
# Frankfurt University Of Applied Sciences

– Fachbereich Informatik –

## Implementation an Application for Sharing Messages, Files and Images, similar to WhatsApp, Discord or Slack

Object-Oriented-Programming Java Advance

Bachelor of Science (B.Sc.)

Submitted by

**Kaddour Alnaasan**

Matriculation number: 1310333

**Mohammed Dawoud**

Matriculation: 1319596

**Harsh Mukhiya**

number: 1400120

Referent : Prof. Dr. Armin Lehmann

Prof. Dr. Armin Lehmann

LOCATION:
Frankfurt am Main

TIME FRAME:
14. Juli 2022

# DECLARATION

We hereby certify that we have written this paper independently and have not used any sources other than those listed in the bibliography.

All parts taken verbatim or in spirit from published or as yet unpublished sources are identified as such.

Drawings or illustrations in this work have been prepared by us or are accompanied by an appropriate source reference.

This work has not been submitted in the same or similar form to any other examining authority.

*Frankfurt am Main, 14. Juli 2022*

Kaddour Alnaasan

Mohammed Dawoud

Harsh Mukhiya

*We have seen that computer programming is an art,*
*because it applies accumulated knowledge to the world,*
*because it requires skill and ingenuity, and especially*
*because it produces objects of beauty.*

— Donald E. Knuth [Knu74]

# ACKNOWLEDGMENTS

Put your acknowledgments here.

Many thanks to everybody who already sent me a postcard!

Regarding the typography and other help, many thanks go to Marco Kuhlmann, Philipp Lehman, Lothar Schlesier, Jim Young, Lorenzo Pantieri and Enrico Gregorio[1], Jörg Sommer, Joachim Köstler, Daniel Gottschlag, Denis Aydin, Paride Legovini, Steffen Prochnow, Nicolas Repp, Hinrich Harms, Roland Winkler, Jörg Weber, Henri Menke, Claus Lahiri, Clemens Niederberger, Stefano Bragaglia, Jörn Hees, Scott Lowe, Dave Howcroft, and the whole LaTeX-community for support, ideas and some great software.

*Regarding LyX*: The LyX port was intially done by *Nicholas Mariette* in March 2009 and continued by *Ivo Pletikosić* in 2011. Thank you very much for your work and for the contributions to the original style.

1 Members of GuIT (Gruppo Italiano Utilizzatori di TeX e LaTeX)

# CONTENTS

# LIST OF FIGURES

# 1

# THEORY

## 1.1 JAVA

Java is one of the most popular programming language. It is developed by Oracle Corporation on May 23, 1995. It is also used to develop mobile apps, web apps, desktop apps, games and much more. It has helped to reduce costs, shortens development timeframes, drive innovation, and improve application services. [Sch22]

There are five primary goals in the creation of the Java language.

- It must be simple, object-oriented, and familiar.
- It must be robust and secure.
- It must be architecture-neutral and portable.
- It must be interpreted, threaded and dynamic.

## 1.2 SOCKET

Socket programming in Java provides a tool that allows communication between two applications that run on different devices. Socket allows a connection between a client and a server. Typically, it provides a way to connect two nodes on a network to exchange data. One node listens on a specific port on an IP, while the other socket connects to the other to establish a stable connection.

Initially, the client must wait for the server to start. While the server is running, the client must send a request to the server and wait for its response. The socket connection uses two computers that have the necessary information about each other's location on the network location, i.e.IP address and port.

## 1.3 THREAD

A thread of execution is the smallest sequence of programmed instructions that can be managed independently by a scheduler, which is typically a part of the operating system. The implementation of threads and processes differs between operating systems,but in most cases a thread is a component of a process. The multiple threads of a given process may be executed concurrently *viamultithreadingcapabilities*, sharing resources such as memory, while different processes do not share these resources. [Thr]

A thread is a thread of execution in a program. The Java Virtual Machine allows an application to have multiple threads of execution running concurrently. Every thread has a priority. Threads with higher priority are executed in preference to threads with lower priority.

