CS2110 Project Report

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Story ID | Story | Acceptance Criteria | Sprint | Status |
| 1 | As a user, I would like to be able to login to the google talk IM network | 1) Entering the correct username and password allows user to login to system | 1 | completed |
| 2 | As a user, I would like to be able to see my buddy list | 1) Upon logging into system, correct buddy list for user shows up | 1 | completed |
| 3 | As a user, I would like to be able to add and remove buddies from my buddy list | 1) System allows user to add buddy to list  2) System allows user to delete buddy from list if he/she is already on the list | 2 | Partially completed |
| 4 | As a user, I would like to be able to send and receive chat messages | 1) Chat will begin when user selects buddy  2) Chat will close upon user closing chat window  3) Chat will begin when buddy sends message to user | 1 | Partially completed |
| 5 | As a user, I would like to be able to post my current status | 1) User should be able to add/remove status  2) Buddies of user should be able to see user's updated status | 2 | Partially completed |
| 6 | As a user, I would like to be able to see which buddies are online and offline | 1) Buddy list distinguishes between online and offline buddies | 2 | completed |
| 7 | As a user, I would like for the IM client to have additional features | 1) Chat history | 2 | completed |
| 8 | As a user, I would like to be able to logout after my session | 1) Buddylist and existing chats close when user selects logout | 1 | completed |

To log in, the user is required to enter a valid GoogleTalk username—including the domain of the username (ex: “@gmail.com”)—and a valid password. Checking the box to “Remember User” will keep the username and password stored for the next log-in. A buddylist appears upon connection. Lists of online and offline buddies are displayed. To add a new buddy, the user clicks the “+” button on the bottom of the screen. The buddy you are adding must accept your request before you are able to chat with them. Double clicking on an online buddy will open up a new chat with them. All previous GChat Client messages with this user are displayed. The “Status” Button allows the user to display a status.

Program Design: descriptions of login, buddyList, addBuddy, chatWindow, changeStatus

**Class: Login**

***Description:*** Upon running the program, a new login screen is displayed that allows for user-input, then sets up a connection provided a valid password/username. The username and password are stored for the next login if the remember box is checked off.

***Important Fields*:** JFrame frame, JtextField login, JPasswordField pwdPass, JCheckBox remember

***Important Methods/Classes:***

main(String [] args): calls run to set up a new login window.

initialize(): creates a new GChat JFrame which allows the user to input information

class SwingAction: upon clicking “login”, sets up a new connection using username and password. Also saves username and password to a file if remember box is selected

**Class: buddyList**

***Description:*** Upon login, sets up a Buddy List that separates online/offline users, and updates information during session. Also defines what occurs when clicking “login”, “+”, and “-“.

***Important Fields:*** XMPPConnection connection, Vector<String> onlineListData, Vector<String> offlineListData, Roster roster

***Important Methods/Classes:***

buddyList (XMPPConnection, String): goes through user’s roster, stores each buddy in offlineListData or onlineListData based on if they are online/offline, calls initialize() at the end, which creates a new buddyList window.

updateBuddyList(): updates the buddylist for each time after login that a buddy changes between online/offline.

class LogoutAction: upon clicking “logout”, a window appears which exits program and ends connection upon confirmation.

class AddFriendAction: upon clicking “+”, the methods defined in the addBuddy class begin

class DeleteFriendAction: upon clicking “-“, removes the currently selected buddy from the buddylist

**Class: addBuddy**

***Description:*** A new window appears upon clicking “+” which allows the user to input an email and a name for the buddy. This new buddy is added to the user’s roster once the new buddy confirms.

***Important Fields:*** JFrame frame, JTextField textfield, Roster roster, JTextField name

***Important Methods/Classes:***

addBuddy(Roster, XMPPConnection): calls initialize() which sets up a new window which allows for user input and updates the user’s roster and connection

class BuddyAdd: upon clicking “Add Buddy”, creates a new buddy to add to the user’s roster based on the inputted email and name.

**Class: chatWindow**

***Description:*** Allows the user to select a buddy and begin a chat. All sent and received messages are displayed in addition to message history.

***Important Fields:*** RosterEntry chatBuddy, ChatManager chatManager, JTextField message, Chat chat, File file

***Important Methods/Classes:***

chatWindow(RosterEntry, chatManager): creates a new chat, accesses previous messages between users, calls initialize()

initialize() : creates a new window upon clicking a buddy and calls sendMessage(String) upon clicking “Send” in the chat window

sendMessage(String): sends a message from the user to the buddy, displays it, and stores the information on the file of message history

**Class: changeStatus**

***Description:***  Allows user to create a status upon clicking “Status”

***Important Fields:*** JFrame frame, Roster roster, XMPP Connection connection

***Important Methods/Classes:***

changeStatus(Roster, XMPPConnection): calls initialize which sets up a window that allows user to input a status, then updates the roster and connection based on the user’s new presence, which is defined in the BuddyAdd class

class BuddyAdd: defines what occurs upon clicking to update status, updating presense