Gebze Technical University Computer Engineering Department

CSE 222 – 2023 Spring

Homework 1 Report

Ethem Mengaaslan

200104004016

System Requirements:

First of all, this assignment is about creating a social media application which runs on the terminal screen, and that has the basic operations as a modern day social media app.

It has to have at least the following functionalities: log-in, log-out, follow, post, comment, like, view profile, direct message etc.

Here are the requirements for this project.

Account class: This class represents the only user type in this program. It has the methods for the actions that a user is supposed to be able to do like; posting, sending direct messages to another users, leaving a comment on another users post etc.

This Class also stores the following / follower user information, and also the sent and taken messages.

Post Class: This class represents a post that a user can post. Similar to any other social media platform, it has like and comment mechanisms. This class stores the data of the post such as the main content of the post, number of likes, the comments on it etc.

Like and Comment classes (derived from an interaction class): These classes was necessary because storing a comment as a string would be complicated and would also be against the object oriented mentality. So the interactions made on a post are seperated as like and comment objects this way.

Message Class: This class represents a direct message, with the following information; the message content, the sender information, the receiver information, an id for the message.
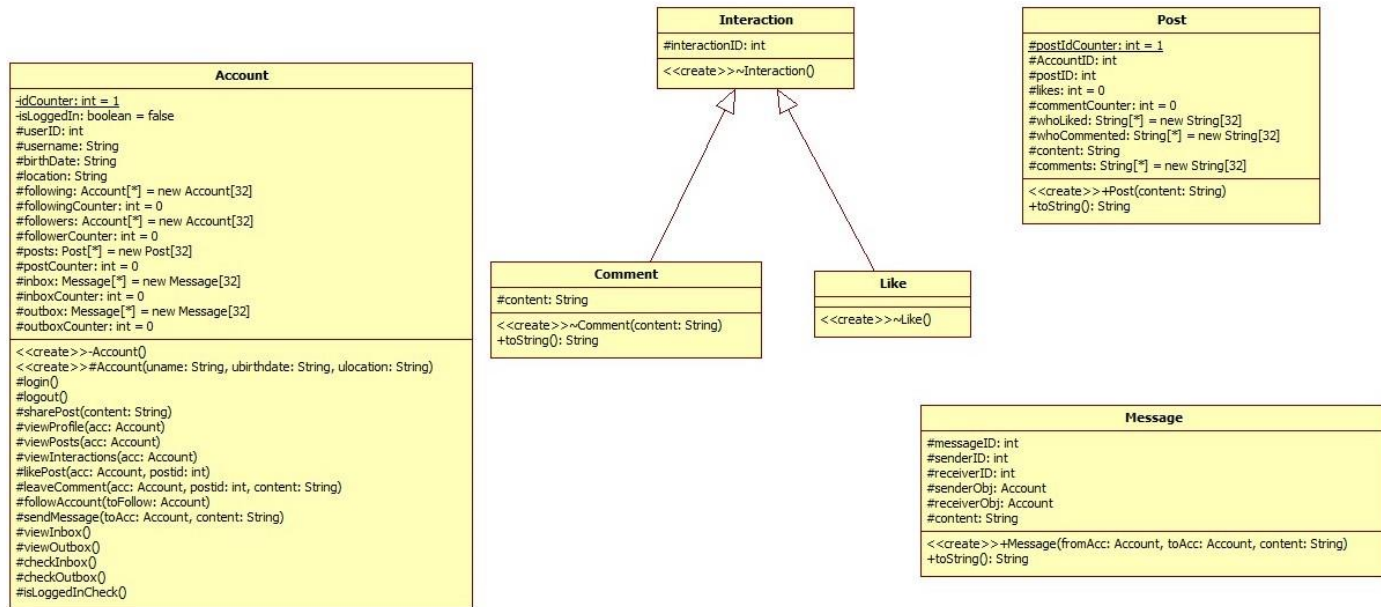
The Problem Solution Approach:

Firstly, I have drawn the given diagram on a paper, and wrote down the functionality that came to my mind which would be necessary for the program to operate. Then I divided that functionality into methods, and those methods to classes.

