# CS421 - Programming Assignment 1

# TextEditor: Client for Collaborative Text Editing

# Fall 2022

**Name: Esad İsmail Tök**

**Student ID: 21801679**

**Department: CS**

**Section: 1**

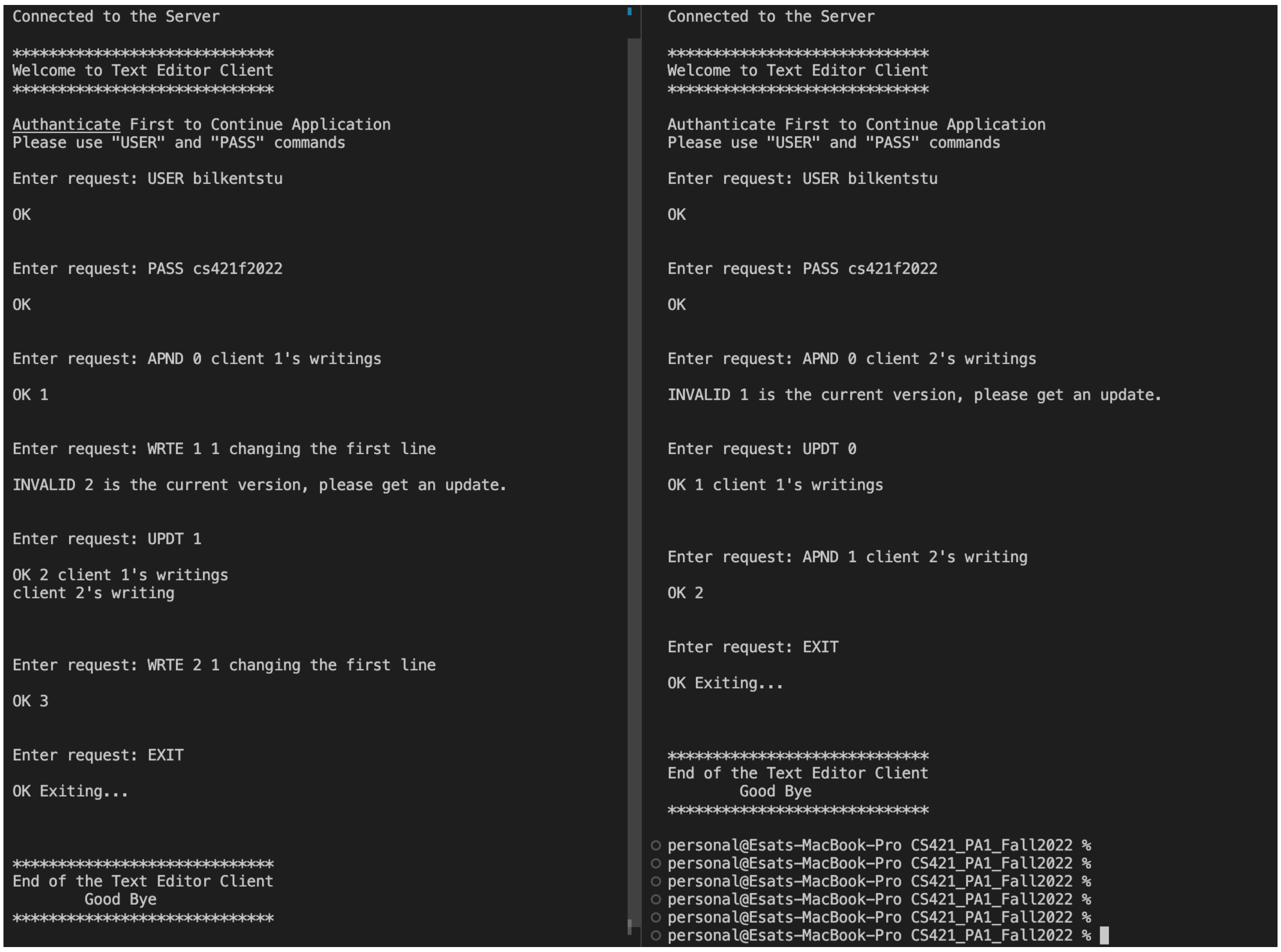
## Overview and Basic Functionality of the Application

I completed my assignment individually. I implemented the assignment in MacOS operating system. I used python as the programming language for this assignment. I have implemented and completed the basic required functionality of the client as well as the extra functionalities that I implemented on both Server.py and TextEditor.py that I will explain in detail in the next section of the report.