## 1.4 GIT

Git is an open source distributed version control system, mainly used for source code management, with an emphasis on speed. Git was initially designed and created by Linus Torvalds for Linux kernel development. Git operates on a decentralized architecture, so every Git working directory is a full-fledged repository with a complete history and full revision-tracking capabilities, and is not dependent upon network access or a central server. Unlike popular non-distributed predecessors, such as Subversion and CVS, Git only needs a central server for one thing: publishing changes to users of that server. You can equally share changes directly with other people without the need to consult a central hub. [lif17]

## 1.5 OUR SOFTWARE PROJECT

In our project, we have developed an application where two or more users can interact and exchange messages of different file types such as text, image, video, PDF, and so on. Mainly there are two GUI windows which make the communication user-friendly and easy to communicate. First window is the server and the other is the client. The server window has the main privilege of starting and stopping the application. When the server is started, the clients can exchange messages. The client sends a message to the server and the message is forwarded by the server to the target user. The recipient client can receive the message. If necessary, admin can delete the clients or the group by selecting.

On the Client window, user can send messages to different users and groups. First, user needs to select user or group he/she want to chat with other user or group, user will be able to send text to the other user and group in the prototype version of our application.There is small text area where user will be able to write in it and if they want to attach other document with it,can add with add File button and forward it to other with send button. User can navigate other users and group using the search. User can also select new contacts and create new group with users. On the config tab, the ip address, port number and name of server and client can be seen and can be saved if save button is clicked.

Our team divided the development of this application into different versions, with each new version our team will add more functionality, features, try to fix bugs and provide user-friendly experience to the users.In the version 1.0, users can simply chat with other user and groups. In the version 2.0, users will be able to send images,text files,pdf files etc. In the version 3.0, user can

send videos and play it on message window. Later, our team will come with more ideas and work with user's suggestions to improve our application.

## 1.6 UDP PROTOCOL

UDP is a communication protocol implemented across the internet for especially time-sensitive transmissions such as video playback or DNS lookups. UDP provides a quick transfer of data. Unlikely, as any other protocol it is also a way of transferring data between two computers in network. By UDP, data are sent in packets directly to the target IP address and port number, without establishing a connection first. One computer can simply start transferring data to others. If there is any errors, loss, and duplication, the UDP doesn't resend the data.

## 1.7 OUR PROTOCOL SEGMENT FORMAT

Our basic protocol segment consist of opcode, sender and receiver. The bases are divided into different bytes arrays. The first 4-bytes of protocol segment is reserved for opcode which whose basic functionality is to store functionality like register,deregister, search, poller, message, create group, join group, leave group and group message. The second chunk of 16 bytes presents IP address of the sender computer. The third chunk of 16 bytes reserved for the receiver IP Address.

### 1.7.1 *Register*

To register the client with the server, the client must send the basic protocol, host name, and port number to the server. The 0x01 opcode is required to register a client with the server using this protocol. The hostname requires 16 bytes of data to be stored as part of the protocol and the port number requires 4 bytes of data.

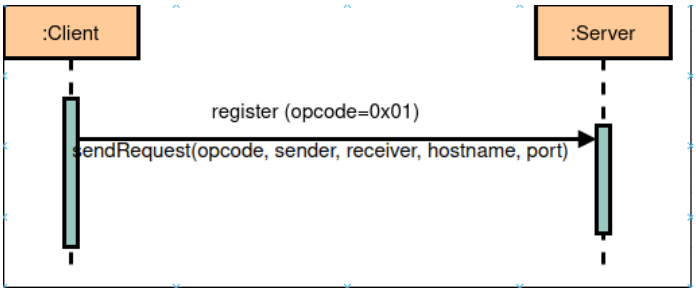


Figure 1: Register Client

### 1.7.2 *Deregister*

To deregister, the client needs to send the opcode 0x02 to server with the sender and recipient IP address. Then the server logs off the client.

