



HACETTEPE UNIVERSITY  
COMPUTER ENGINEERING DEPARTMENT

BBM365 2022 FALL

---

## Assignment-2

---

November 19, 2022

Group Number: 37

*Tuğrul ACAR*  
2210356144

*Alper SOLMAZ*  
2200356039

## Problem

In this project, we are expected to develop a simple message box system. The system stores and protect the messages left by visitors. Visitors will leave messages to the registered users. Registered users can see the messages by using message codename, message password and their username and password. A user cannot see other user's messages. The user must know the message codename and password.

## Important Notes About the Assignment

- We used JAVA 14 version. Our program works as specified on this version.
- Detailed description of all functions given in the code.
- We have 4 preregistered users in the system. Users username and password informations given in the table below.
- There is no user registration system in the program.

Username	Password
freya	fourdoor65
thor	siskoadam13
odin	sinsitanri14
kratos	adamgibiadam

## Solution for the Problem

We have 5 java classes in this project:

- User class creates instances for registered users with attributes username and password.
- Message class create instances for each message which is left by visitors.
- Page class includes implementations of GUI.
- Util class has different methods, which are IO operations, encryption-decryption and some split operations.



Figure 1 UML Diagram

Program uses 2 txt file to store data. *messages.txt* stores messages left by visitors and *users.txt* stores users informations. Format of these files:

**messages.txt**

*username1 - message\_code1 - message\_password1 - message1*  
*username2 - message\_code2 - message\_password2 - message2*  
...

**users.txt**

*username1 - user\_password1*  
*username2 - user\_password2*  
...

message\_passwords and user\_passwords stored in hashed form. Also message stored as encrypted. Hashed values and encrypted message encoded as **Base64** to be able to written into txt file.

We preferred to use **MD5** Hash Algorithm to store passwords safely. We used **DES** Algorithm with **ECB** encryption mode to store messages as encrypted.

We used **Java Swing** for the GUI. There are 4 different pages. These are *Home page*, *Message View page*, *Register page* and *View page*.

## Home Page

Home page is the main page of the program. There are two buttons to direct a user to the access page to view a message and the register page to leave a message to the system.

