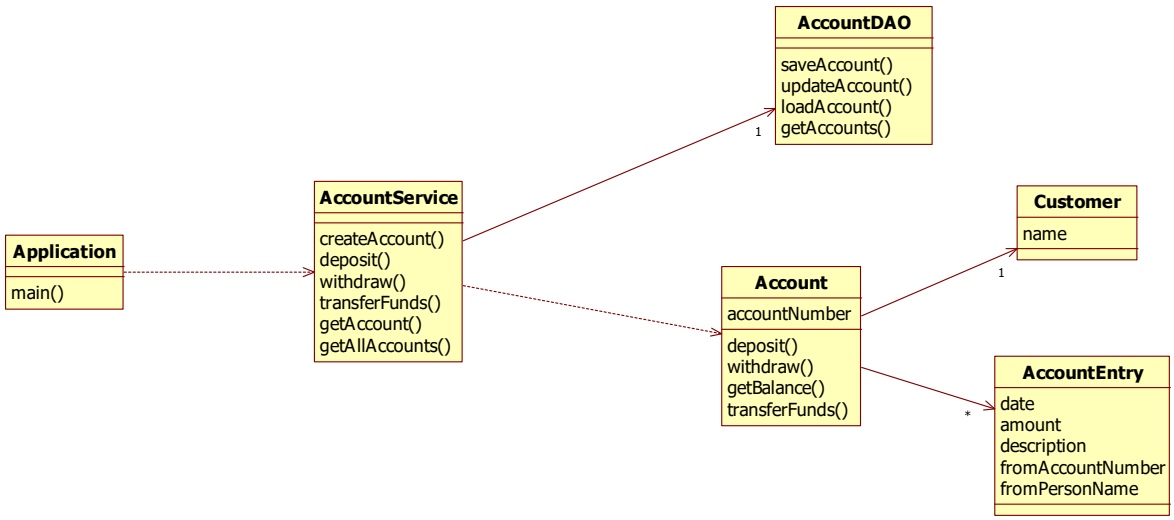
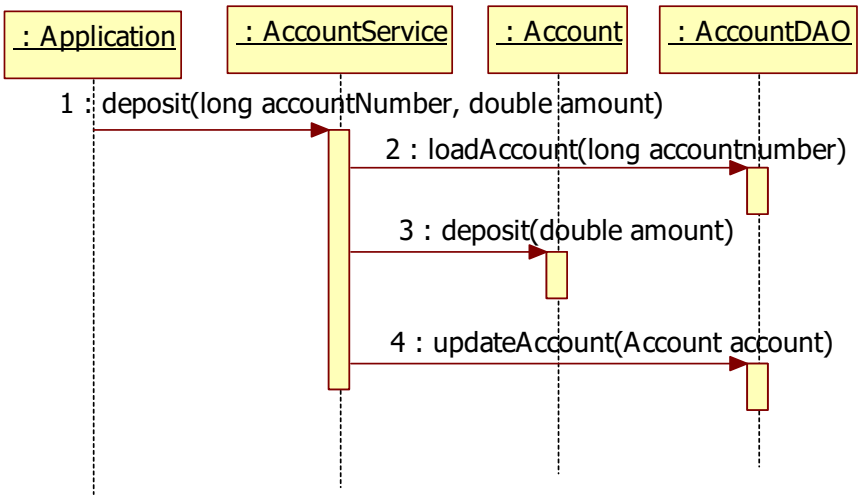
Lab 4

a. Given is the following bank application:





We want to add new functionality whenever the Account balance is changed. Implement the observer pattern in the given code. Add the following observers:

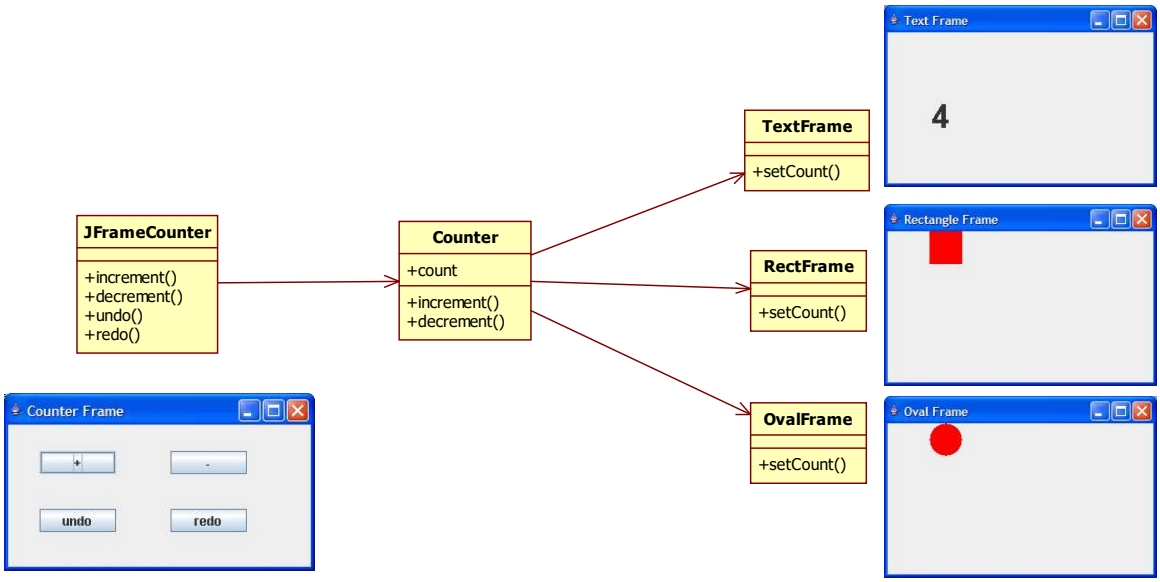
- Add a Logger class that logs every change to an Account. (The Logger should do a simple System.out.println() to the console)
- Add a SMSSender that sends a SMS at every change to an Account. (The SMSSender should do a simple System.out.println() to the console)
- Add an EmailSender that sends an email whenever a new Account is created. (The EmailSender should do a simple System.out.println() to the console)

Draw the modified class diagram with the observer pattern (using the pull model) applied.

b. Draw a sequence diagram that shows how your new design works. On the sequence diagram show the following scenario:
   1. First create a new account
   2. Then deposit $80 on this new account

c. Implement the observer pattern using the pull model in the given code.

In the previous lab the following counter application was given:



The problem with this application is that the Counter class is tightly coupled with the UI classes TextFrame, RectFrame and OvalFrame. If we want to add another view of the counter, for example a binaryFrame that shows the value of the counter in binary, then we have to change the increment() and decrement() method in the Counter.

d. Draw the class diagram of a better design (using the push model) so that it will be much easier to add different views of the counter value. So your diagram should

e. Draw the sequence diagram (using the push model) that shows the following scenario:
   1. The user clicks the increment button
   2. The user clicks the decrement button

f. Implement your new design in the given code in Java. Your solution should only contain the observer pattern (using the push model) and not the command pattern.

g. Now modify the solution of part e such that your solution contains both the command and the observer pattern in the same application

What to hand in?

1. A jpeg picture of part a, b, d and e
2. A zip file containing the project of part c, f and g
3. A **readme.txt** file with the following statement:

   I hereby declare that this submission is my own original work and to the best of my knowledge it contains no materials previously published or written by another person. I understand that if I submit one or more solutions that I did not create myself I will fail the course with an NC.

   [your name as signature]