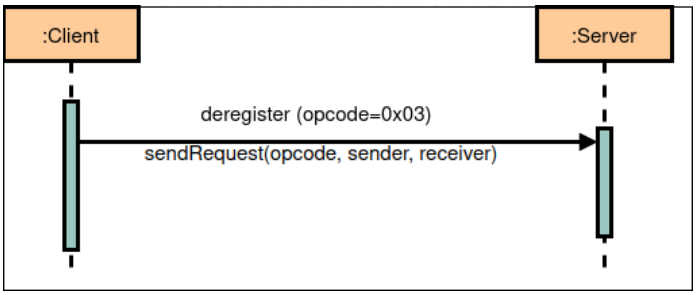


Figure 2: Deregister Client

### 1.7.3 *Search Clients*

In order to get list of clients, client actually need to send the opcode 0x03 to the server. After receiving this opcode, client get the all of all client on the server and will be able to send message to the different client.


Figure 3: Get list of clients

### 1.7.4 *Puller*

Puller is actually send a signal to server. If any new message is received by the server. If server has received new messages, then it will return messages to client.

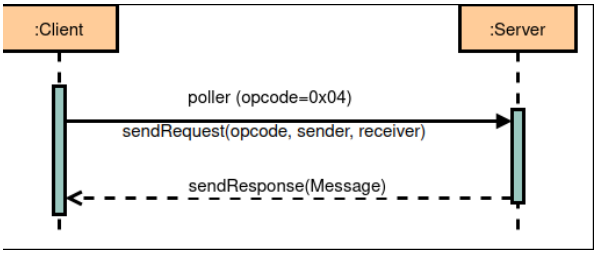


Figure 4: Pull message

### 1.7.5 *Message*

If the sender wants to send data to the receiver, it must specify the base protocol and type the content and the content itself, where the type of data is stored in 4 data bytes and the content can have any length.

The type of data has five different type.

- Message type register is presented by 0x01.
- Message type deregister is given by 0x02.
- Message type search is reserved as 0x03.
- Message type message and file is given by 0x04 and 0x05 respectively.

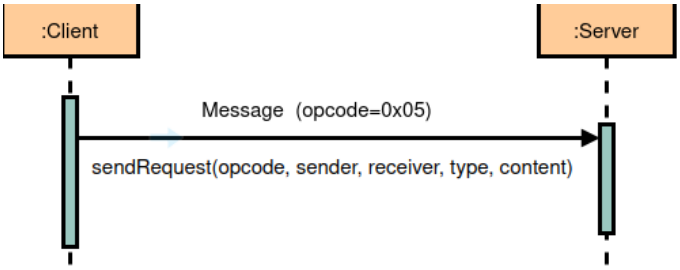


Figure 5: Send Message

### 1.7.6 *Create Group*

To obtain a list of clients, the sender must use the opcode 0x03 with the IP address of the sender and the recipient and the name of the group. To create a group, sender needs to implement base protocol with 0x06 opcode and name of the group. Usually, the hostname and name of group takes 16 bytes of data.

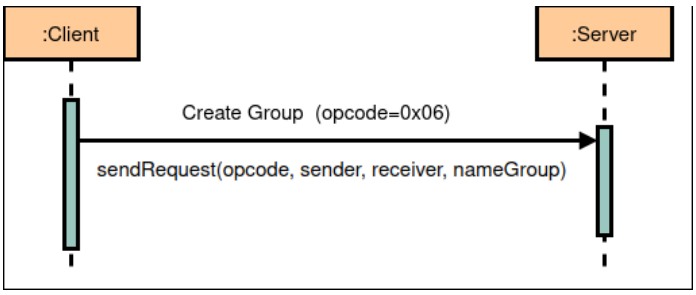


Figure 6: Create Group

### 1.7.7 *Join Group*

In the same way, sender can to apply base protocol with 0x07 opcode and name of the group to join the group.

