

Table of Content

1 Introduction
1.1 Overview
1.2 Goals and Objectives
1.3 Scope
1.4 Definitions
1.5 Document Conventions
1.6 Assumptions
2 General Design Constraints
2.1 Product Environment
2.2 User Characteristics
2.3 Mandated Constraints
2.4 Potential System Evolution
3 Nonfunctional Requirements
3.1 Operational Requirements
3.2 Performance Requirements
3.3 Security Requirements
3.4 Safety Requirements
3.5 Legal Requirements
3.6 External Interface
3.6.1 User Interface
3.6.2 Software Interface
4 System Features
Server
4.1 Feature: Registration and Login

4.1.1.1 Description: Registration
1 Introduction
4.1.1.2 Use-case: Registration
4.1.2.1 Description: Login
4.1.2.2 Use-case: Login
4.2 Feature: Messaging
4.2.1.1 Description: Private Message
4.2.1.1 Use Case: Private Message
4.2.2.1 Description: Multicast Message
4.2.2.2 Use-case: Multicast Message
4.2.3.1 Description: Broadcast Message
4.2.3.2 Use-case: Broadcast Message
Client
4.3 Feature: Registration and Login
4.3.1.1 Description: Registration
4.3.1.2 Use-case: Registration
4.3.2.1 Description: Login
4.3.2.2 Use-case: Login

1 Introduction

1.1 Overview

This document defines the requirements of the server and client that we develop in the laboratory. For this reason every section will be divided into two parts, one for the client and one for the server.

This document will be useful for the user and for the developer that will need to work on the code.

1.2 Objectives

Server

- Develop a server able to make chat client.
- The server will work with different clients, but using the same protocol.

- The server have some feature to help the final user in the use of the software.

Client

- Develop a client able to speak with other clients across a server
- Gives the user an easy interface to use the client
- Develop a security access to the user's privacy

1.3 Scope

Server

The scope of the server is make communicate an N number of clients between them, and offert at the users some function that can halp, like see the online users and write a broadcast message.

Client

The scope of the client is to communicate with an N number of other clients through a common server, and gives the user the possibility to utilize the function offered by the server (like send normal message, send broadcast message, see the online user or group message)

1.4 Definitions

definizioni di cose che possono non essere chiare nel documento

1.5 Document Conventions

GUI - is the acronym of Graphic User Interface

1.6 Assumptions

It is assumed that the client and server are in the same local network, and the firewall is disabled or is integrated a rule that allow the flow of the packet.

2 General Design Constraints

2.1 Product Environment

Server

The server's software will be put on a server machine, that have to be always active, to permit all the client to connect it and use his services.

Client

The client software can be utilized on any type of machine, and you can start and stop it whenever you need to.

2.2 User Characteristics

Server

In the server there are no different user, there is only one basic level that allow you to chat with every client online.

Client

In the client only one user at the time can access, and to access it must use his username and password.

2.3 Mandated Constraints

The only constraints that we encounter is the obligation to use the protocol that the professor De Carli gave us, this to make all the communication protocol the same.

2.4 Potential System Evolution

Server

At this moment the software of the server is done, but is very basic, so it can be updated in the following ways:

* Add more security control * Product a log system to keep track of the userm movements. * Encrypt the message sent in network.

Client

At this moment the software of the client is done, but is very basic, so it can be updated in the following ways:

* Make the graphic interfaces more easier for the user to understand * Encrease the efficency of the client code

3 Nonfunctional Requirements

There are no nonfunctional requirements at this time.

3.1 Operational Requirements

There are no operational requirements at this time.

3.2Performance Requirements

Server

No requirements for speed have been set forth.

Memory requirements will be set by the size of the .csv file, that contains all the username and password of the users, but is a simple text file, so it is not relevant.

3.3 Security Requirements

There are no security requirements at this time.

3.4 Safety Requirements

There are no safety requirements at this time.

3.5 Legal Requirements

There are no legal requirements at this time.

3.6 External Interface

3.6.1 User Interface

Server

The server have no a GUI, so the only thing you will se on the screen is which computer identify by the IP address is connect at the server.

Client

The client user interface now is very easy to use. It's composed by 3 different text spaces (one for each the type of message), a column where you can see all the online user, and the button to do the logout.

3.6.2 Software Interface

Server

See User Interface

Client

The client have no a server interface

4 System Features

Server

4.1 Feature: Registration and Login

This section describes how new user information will be added to the list of users and how a user can login.

4.1.1.1 Description: Registration

The registration function allow who don't have a account to create a new account and add the information to the User.csv file.