The server program runs with the command "*python3 Server.py 127.0.0.1 <port_number>*" in which the port number can be any unused free port number. After the server program is started, the client program which is TextEditor.py can be run using "*python3 TextEditor.py 127.0.0.1 <port_number>*" in which, port number must be the same as the port number that is used to run the server program. The "\r\n" command is automatically concatenated to the input that user enters as a command in the application so that the usability of the application is increased since the user will not be required to enter "\r\n" at the end of the each command. So the general format of a command in the application is: "`<MessageType><Space><Arguments>`".

In order to proceed to the application, user needs to log in using "*USER bilkentstu*" and "*PASS cs421f2022*" commands in which username and password are the same for each user. Then the user can perform the operations such as "*WRTE*", "*APND*", "*UPDT*", "*EXIT*" as well as the additional extra command that I added but in order to better organize the report, the additional commands are introduced in the next section of the report.

After the user logged-in, in order to perform any update in the .txt file, user needs to have the current version of the .txt file. Therefore if the user does not have the current version, a specific error message is displayed to the user together with the current version of the .txt file so that the user can get the appropriate "*UPDT*" and then proceed with his/her commands. Appropriate messages and descriptions are provided to lead the user better throughout the application. In the figures below, I will describe the application in action and show that multiple clients can make updates on the .txt file at the same time.



```
Connected to the Server

****************************
Welcome to Text Editor Client
****************************

Authanticate First to Continue Application
Please use "USER" and "PASS" commands

Enter request: USER bilkentstu

OK


Enter request: PASS cs421f2022

OK


Enter request: APND 0 client 1's writings

OK 1


Enter request: WRTE 1 1 changing the first line

INVALID 2 is the current version, please get an update.


Enter request: UPDT 1

OK 2 client 1's writings
client 2's writing



Enter request: WRTE 2 1 changing the first line

OK 3


Enter request: EXIT

OK Exiting...



****************************
End of the Text Editor Client
        Good Bye
****************************
```

```
Connected to the Server

****************************
Welcome to Text Editor Client
****************************

Authanticate First to Continue Application
Please use "USER" and "PASS" commands

Enter request: USER bilkentstu

OK


Enter request: PASS cs421f2022

OK


Enter request: APND 0 client 2's writings

INVALID 1 is the current version, please get an update.


Enter request: UPDT 0

OK 1 client 1's writings



Enter request: APND 1 client 2's writing

OK 2


Enter request: EXIT

OK Exiting...



****************************
End of the Text Editor Client
        Good Bye
****************************

personal@Esats-MacBook-Pro CS421_PA1_Fall2022 %
personal@Esats-MacBook-Pro CS421_PA1_Fall2022 %
personal@Esats-MacBook-Pro CS421_PA1_Fall2022 %
personal@Esats-MacBook-Pro CS421_PA1_Fall2022 %
personal@Esats-MacBook-Pro CS421_PA1_Fall2022 %
personal@Esats-MacBook-Pro CS421_PA1_Fall2022 %
```

*Figure 1: Running the application in multiple clients*

*Figure 1* shows how multiple clients together can make updates on the .txt file. *Figure 1* displays 2 consoles as divided, left side console is the console of the first client and the right side console is the second client. First of all both clients are logged-in using the same username and password as expected. At first, the version of the file is 0 initially. First client makes an "*APND*" and changes the file. Then the second client tries to makes an append providing his version number as 0 but since the version of the .txt file is changed due to the changes of the first client, request is not accepted and update required. The the second client makes the update using "*UPDT*" command and providing his current version and therefore getting the latest version as described in the Project Assignment. After that the second client can successfully make an “*APND*”. Then the first client tries to use the “*WRTE*” command providing the version number 1 which is the current version number of the first client. But since the latest version is now 2 due to the changes of the second client, the request of the first client is rejected and again appropriate message is displayed. Then the first client gets the appropriate version number using “*UPDT*” command as well and the providing the latest version, first client can perform the “*WRTE*” command. At the end both clients terminate the application using the “*EXIT*” command.