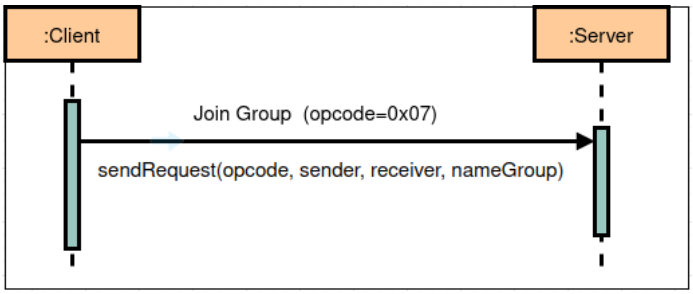


Figure 7: Join Group

### 1.7.8 *Leave Group*

By just changing opcode of above line to 0x08, sender can easily leave the group.

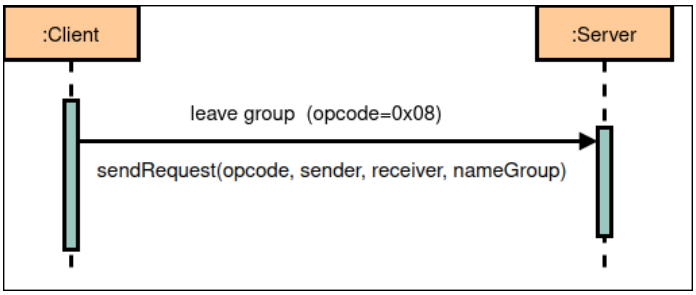


Figure 8: Leave Group

### 1.7.9 *Send Message to group*

To send a message to a group, the sender who wants to send data to the group must specify the basic protocol with opcode 0x09, in the addition of the name of the group of 16 bytes of data and the type of the content and the content itself, where the data type is stored in 4 data bytes and where the content can be of any length.

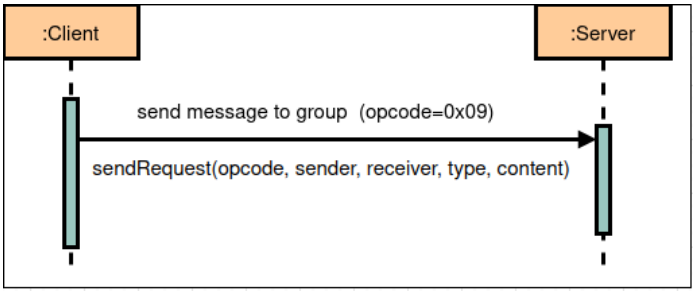


Figure 9: Send Message to group

## 1.8 IMPROVEMENT IN APPLICATION

### 1.8.1 *API-implementation*

To further improve our application in the future, we would use an API for frontend, backend and database and to separate frontend, backend and database from each other.

### 1.8.2 *Usable on multiple devices*

The current version of our application can be used on Linux, Mac and Windows. In the upcoming version we will bring it to other devices like Android, tablets, etc.

### 1.8.3 *Implementation of database*

Our team will integrate this application to store data, backup client and server data and reimplementation of the data.

### 1.8.4 *Implementation on higher protocol*

To use it on a higher protocol, our team will apply it with the https protocol. so we can use it in websites and browsers.

### 1.8.5 *Overall validation*

To improve the viability of our application, we will add some validation features to make the application more realistic and error-free. And know that users are getting the right data.

### 1.8.6 *Validation of port number*

We would add some functions to confirm and validate the port number on the devices that the data is transferred to the right person and device.

# 2

# IMPLEMENTATION

## 2.1 PROBLEM

Forward the message to the specified client. With the help of java suckt it is possible to establish a connection between the server and a client. Through this connection the client and the server can communicate with each other. The client can send a message to the server, and the server can also reply to the client and send retransmissions.

The difficulty is that the client sends the messages to the other client. In this case, the message must be sent from the first client to the server, and then the messages must be forwarded from the server to the correct client. After that, the other client must send the response back to the first client via the server.

