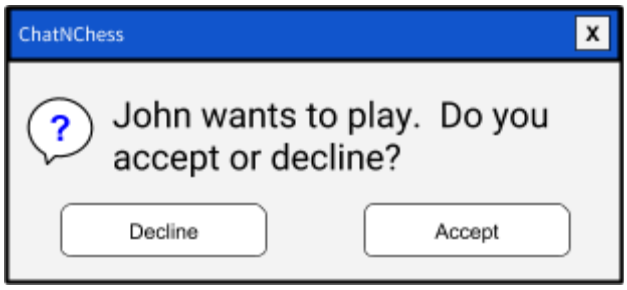

Fig 5: When a user receives a game invite, they can accept or decline.

A user can receive invites to a chess game and will be prompted by a message as shown in Fig 5. When a user accepts, a new window will pop-up with the chess game and chat log.

1.1.4 Server Menu

The server can can be accessed via ASCII interface. The server will show that it is listening and print out any response that it sends to each client. While on the server, the user can type in userlist to see the list of users.

1.2 Goals

The main goal of this project is to create an application capable of letting two users communicate via a network. In addition, the application will have a GUI interface and be able to save and call information relating to user login credentials and contact lists. The user must provide one of the EECS beach servers (i.e. zuma.eecs.uci.edu) and a port number above 2000 (i.e. 2400). This will connect the user to the server, which will receive messages and direct the messages to the desired user.

1.3 Features

The chat program has a variety of features for both the client and server applications. The client application allows users to create user profiles, and manage a friends list.

The server application manages the communication between users and stores the information and credentials of each user. The stored information is used for login and user friends/contact lists. The stored information will be stored in a hash table for easy access and increased efficiency.

For a detailed description of each feature, refer to section 3 of this manual.

1.3.1 Client Application Features

- **Register Account/Login**
- **Adding/Removing Friends**
- **Chat with other users**

1.3.2 Server Application Features

- **Sees and Manages all requests from clients**
- **Handles messaging between clients**
- **Stores user information for login and contact list purposes**
- **Hash Table for User Database**

Installation

2.1 System Requirements

OS: Windows, Mac, or Linux operating system
Environment: Red Hat Linux Environment
Accounts: UCI EECS account

Hard Disk Space: 250 MB
RAM: 1 MB
CPU: Intel Pentium
Graphics Card: Intel Integrated Graphics

2.2 Setup and Configuration

2.2.1 Configuring X11 Forwarding

For Windows OS:

1. Download XMING http://www.straightrunning.com/XmingNotes/
2. Follow the instructions on the installation wizard
3. Enable X11 Forwarding in your Linux environment

For Mac OS:

1. Download XQUARTZ https://www.xquartz.org/
2. Follow the instructions on the installation wizard
3. Enable X11 Forwarding in your Linux Environment

2.2.2 Running the Application

For Windows OS:

1. Run a terminal emulator (PuTTY, MobaXterm, etc.)
2. Enter the EECS Linux server address (zuma.eecs.uci.edu, bondi.eecs.uci.edu, etc.)
3. Login using your UCI login information
4. In the desired directory, enter these commands:
    a. % gtar xvzf Chat_V1.0.tar.gz
    b. % evince Chat_V1.0/doc/ChessMultiplayer_UserManual.pdf
    c. % cd Chat_V1.0
    d. % ./client <linuxserver>.eecs.uci.edu <port number> (Example: ./client laguna.eecs.uci.edu 9002)
5. To access the server, enter these commands:
    a. % gtar xvzf Chat_V1.0.tar.gz
    b. % evince Chat_V1.0/doc/Chat_UserManual.pdf
    c. % cd Chat_V1.0
    d. % ./server <port number> (Example: ./server 9002)

For Mac OS:

1. Run the program Xquarts (X11 in older versions)
2. Type the command:
    a. % ssh -X -Y Username@ServerName (UCI_UserName@zuma.eecs.uci.edu)

    b. Enter the password for your UCI account
3. In the desired directory, enter these commands:
    a. % gtar xvzf Chat_V1.0.tar.gz
    b. % evince Chat_V1.0/doc/Chat_UserManual.pdf
    c. % cd Chat_V1.0
    d. % ./client <linuxserver>.eecs.uci.edu <port number> (Example: ./client laguna.eecs.uci.edu 9002)
4. To access the server, enter these commands:
    a. % gtar xvzf Chat_V1.0.tar.gz
    b. % evince Chat_V1.0/doc/Chat_UserManual.pdf
    c. % cd Chat_V1.0
    d. % ./server <port number> (Example: ./server 9002)

2.3 Uninstalling

1. Run the terminal emulator program
2. Enter the EECS server and the account in which the program was installed
3. Enter the directory where the program exists
4. Type the following command: % rm client server

Chat Program Functions & Features

3.1 Client and Server Communication

Users will be able to log on and specify which server from a list they would like to join. The client application will consist of the login menus, chat rooms, and chess game.

After successfully connecting to the server, the user's friends list will be displayed. There will also be an option to add or remove a friend. If the add option is selected, the client will be prompted to enter a username to send a friend request to. The user will only be added onto the friend request if the request is accepted. An error message will be displayed if the username doesn't exist or if the user enters their own username. If the remove option is selected, the user must enter a username in their friends list or cancel the option. The same error messages apply.

Users can add other users and can request to chat with them. Each user is connected to the main server which holds user profiles and information. When a user wants to chat with a user, their request is sent to the central server and the request is sent to the other user. When the request is approved, both users are connected and can chat with each other.

The provider, or server, will store all the users of the application and the associated contacts of each user. If a new account is added, then the server will add the user information into a data structure. The server handles all info of registered users, including their username, their password, and their IP address. The client communicates with the server to authenticate and to send requests to other users.