## Additional Functionalities and Features

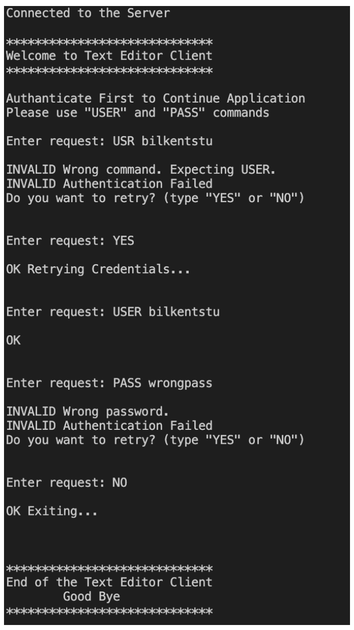
In this section of the report I will explain the additional functionalities, corrections, commands, and features that I implemented. In order to implement those functionalities, the Server.py program is modified and the Server.py program in my submission is the modified code. The client program which is TextEditor.py is implemented so that it  can work with both modified and unmodified, original version of the server program.

**Fixes in “*WRTE*” Command:** In the provided server program the “*WRTE*” command were using the line number as the index but it is more appropriate to use it as the real line number and convert it to index in the function. so I modified it so that now the command accepts the line number, for example “*WRTE 0 2 test string*” adds the string to the line number 2. Also the command was not using file.flush() function so there were a problem in writing and I fixed it as well.

**Log-in Enhancements:** When a user enters credentials using “*USER*” and “*PASS”* commands, if he/she enters wrong username or password, or if he/she misspells or enters another command, the program used to terminate the application but I fixed it so that now in terms of any problem in the log-in phase, server asks if the user wants to continue trying to log-in or wants to terminate the program. So that it is more useful now.

*Figure 2* below shows the new log-in mechanism.

In *Figure 2,* first the user misspells the “*USER*” command and the question asked for the proceeding of the application. User wants to continue so enters “YES” and then after correctly entering username, user enters wrong password and then again the menu is shown. This time user want to quit the application and types “NO” which end the program and exits.



```
Connected to the Server

*****************************
Welcome to Text Editor Client
*****************************

Authanticate First to Continue Application
Please use "USER" and "PASS" commands

Enter request: USR bilkentstu

INVALID Wrong command. Expecting USER.
INVALID Authentication Failed
Do you want to retry? (type "YES" or "NO")


Enter request: YES

OK Retrying Credentials...


Enter request: USER bilkentstu

OK


Enter request: PASS wrongpass

INVALID Wrong password.
INVALID Authentication Failed
Do you want to retry? (type "YES" or "NO")


Enter request: NO

OK Exiting...



*****************************
End of the Text Editor Client
        Good Bye
*****************************
```

*Figure 2: Login Enhancement*

**Unknown Command Enhancements:** In the given version of the server program, when an unknown command is entered the program used to terminate. But with the additions I made the program asks the user to continue or exit. I added two new commands for this purpose. One is "*YES*" the other is "*NO*". Those commands can only be used after an unknown command is entered, otherwise program disables the user to use those commands. The user does not provide version number when using those commands since those command have nothing to the version. Also when "*YES*" or "*NO*" commands are asked to user, user can only give those commands but not others. The program ensures that condition.

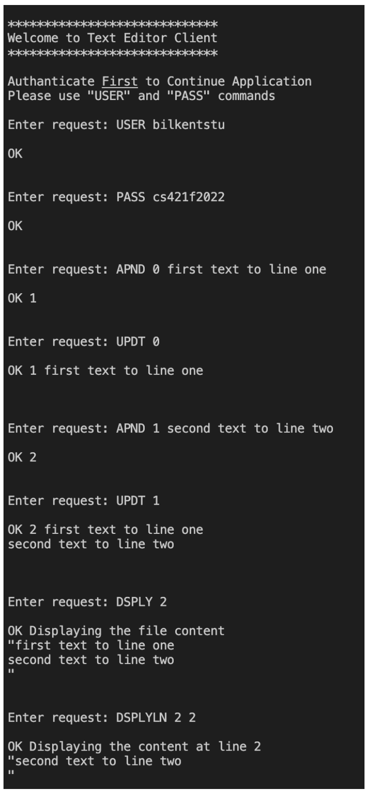
**"*DSPLY*" Command:** I added a new command to display the entire text in the .txt file. The usage of the command is "*DSPLY <version_number>*". The version number needs to be the latest otherwise an update is needed. This command does not change the version of the file since no updates are made with this command.

**"*DSPLYLN*" Command:** I added a new command to display a specific line in the .txt file. The usage of the command is "*DSPLY <version_number> <line_number>*". If there is no such line in the file, appropriate message is sent to the user and command is rejected. The version number needs to be the latest otherwise an update is needed. This command does not change the version of the file since no updates are made with this command.