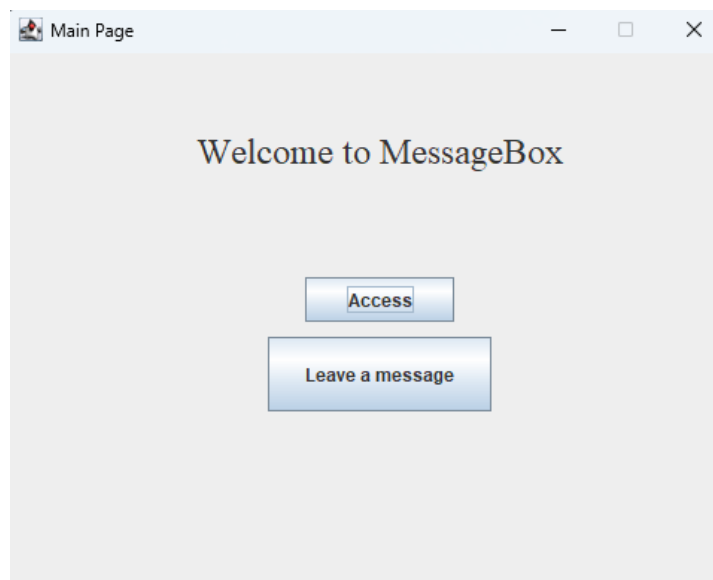
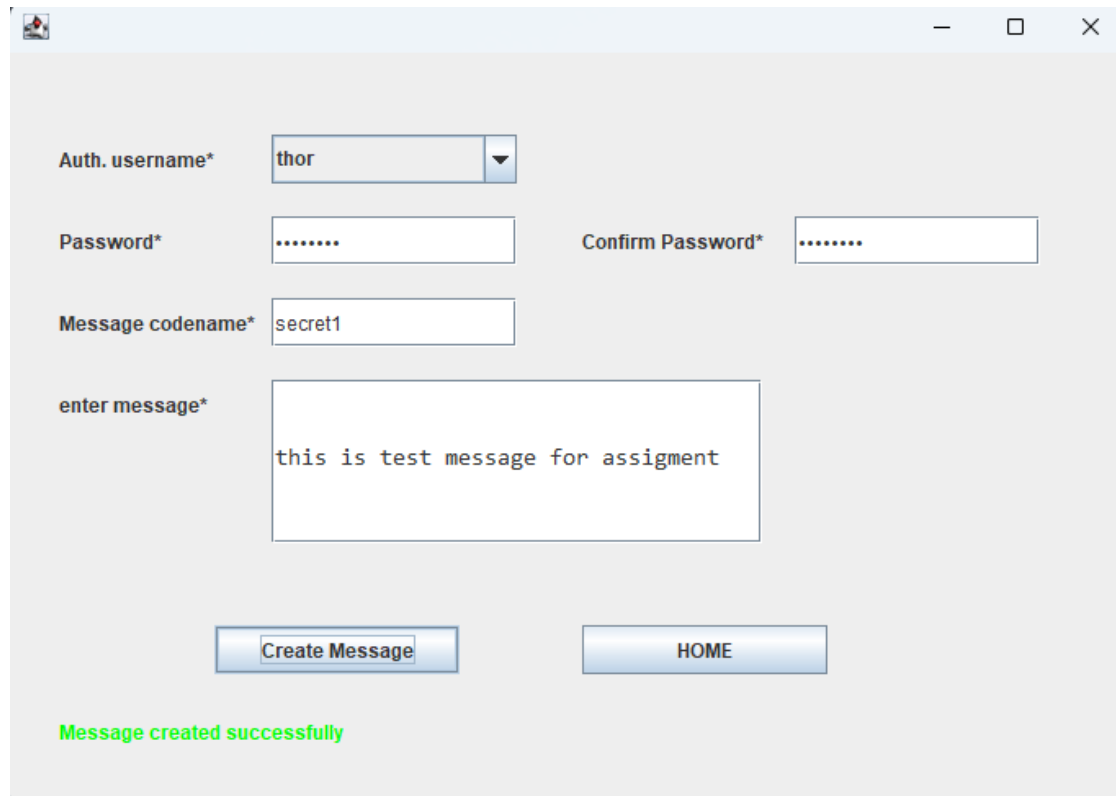


Figure 2 Home page view

## Message Register Page

To create new messages, there is a message register view that includes a registration form. This form include corresponding fields for message content, message id, and password. Also include the authorized username. On the form page, there are two buttons, one for sending the form to create a new message, and the other one to return to the home page.

There are some constraints. Fields cannot be left empty. Message passwords must be at least 8 characters. 2 password field should be matched. Otherwise there will be printed a red error message left bottom of the page. If all conditions are satisfied message will be created with a green information message printed left bottom of the page.



The screenshot shows a web application window titled "Message Register View". It contains a registration form with the following fields and values:

- Auth. username\***: A dropdown menu showing "thor".
- Password\***: A text input field containing ".....".
- Confirm Password\***: A text input field containing ".....".
- Message codename\***: A text input field containing "secret1".
- enter message\***: A large text area containing "this is test message for assigment".

At the bottom of the form, there are two buttons: "Create Message" and "HOME". Below the buttons, a green message "Message created successfully" is displayed.

