1. Class Diagram UML

Password
- + password: String
- + length: int
- + validCharacters: String[]
- + changePassword(String):
- + isValid(String):boolean

User
- + userType: UserType
- + id: int
- + firstname: String
- + lastname: String
- + username: String
- + password: Password
- + blockedFlag: boolean
- + state: UserState
- + chatLogs : ListArray<ChatLog>
- + User(String, Password)
- + changeUsername(String)
- + changePassword(String)

<<enumeration>> UserState
- Online
- Offline
- Spam
- Away

<<enumeration>> UserType
- Regular
- Admin
- Default

Server
- + users: ArrayList<User>
- + chatLogs: ArrayList<ChatLog>
- + messageQueue(): Queue
- + MSGMAXLENGTH: int
- + startUp()
- + loadMessageThreadArray()
- + loadUsers()
- + processMessageQueue()
- + updateMessageThread()
- + sendUpdatedMessageThread()
- + checkLogin(Message): Message
- + blockUser(User)
- + unblockUser(User)
- + closeServer()
- + saveCharLog()
- + saveUsers()
- + userLogout()

MessageThread
- + id: int
- + messageList: ArrayList<Message>
- + participants: ArrayList<User>
- + created: Date
- + lastUpdated: Date
- + lastServerUpdateSent: Date
- + displayName: String
- + MAXMESSAGE: int
- + ChatLog(type): type
- + addMessage(User, Message):

Message
- + counter: int
- + id: int
- + type: MessageType
- + content: String
- + owner: User
- + created: Date
- + editedFlag: boolean
- + dateEdited: Date
- + messageThreadID: int
- + Message(MessageType):
- + Message(User, String, MessageType)
- + Message(User, String, ChatLog)

<<enumeration>> MessageType
- Login
- Logout
- Initial Connection
- Invalid Login
- Valid Login
- New Text
- Message Thread
- Block User
- Unblock User
- Check User

GUI
- + user: User
- + createMessage():
- + createMessageLog():
- + login()
- + logout()
- + processCommands ()
- + sendMessage():
- + view/OnlineUsers():
- + viewTLogs():
- + view messages():

Client
- + currentUser: User
- + notifications: ListArray<Notification>
- + SessionActive: Boolean
- + messageQueue(): Queue
- + client(): void
- + login(username String; password String): void;
- + logout():
- + pingServer():
- + updateMessageThreads():
- + sendMessage(Message message; ChatLog): void
- + receiveMessage(): Message
- + SetUserStatus(UserState): User
- + GetUserStatus(username String): User
- + print(Message message): void
- + GetChatLog(username String): ChatLog

Client Interface
- + processCommands()

{ordered, unique}
*{ordered, unique}
*{ordered, unique}
*{ordered, unique}
0..*