## 2.2 SOLUTION

After a lot of thinking, a solution was found that uses the Thread class. The Thread class has to be implemented for the client, so that the client asks the asks the server if there are new messages for the client, if there are new messages they are sent one by one from the server to the client. This process is performed by the thread every second.

## 2.3 SERVER GUI

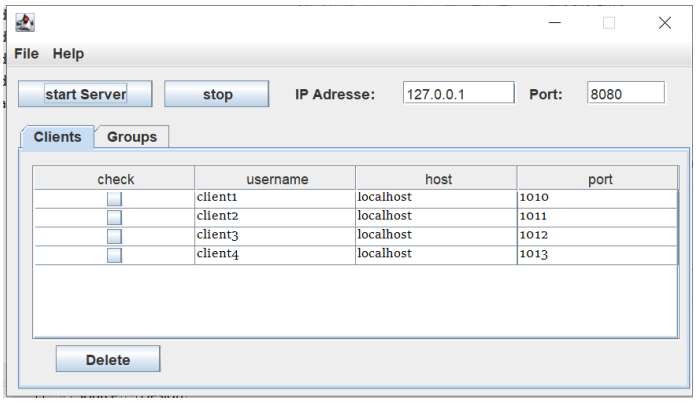


Figure 10: GUI-Server-Clients

In the server mask you can enter a server with its IP address and port $As default 127.0.0.1 : 8080$ and then click the Start server button to start the server. When the server is started, two tables will display all connected

clients and all created groups, respectively. If you click on the Client tab, a table with the following columns will be displayed (checked, username, host, port).

The column with the check boxes are used to select the clients to be deleted. After the clients are selected, you should click the "Delete" button under the table.

The "Groups" table has less columns than the "Clients" table, namely (Check, Name of the group, count of clients). The "Check" column has the same principle of operation as in the "Clients " table, the groups can be selected to be deleted.

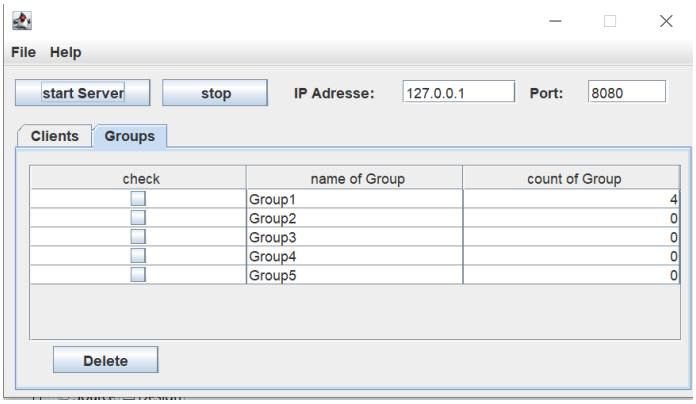


Figure 11: GUI-Server-Groups

when a client is deleted, a message is sent to all other clients with the content (the client is deleted with server). This message *theGroupisdeletedwithserve* is also sent for all clients when a group is deleted.

If you want to stop the server, it must click on the "Stop" button at the top of the mask. when you want to close the server application, click on File –>Exit.

## 2.4 CLIENT GUI

There are two panels in the client interface, namely the configuration panel and the chat panel. In the configuration panel you have to enter the ip address and port of the server but as default values they are entered as follows ( IP-address :127.0.0.1 , port : 8080 ) After that the IP-address and port of client is detected.