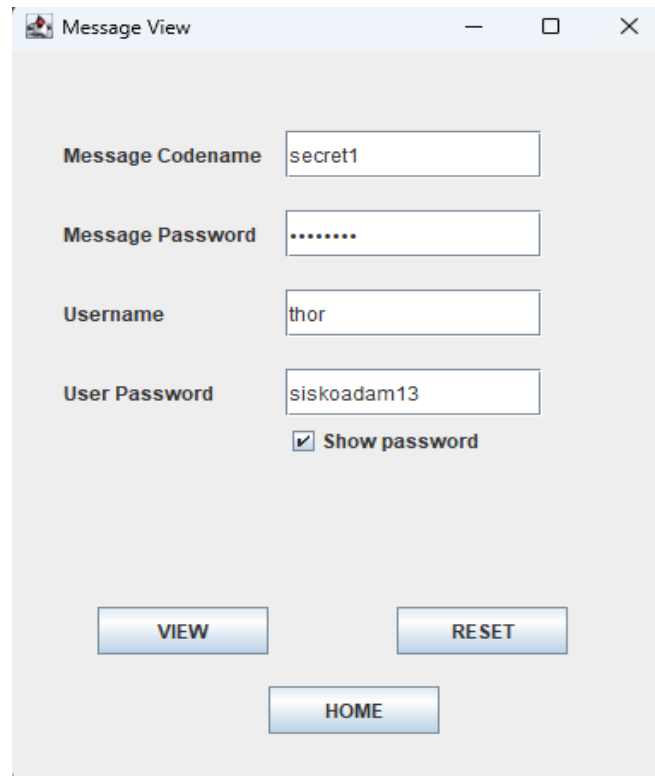
Figure 3 Message Register View

## Access View Page

This page includes a form to get the credentials from the user. There are 4 fields: message id and password, username, and user password. The page include a "View" button. If the given credentials are correct and the user has authorization, s/he access the message. There is "Home" button to return to the home page and a "Reset" button to clear all fields.

## Message View Page

After the access page, user directed to the message page, which includes message. The message should be on the frame, and there should be a button to return.



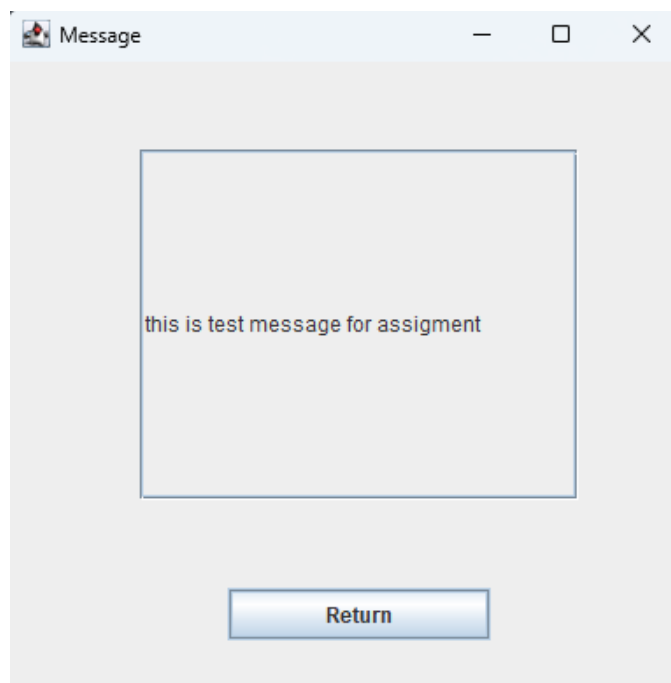
A screenshot of a web application window titled "Message View". The window has a light gray background and standard window controls (minimize, maximize, close) in the top right corner. It contains four input fields: "Message Codename" with the value "secret1", "Message Password" with masked characters ".....", "Username" with the value "thor", and "User Password" with the value "siskoadam13". Below the "User Password" field is a checkbox labeled "Show password" which is checked. At the bottom of the window are three buttons: "VIEW", "RESET", and "HOME".

Message Codename	secret1
Message Password	.....
Username	thor
User Password	siskoadam13
<input checked="" type="checkbox"/> Show password	

VIEW      RESET

HOME

Figure 4 Access Page View



A screenshot of a web application window titled "Message". The window has a light gray background and standard window controls (minimize, maximize, close) in the top right corner. It displays a large rectangular box containing the text "this is test message for assigment". Below this box is a single button labeled "Return".

this is test message for assigment

Return

Figure 5 Message Page View

## References

- <https://www.javatpoint.com/java-swing>
- <https://www.baeldung.com/java-md5>
- <https://www.geeksforgeeks.org/data-encryption-standard-des-set-1/>