Place

Part 6: Final Report

CSCI 4448

Elizabeth Boese

Ross Blassingame

Michael Tang

Josh Jenkins

Griffith Neumark

Questions 1 and 2

1,2)

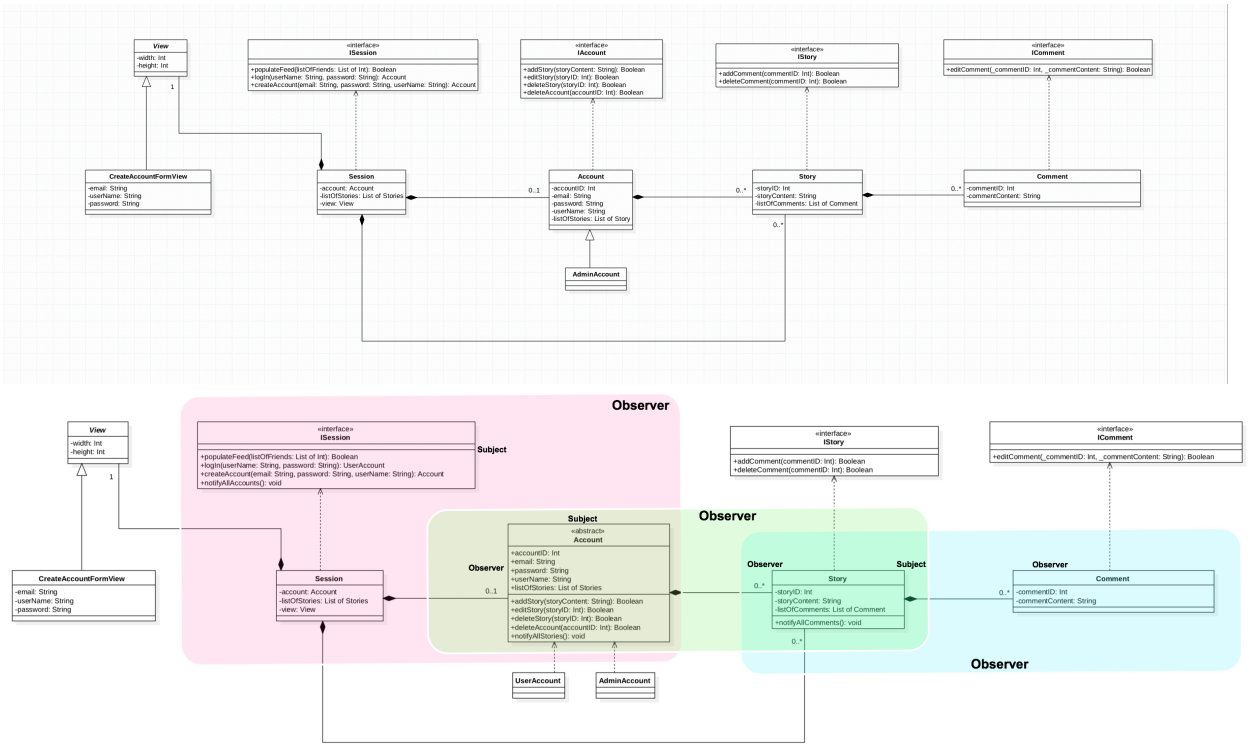
UC	Requirement	Implemented
UR-01	As a user, I want to view my profile	No
UR-02	As a user, I want to send a private message to another user	No
UR-03	As a user, I want to post to a feed so everyone can see	Yes
UR-04	As a user, I should be able to add friends	No
UR-05	As a user, I should be able to create an account	Yes
UR-06	As a user, I should be able to delete my own account	No
UR-07	As an admin, I want to be able to delete an account	No
UR-08	As a user, I want to be able to delete posts	No
UR-09	As a user, I want to be able to edit my post	No
UR-10	As an admin, I want to be able to delete a post	No
UR-11	As an admin, I want to be able to edit a post	No
UR-12	As a user, I want to be able to comment on a post	Yes
UR-13	As a user, I want to be able to comment on a post	No
UR-14	As a user, I want to edit my profile	No
UR-15	As a user, I should be able to log in	Yes
UR-16	As a user, I should be able to view my feed	Yes
UR-17	As a user, I want to be able to remove friends	No
UR-18	AS a user, I want to be able to post photos	No
NRF-01	DB to store user information	Yes

UC	Requirement	Implemented
NRF-02	All users must have a minimum of 8-digit password	Yes
NFR-03	All users must have a unique email to create an account	Yes
NFR-04	No profanity on posts	No
NFR-05	Limit posts to 1,000 characters	No
NFR-06	Hide a post over 100 characters by default	No

Question 3

The first picture below is our part 2 class diagram, and the last picture (colored) is the final class diagram, which is also our class diagram from part 3. The final class diagram is unchanged from the part 2 class diagram because we spent a lot of time on the planning of our system, which made everything a lot easier moving forward. The only differences between the two class diagrams below is that the final class diagram, the one on the bottom, is color-coded to include some of the design patterns we implemented, and a few classes have additional methods to help implement the design patterns. We talked in depth about the design patterns we chose in part 3. The observer pattern is the pattern we made the most use out of, since we have three separate instances of the observer pattern in our code.