The username is set as default by "Client" and "Port" *ExampleUsername* : *Client*65001. After that you have to click on the Save button to start the client. After adding the server and the client the configuration Panel will be displayed and chat Panal will be enabled. In the chat you can select contacts or groups so that the user can communicate with them. The user can also create and join or left a group by himself. If you want to see all clients and Groups , click the Search button The chat panel shows all clients with that the user has written. The messages between the clients are displayed and

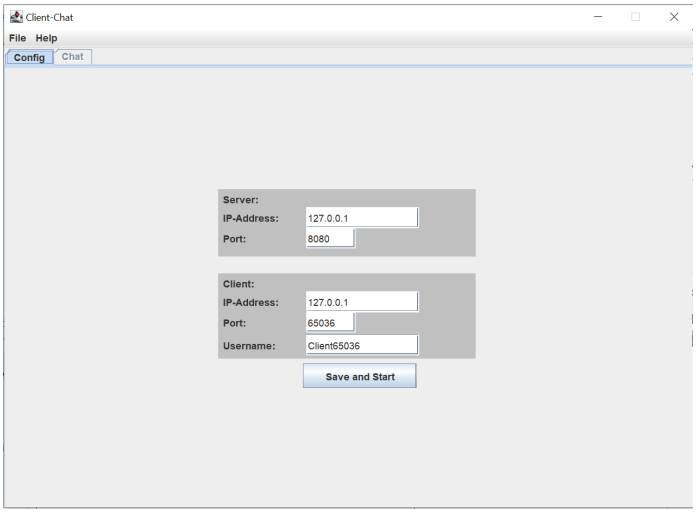


Figure 12: Configuration of Client

where the client can write. The chat also has a button to send different files like video, image and audio.

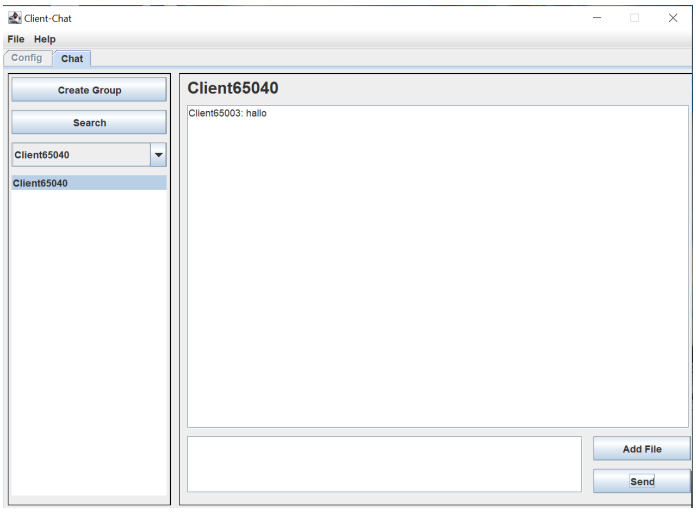


Figure 13: Chat of Client

## 2.5 GROUP

When you click Create Group, a group is created on the server and displayed in a list where you can see all the created groups and clients. In this list you can search for the group you are looking for and find it quickly. The client that created the group will be the first member added to the group. Any client can create a group and then all other clients can join this group, by clicking on the selected group. When a client writes to the group or sends a file, all other clients see this file. Any client can also leave the group at any time

