# HLD

***High level Design***

The client is built on four main layers :

**Presentation layer**

The presentation layer contain all the necessary functions that allow the program to interact with the user directly throw the GUI all interaction with the user is done by this layer from display and reserving information from the user , our gui is made by wpf form.

**Business layer**

This layer is also called the “logic layer“ and its manage all the logical decisions based on the information given by the other layers ,managing the interaction between the layers and proccing the information giving by it ,to give each layer independence from the other layers allowing adding new layers for more flexibility or upgrading the existing once .

**Persistent layer**

This layer is responsible for communicating with the SQL server , its responsible to take care of completing the 1) registration 2) log-in 3) retrieving messages 4) filtering messages 5) sending messages ( basically everything connected or requires a connection with the sql server ) also it makes sure to store the log file on the running machine in order to debug problems if needed.

## **Terminology**

**Chat Room**

A virtual environment in which users can post their messages and read the messages written by other users. The chat room’s display is updated each 2 seconds with the latest messages that were sent. In the chatroom the user can filter and update the list of messages that is being displayed on the screen . moreover the user can edit his messages if he wants to . giving him the opportunity to change his mind or correcting a mistake in a message that he sent before.

**User**

A person who interacts with the system , a user has a unique ID stored in the database in addition to his the Nickname , Password and Group id.

**Nickname**

A familiar or humorous name the user uses to identify himself , the nickname shall be no longer than 10 chars.

**Password**

A private field that is required to identify the user , the password length must be longer that 3 chars and shorter 18 and must contain only characters and numbers. Salt will be added to the password and then it will be hashed and then it will be stored in the database when registering . this increases the security of our database.

**Group ID:**

A non-negative integer that represents the number of group which the user is registered to.

**Registration**

The act of recording user details , in order to register successfully the user is asked to type his own Nickname , Password and Group ID , of course they must be legal , anyway the app is programmed in a way that tells the user if he typed something illegal . in addition , those fields mustn’t match other user’s field ( a user that was registered in the past ) .

**Login**

A login request is initiated by the user, the request is sent to the server as a request to participant in the chat room , which is sent with user ID and password .

**Logout**

A logout request is initiated by the user, the request is sent to the server as a request to disconnect the user form the chat room , which is sent with user ID and password .

**Login/Logout**

The act of signing into and out of the system by the user. In order to log-in the user is asked to fill the required fields and if the system recognize the user ( if he is registered in the database ) a chatroom will open , where the user can do some operations.

**Message**

The text which the user delivers. Message content is limited to 150 characters. The message has a unique GUID which is stored in the database and is not shown to the public (users ) , a message knows the user that sent it and

**Message Frame**

A written communication sent between the users of the system. A wrapper for a message. The message has a lot of fields but the most important ones are the : GUID , user who sent it , when it was sent , the group id of the sender and of course its body (content) .

## **Requests**

**Displaying the messages :**

Every two seconds the displayed messages are updated automatically , NOTE : the interface only display the newest 200 messages , after that it starts adding to the interface and deleting/hiding the least updated message from the displayed messages. The Messages displayed on screen include the following details: sender group and name, message time in local timezone, message body.

**Send message :**

A send message request is initiated by the user, the request is sent to the server, which assigns the message with a unique ID (GUID) and the server’s timestamp. If the message is legal due to its length the app will inform the user about the problem and it will not send the message.

**Edit message :**

A user can ONLY edit messages that HE wrote , the user can select an image from the interface and if he was the sender the app will show a box which allows the user to edit the message , the edit message’s length is also limited to 100.

**Filter :**

The user can determent the way the messages are displayed on the GUI . the user can filter the messages by : 1) User name 2) user name AND group id.

**Sorting :**

The user can sort the displayed messages by time OR groupID OR sender name . in acceding or descending order.

## **Actors :**

**Users**

A person connected to the chatroom using a client software, for sending ,editing and displaying messages. A user is identified by his group ID that is unique to his group , a password and a nickname .