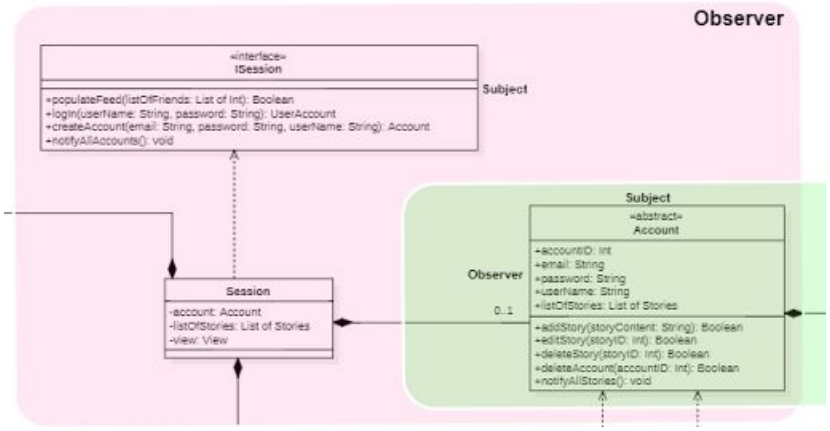
Spending time on up-front planning may seem like a waste of time at first – it definitely did to us. However, as the project comes to a close, it's becoming more and more clear how much time we saved in the long run by spending that time on planning up front.



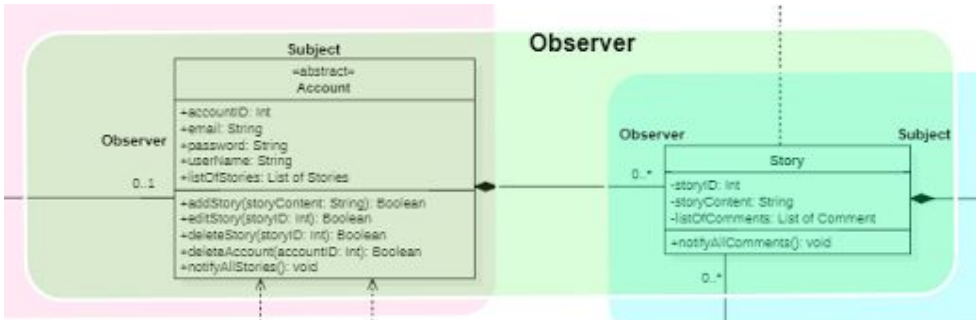
Project Part 6

Question 4.

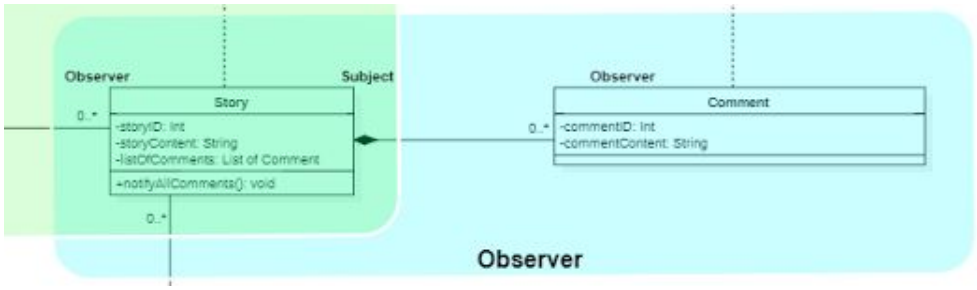
Observer pattern: Account notification



Observer pattern: Story notification



Observer pattern: Comment notification



The main design pattern used when planning our application was observer. This was implemented across multiple classes, and operates similarly between them. This pattern is used on construction of an Account, Story, or Comment to notify the session of its creation, and provide an instant feedback to the UI. The pattern functions the same for creation of these entities, in that it displays a message to the feed containing the object's information, but the

Story and Comment objects have edit and delete methods which also must trigger the notification to the feed, but instead of simply adding a new message, one must be changed or deleted.

Question 5

I've learned a considerable amount about the process of analysis and design throughout the semester. I believe one of the most critical things I have learned is the importance of planning. All of the diagrams that we were required to do are very useful when it comes to actually implementing the system. Seeing how all of the pieces connect makes starting your implementation considerably easier. I also learned that sometimes you need to stray away from your original plan. Often times something won't work quite the way you thought it would or you will find some way to implement a feature that you hadn't thought of before. During implementation, analysis is key in identifying what is working well and what needs to change in order for the system to work as intended.