Then I implemented the is a relations in the given diagram, creating the classes with the necessary inheritances etc. .
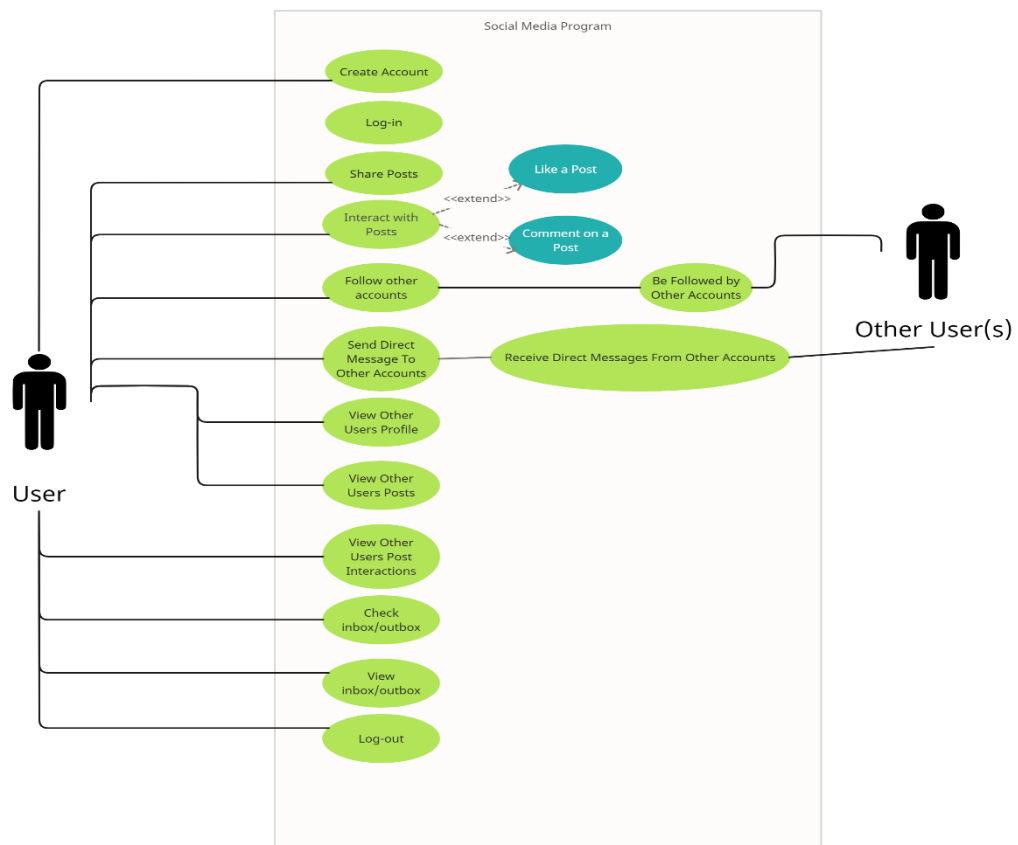
At the end, i thought about each class in an abstract frame, to make the classes independent from each other. At the end, I had a very clear blueprint of the program, than I started to implement the methods one by one.

The Class Diagram

Interaction

#interactionID: int

<<create>>~Interaction()

Post

#postIdCounter: int = 1
#AccountID: int
#postID: int
#likes: int = 0
#commentCounter: int = 0
#whoLiked: String[*] = new String[32]
#whoCommented: String[*] = new String[32]
#content: String
#comments: String[*] = new String[32]

<<create>>+Post(content: String)
+toString(): String

Account

-idCounter: int = 1
-isLoggedIn: boolean = false
#userID: int
#username: String
#birthDate: String
#location: String
#following: Account[*] = new Account[32]
#followingCounter: int = 0
#followers: Account[*] = new Account[32]
#followerCounter: int = 0
#posts: Post[*] = new Post[32]
#postCounter: int = 0
#inbox: Message[*] = new Message[32]
#inboxCounter: int = 0
#outbox: Message[*] = new Message[32]
#outboxCounter: int = 0

<<create>>-Account()
<<create>>#Account(uname: String, ubirthdate: String, ulocation: String)
#login()
#logout()
#sharePost(content: String)
#viewProfile(acc: Account)
#viewPosts(acc: Account)
#viewInteractions(acc: Account)
#likePost(acc: Account, postid: int)
#leaveComment(acc: Account, postid: int, content: String)
#followAccount(toFollow: Account)
#sendMessage(toAcc: Account, content: String)
#viewInbox()
#viewOutbox()
#checkInbox()
#checkOutbox()
#isLoggedInCheck()

Comment

#content: String

<<create>>~Comment(content: String)
+toString(): String

Like

<<create>>~Like()

Message

#messageID: int
#senderID: int
#receiverID: int
#senderObj: Account
#receiverObj: Account
#content: String

<<create>>+Message(fromAcc: Account, toAcc: Account, content: String)
+toString(): String

(a high resolution version is in the files)

The Use Case Diagram

(A high resolution copy is in the files)

Test Cases:

Note: I used empty println statements for the output to be pretty, so that our lovely T.A.s could read them easily. Please ignore them.

```java
Account sibelgulmez = new Account( uname: "sibelgulmez", ubirthdate: "01.01.1999", ulocation: "Darica");
Account gokhankaya = new Account( uname: "gokhankaya", ubirthdate: "03.05.1988", ulocation: "Tuzla");
Account gizemsungu = new Account( uname: "gizemsungu", ubirthdate: "05.06.1998", ulocation: "Pendik");
System.out.println("");

sibelgulmez.login();
System.out.println("");

sibelgulmez.sharePost( content: "I love OOP!");
sibelgulmez.sharePost( content: "Python is for kids :P");
System.out.println("");
```

```
Creating Accounts...

An account with name sibelgulmez has been created
An account with name gokhankaya has been created
An account with name gizemsungu has been created

Logged in as sibelgulmez

Shared a post.
Shared a post.
```

```java
sibelgulmez.login();
System.out.println("");

sibelgulmez.sharePost( content: "I love OOP!");
gokhankaya.login();
sibelgulmez.sharePost( content: "Python is for kids :P");
System.out.println("");
```