Currently, the program handles user information by having a txt file with all the information of users. Whenever new users register, the server writes to the txt file, adding the new entry. When it needs to get information, it reads this file and checks if the user is in the database.

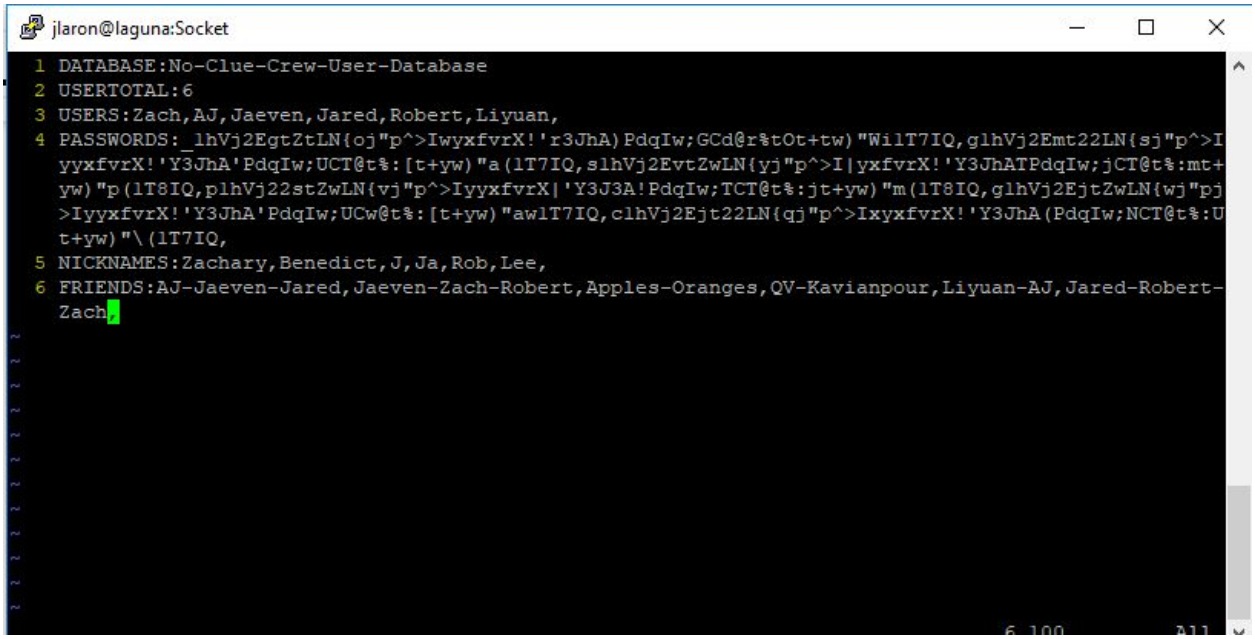

Fig 7: The server creates a txt file to store user info.

3.2 Log-in/Registering

When the application is first opened, the user will be prompted to enter an account username and password. There will be an option to login or register the account. The minimum string length for the username and password is 8 and maximum length is 30. An error message will notify the user if the inputs are out of range. The username is not case sensitive but the password is case sensitive. If attempting to login, an appropriate error message will be displayed if the username doesn't exist or if the password associated with the username is incorrect.

If registering an account, an error message will occur if the username already exists. Once an account is registered, a success message will be displayed, but the user will have to login in order to connect to the server. If the username and associated password are valid, then the client will be successfully connected to the server.

Back Matter

4.1 Copyright

Specifications are subject to change, with regards to the discovery of more optimal methods. This includes advancements in the technology of both hardware and software. The reproduction of this manual or any software, produced by its creators, is strictly prohibited. The entirety or partial reproduction is prohibited. This message relieves the creators, No Clue Crew, of all liability.

4.2 Error Messages

This program handles a series of different error checks to ensure that the user makes valid inputs throughout the entirety of the program, from menu selection to in-game player choices. The error checks and subsequent error messages that are handled by the program are listed below.

4.2.1 Error Messages: Username is Taken

("Username is already taken! Please enter another one.")
When a user who is registering a new account enters a username that is already taken, they will receive an error message

4.2.2 Error Messages: Username and/or Password is incorrect or does not exist

("Account with entered username and password does not exist. Please be sure the entered credentials are correct. Password is case-sensitive.")
When a user enters a username and/or password that does not match those stored, they will receive this error message and a prompt to re-enter a username and password that are valid.

4.2.3 Error Messages: Username Requested does not exist

("The account you requested does not exist. Please enter a valid account username.")
When a user requests to friend an account by a username that does not exist, they will be prompted to re-enter another valid username.

4.3 Index

Account pp. 1, 2, 4, 5, 8, 9, 10, 11, 12, 13
Algorithm pp. 1, 11
Application pp. 1, 4, 8, 9, 11, 12
Chat pp. 1, 6, 7, 8, 11, 12
Chess pp. 2, 6, 7, 8, 9, 10, 11, 12
Client pp. 1, 5, 5, 11
Contact pp. 1, 6, 8, 11
Hash Table pp. 2
GUI pp. 2, 8
Host pp. 2
IP Address pp. 2, 11
Login pp. 1, 4, 8, 9, 11
Peer to Peer (P2P) pp. 2
Server pp. 1, 2, 8, 9, 10, 11
TCP Address pp. 2
Txt file pp. 2
Username pp. 1, 10, 11, 13, 14

4.4 References

Chess_UserManual, 1st ed. No Clue Crew, Irvine, CA, 2019.

Harrison, Colin. “Xming X Server for Windows.” Xming X Server for Windows - Official Website, 2015, www.straightrunning.com/XmingNotes/.

McKay, Kyle J. “XQuartz.” XQuartz, www.xquartz.org/.