4.1.1.2 Use-case: Registrant

Actors: anyone does not have an account

Description: a user without account connect to the server from a client, and begin the registration procedure.

Phat: 1. The client sent to the server a packet format by a 10(that is the number of the registration packet)(1 byte), the length of the data field(2 byte), and after the username and password of the new account separated by a 0. 2. The server processes this packet and, if all go fine, add the new account at the user.csv file. 3. If all go fine, the server respond at the client with an OK packet, but if somethigs go wrong, it respond at the client with an ERROR packet

4.1.2.1 Description: Login

The login function allow a user that have an account to use all the function of the server.

4.1.2.2 Use-case: Login

Actors: all the user with an account

Description: a user that need to login to chat from a client to another client.

Phat: 1. The client sent to the server a packet format by a 11(that is the number of the login packet)(1 byte), the length of the data field(2 byte), and after the username and password of the account separated by a 0. 2. The server processes this packet and compares the username and password to the User.csv file. 3. If find a corrispondence, send to the client a OK packet, but if don't find it, it will send an ERROR packet, specifying the type of error. 4. If the login go fine, the server set the user to online, and unlock the messaging function

4.2 Feature: Messaging

This section show all the type of message the server can manage, like: 1. Private message; 2. Multicast message; 3. Broadcast message;

4.2.1.1 Description: Private Message

The private message is a message that one user can send to another online user, so a point to point message.

4.2.1.2 Use-case: Private Message

Actors: two login users

Description: the A user want to send a message to the B user using the chat server

Phat: 1. The client of the A user sent to the server a packet format by a 22(that is the number of the sender private message packet)(1 byte), the length of the data field(2 byte), and after the recipient and text of the message separated by a 0. 2. The server processes the packet and, if the recipient is online, send a packet to the B user client. 3. The packet is format by a 23(that is the number of the recipient private message packet)(1 byte), the length of the data field(2 byte) and the sender and text of the message separated by a 0. 4. If all go fine, the server respond at the A user client with an OK packet, but if somethings go wrong, it respond at the client with an ERROR packet

4.2.2.1 Description: Multicast Message

The multicast message is sent from one user to N users, so the user need to specify at who he want to send the message.

4.2.2.2 Use-case: Multicast Message

Actors: a user and N online users.

Description: a A user want to send a message to N number of online users.

Phat: 1. The client of the A user sent to the server a packet format by a 24(that is the number of the sender multicast message packet)(1 byte), the length of the data field(2 byte), the N recipient of the message separated by a 0 and at the end the text of the message. 2. The server processes the packet and, if all the recipients are online, send a packet to all the N user client. 3. The packet is format by a 25(that is the number of the recipient multicast message packet)(1 byte), the length of the data field(2 byte) and the sender and text of the message separated by a 0. 4. If all go fine, the server respond at the A user client with an OK packet, but if somethings go wrong, it respond at the client with an ERROR packet

4.2.3.1 Description: Broadcast Message

The broadcast message is sent from one user to all the online user, so the user don't need to specify the recipient.

4.2.3.2 Use-case: Broadcast Message

Actors: a A user and all the online users.

Description: a A user want to send a message to all the online users.

1. The client of the A user sent to the server a packet format by a 20(that is the number of the sender broadcast message packet)(1 byte), the length of the data field(2 byte) and the text of the message.
2. The server processes the packet and send a packet to all the online users.
3. The packet is format by a 21(that is the number of the recipient broadcast message packet)(1 byte), the length of the data field(2 byte) and the sender and text of the message separated by a 0.
4. If all go fine, the server respond at the A user client with an OK packet, but if somethings go wrong, it respond at the client with an ERROR packet

4.3 Feature: Registration and Login

This section describes how to add a new user from the client and made a login.

4.3.1.1 Description: Registration

The registration function allow who don't have a account to create a new account and send the information to the server

4.3.1.2 Use-case: Registrant

Actors: anyone does not have an account

Description: a user without account connect to a client, and begin the registration procedure through the user interface.

Phat: 1. The user on the GUI taps on the registration button 2. The client open a new page where the user put a username and a password that will use to access tp his account. 3. This data will be sent by the client to the server

4.3.2.1 Description: Login

The login function allow a user that has an account to use all the function of the server.

4.3.2.2 Use-case: Login

Actors: all the user with an account

Description: a user that need to login to chat from a client to another client.

Phat: 1. The user on the GUI taps on the login button 2. The client open a new page where the user can put his username and password and after the user taps on the send button 3. The client sent this data to the server and if this data corresponds with a real account the server send to the client a OK message and the client open to the user the GUI with the possibility of send message, else the server send to the client an ERROR message, and the client communicate it to the user.