A user (gokhankaya) tries to log-in while another user (sibelgulmez) is logged-in. Since 2 accounts can not be logged-in at the same time, the gokhankaya's log-in process is not accepted.

```
Logged in as sibelgulmez

Shared a post.
First log out from the current user to log in to your account...
Shared a post.
```

The gokhankaya is prohibited to log-in in this case, but the program is not terminated. This mistaken log-in is not a cause for the program to terminate, it just gives a warning message

that two account's can not be online at the same time. The current user still can surf on the social media program.

```
sibelgulmez.logout();                        Logging out from account sibelgulmez
System.out.println("");


gokhankaya.login();                          Logged in as gokhankaya
System.out.println("");
```

Above is the valid way to log-in.

Below is what happens when you try to operate with an account without logging-in first.

```
sibelgulmez.login();
System.out.println("");

gokhankaya.viewProfile(gizemsungu);
```

```
Logged in as sibelgulmez


Please Log-In to the account first. The account you are trying to operate is not logged in. Terminating program...
```

```
gokhankaya.viewPosts(sibelgulmez);
System.out.println("");


gokhankaya.likePost(sibelgulmez, postid: 1);
gokhankaya.leaveComment(sibelgulmez, postid: 1, content: "OOP Rocks!");
System.out.println("");
```

```
Viewing sibelgulmez posts...
(Post ID:1)  I love OOP!

(Post ID:2)  Python is for kids :P



Liking (Post ID: 1) of sibelgulmez...
Adding a comment on sibelgulmez's post...
```

Below is the test of messaging functionality.

```
gokhankaya.sendMessage(gizemsungu, content: "How are you doing??");
System.out.println("");



gokhankaya.logout();
System.out.println("");
```

```
Sending a message to gizemsungu...

Logging out from account gokhankaya
```

```
gizemsungu.login();
System.out.println("");

gizemsungu.checkOutbox();
System.out.println("");

gizemsungu.checkInbox();
System.out.println("");
```

```
Logged in as gizemsungu

Checking outbox...
There is/are 0 message(s) in the outbox.

Checking inbox...
There is/are 1 message(s) in the inbox.
```

```
gizemsungu.viewInbox();
System.out.println("");
```

```
Viewing inbox...
--------------------------------------------
Message ID: 1
From: gokhankaya
To: gizemsungu
Message: How are you doing??
```

```
gizemsungu.likePost(sibelgulmez, postid: 1);
gizemsungu.likePost(sibelgulmez, postid: 2);
System.out.println("");

gizemsungu.viewInteractions(sibelgulmez);
System.out.println("");
```

```
Viewing sibelgulmez post interactions...
---------------------------------
(Post ID:1)  I love OOP!
The post has 2 like(s)
The post was liked by the following accounts: gokhankaya,
gizemsungu,
The post has 1 comment(s)
Comment 1: 'gokhankaya' said: OOP Rocks!
---------------------------------
(Post ID:2)  Python is for kids :P
The post has 1 like(s)
The post was liked by the following accounts: gizemsungu,
The post has 0 comment(s)
```

Some more test cases…

```
gokhankaya.login();
gokhankaya.viewProfile(gizemsungu);
gokhankaya.leaveComment(gizemsungu, postid: 2, content: "Nice!");
gokhankaya.sendMessage(gizemsungu, content: "Hello!");
gokhankaya.logout();
```

```
Logged in as gokhankaya

Viewing gizemsungu's Account...
----------------------------------
Username: gizemsungu
Location : Pendik
Birth Date: 05.06.1998
This Account has 2 post(s).
gizemsungu is following 0 account(s) and has 2 follower(s).

Adding a comment on gizemsungu's post...

Sending a message to gizemsungu...

Logging out from account gokhankaya
```

```java
gizemsungu.login();
System.out.println("");

gizemsungu.viewProfile(gizemsungu);
System.out.println("");

gizemsungu.checkInbox();
System.out.println("");

gizemsungu.viewInbox();
```

```
Logged in as gizemsungu

Viewing gizemsungu's Account...
----------------------------------
Username: gizemsungu
Location : Pendik
Birth Date: 05.06.1998
This Account has 2 post(s).
gizemsungu is following 0 account(s) and has 2 follower(s).

Checking inbox...
There is/are 2 message(s) in the inbox.

Viewing inbox...
--------------------------------------------
Message ID: 1
From: gokhankaya
To: gizemsungu
Message: How are you doing??

Message ID: 2
From: gokhankaya
To: gizemsungu
Message: Hello!
```