

Milestone 1-Low Level Design

Brief information on the project:

Creating a virtual environment in which users can post their messages and read the messages written by other users connected to the same server.

Classes:

User Class

General functionality and brief info:

The class "User" represents an actual person registered to the system or using the chat.
The "user" class has the purpose of setting each person who uses the chat with specific identification properties.

Fields:

Nickname – unique name for the user as an identification mean.

g_ID – this is the group ID of each user.

Status – the current status of the user (Online or Offline).

Functionality:

sendMessage – Allows the user to send messages to the chat room.
The method receives a String (content of the message) and returns a full message object containing all the relevant details including the content itself.

logout – logging out of the system.

changeNickname – Receives a String with a new nickname for a user and changes the user name to the new one.

isOnline - This method is the user's status in the system at any time.

true – Online.

false – Offline.

Message Class

General functionality and brief info:

The Message class is a way to represent each message that goes through the server and to the chat room. Each message has unique details such as time, date, nickname, guide retrieved from server etc.

Fields:

timeStamp – time and date in form of String.

guidNum – The ID number of a message given by the server.

messageBody – A string with the message's content.

isSent – Boolean referring to the message status (been sent or not).

isReceived - Boolean referring to the message status (been received or not).

Writer_name – String holding the name of the user who sent the message.

Writer_gID – String holding the group id of the user who sent the message.

Functionality:

checkContentValidity – Used mostly by the constructor. Check each message for the specified conditions (150 characters at most, proper language etc.).

Milestone 1-Low Level Design

convertMessage – this method receives an Imessage object and converts it to a message.

ChatRoom Class

General functionality and brief info:

ChatRoom class is the “Manager” of a single chat room. It manages the messages and users in form of lists and has the status of all current actors at each moment.

Fields:

loggedInUser – Represents the user who is currently connected to the chat room.

Messages – List of all the sent messages.

Users – List of all the registered users.

url – The address of the server.

Functionality:

registration – This function is handling the process of creating new users. it receives nickname and group ID and if those details are correct (a verification is needed) , new user is being created.

login - Receives a nickname and group ID as identification properties and if the detailed are correct, the user can enter the chat room.
the user's status changes to online and loggedInUser (in the chat room) changes as well.

send – Through this method, an online user can send message into the chat room.
It receives the message content and reaches to the user's class to send the message from the user itself (OOP).

dispMsgOfUsr – This method displays all the messages sent by a specific user.
It receives nickname and group id in order to identify the specific user.

dispRetrieveMessages – This method receives a number and retrieves from the server the number of last messages from the current chat.
after those messages are retrieved, the method displays it.

retrieveMessages – This method receives a number and retrieves the last number of messages from the server (using list).

changeNickname - This method receives a string with the new nickname and update the list of messages and users with the new nickname (after checking existance).

GUI_CLI Class

General functionality and brief info:

GUI_CLI – Basic console display for communication with a user.

functionality:

startUpMenu - Presents the chat room GUI.

closeChatApp – Closing the application.

Milestone 1-Low Level Design

FileHandler Class

General functionality and brief info:

FileHandler class is used to handle files, the class instance is a singleton object that classes related to the Business Logic layer can use to execute commands related to reading and writing files.

Fields:

instance – An instance of the singleton filehandler.

currDir – The directory manager.

userListPath – An address to the users list file location.

msgListPath – An address to the messages list file location.

streamUsers – A stream used to open a “gate” to write and read from users file.

streamMessages – A stream used to open a “gate” to write and read from messages file.

bf – An object used to convert files to a code and the other way around.

Functionality:

saveMessageList – saving the messages list to the data base.

saveUsersList – saving the users list to the data base.

editUserNickname – used to edit an existing user object nickname.

loadMsgList – used to load the last updated list of messages, carry an input of a given number to load a specific amount of messages.

loadUserList – used to load the last saved users list, carry an input of a given number to load a specific amount of users.