**"*CLR*" Command:** I added a new command to clear the content of the .txt file. The usage of the command is "*CLR <version_number>*". The version number needs to be the latest otherwise an update is needed. This command increases the version number since it makes an update in the file.

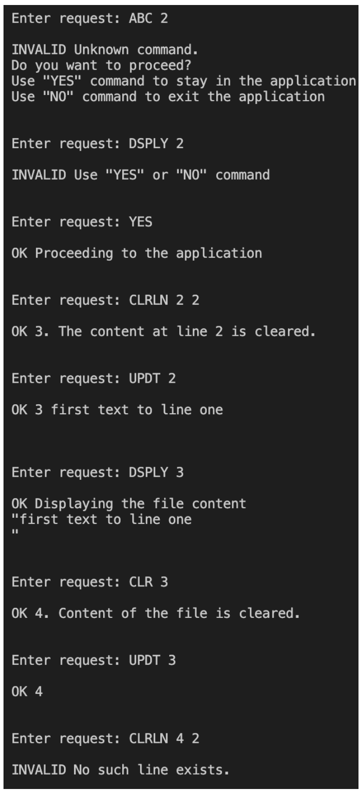
**"*CLRLN*" Command:** I added a new command to clear a specific line in the .txt file. The usage of the command is "*CLR <version_number> <line_number>*". If there is no such line in the file, appropriate message is sent to the user and command is rejected. The version number needs to be the latest otherwise an update is needed. This command increases the version number since it makes an update in the file.

*Figure 3, Figure 4, and Figure 5* below display the usage of additional functionalities and features that I added to the application. All 3 figures belongs to the same run. *Figure 4* is the continuation of *Figure 3,* and *Figure 5* is the continuation of *Figure 4.*



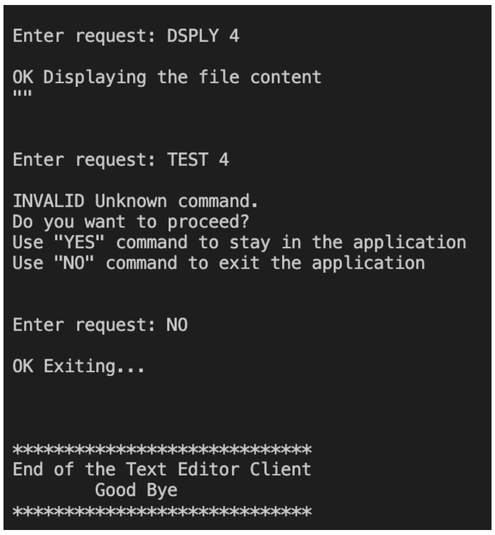
```
*****************************
Welcome to Text Editor Client
*****************************

Authanticate First to Continue Application
Please use "USER" and "PASS" commands

Enter request: USER bilkentstu

OK


Enter request: PASS cs421f2022

OK


Enter request: APND 0 first text to line one

OK 1


Enter request: UPDT 0

OK 1 first text to line one



Enter request: APND 1 second text to line two

OK 2


Enter request: UPDT 1

OK 2 first text to line one
second text to line two



Enter request: DSPLY 2

OK Displaying the file content
"first text to line one
second text to line two
"


Enter request: DSPLYLN 2 2

OK Displaying the content at line 2
"second text to line two
"
```

Figure 3: First Part of the Run



```
Enter request: ABC 2

INVALID Unknown command.
Do you want to proceed?
Use "YES" command to stay in the application
Use "NO" command to exit the application


Enter request: DSPLY 2

INVALID Use "YES" or "NO" command


Enter request: YES

OK Proceeding to the application


Enter request: CLRLN 2 2

OK 3. The content at line 2 is cleared.


Enter request: UPDT 2

OK 3 first text to line one



Enter request: DSPLY 3

OK Displaying the file content
"first text to line one
"


Enter request: CLR 3

OK 4. Content of the file is cleared.


Enter request: UPDT 3

OK 4


Enter request: CLRLN 4 2

INVALID No such line exists.
```

Figure 4: Second Part of the Run



```
Enter request: DSPLY 4

OK Displaying the file content
""


Enter request: TEST 4

INVALID Unknown command.
Do you want to proceed?
Use "YES" command to stay in the application
Use "NO" command to exit the application


Enter request: NO

OK Exiting...



*****************************
End of the Text Editor Client
       Good Bye
*****************************
```

Figure 5: Third Part of the Run