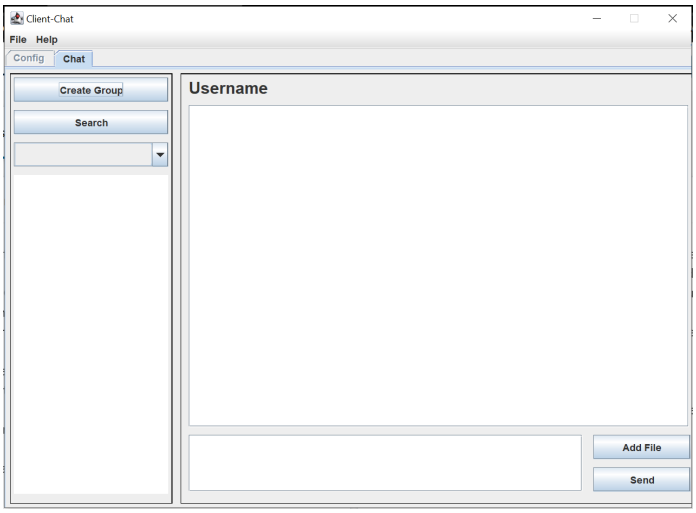


Figure 14: Gruppe

# 3 IMPROVEMENT

## 3.1 DATABASE

In our project an array was used to store the data but if the project needs to be improved, it would be very good and very helpful to use the database, because the use of databases prevents redundancies *multiplestorageofthesameinformation* and inconsistencies *problemswithupdatingmultiplestoredrecords* in the data. When using the database, the client information and its messages are stored permanently, so the client has to register with the server only once, and then he has the possibility to log in. On the other hand, client messages are automatically stored and always available.

Benefit of Using the DB:

- Databases can easily handle multiple users.
- Databases are very reliable because they store information accurately.
- Databases avoid unnecessary information.
- Databases allow powerful and interesting information processing.
- Databases are extensible.
- Databases can handle large amounts of data

Even if you don't need to process large amounts of data yourself (yet), databases are useful for smaller amounts of data. With the capacity to handle so much data, a well-designed database can serve you for many years as it grows with you.

## 3.2 APPLICATION-PROGRAMMING-INTERFACE

it enables data exchange between the server and the client. The API concept was designed by Roy Fielding in 1994. It uses http requests to access information via GET, POST, PUT and DELETE. When using REST interface, a clean separation is created between the backend (server) and the frontend (client). In API, (de-) serialization is passed, and it is done by determining the media-type of REST interface. API has different layers, based on the architecture of the API the layer is determined

- Level 0 "The Swamp of POX": the exchange of information is done only through an API.
- Level 1 "Resources": an API must be implemented for each object.

- Level 2 "HTTP Verbs": the http verbs (GET, POST, PUT and DELETE) are implemented each API
- Level 3 "Hypermedia Controls": a URL is returned for each child object instead of the object.

The disadvantage of layer "0" and "1" is that only the POST request is implemented in the API, which means you can read, write, edit and delete the data with the same request. This means that whoever has read permission can also edit data. The advantage of layer "2" is that with the help of HTTP request can be controlled which right has this request. Layer "3" brings a special advantage that the nested-objects are fetched with an API and not packed in the root object.

## 3.3 VIDEOCONFERENCING

Video conferencing can be very important in our project in the future, because it has many advantages for companies and people, for example 1-Video conferencing can easily overcome spatial distances. Location-independent communication saves valuable time and long journeys, and can thus speed up workflows and decision-making processes enormously. 2-The video conferencing is very helpful for people working abroad and also for students studying abroad. so that you can contact with their families. You can add the ability of screen sharing for video conferencing, so that you can have multiple applications in one application.

## 3.4 VALIDATION

Data validation may save the day of computer programmers, whatever programming language they use. In fact, processing invalid data is a waste of resources at best, and a drama at worst if the problem remains unnoticed and wrong results are used for business. Answer Set Programming is not an exception, but the quest for better and better performance resulted in systems that essentially do not validate data in any way. Even under the simplistic assumption that input and output data are eventually validated by external tools, invalid data may appear in other portions of the program, and go undetected until some other module of the designed software suddenly breaks. an example about validation for our project is that you use validation for username. username must contain a minimum of 3 letters and a maximum of 16 letters, if the user enters more than 16 letters or less than 3 letters, the program must show a pale.

# BIBLIOGRAPHY


[Knu74] Donald E. Knuth. "Computer Programming as an Art." In: *Communications of the ACM* 17.12 (1974), pp. 667–673.

[Sch22] W3 School. "Java Tutorial." In: (2022). URL: https://www.w3schools.com/java/.

[Thr] "Thread (computing)." In: (June 2022). URL: https://en.wikipedia.org/wiki/Thread_(computing).

[lif17] Mike lifeguard. "Git." In: (Oct. 2017). URL: https://en.wikibooks.org/wiki/Git.