CSCI 5448 Project - Part 5

Team Members:
Yihua Shi
Xueyan Wu

Team #: 29

Project Title:
A Craigslist Like Site

1. List the features that were implemented (table with ID and title).

User Requirements			
ID	**Description**	**Topic Area**	**Priority**
US-01	User can create account.	Authentication	High
US-02	User can view postings as guest.	User Operation	High
US-03	User can sign in his account.	Authentication	High
US-04	User can post postings or save unfinished posting as draft.	User Operation	High
US-05	User can view his postings history.	User Operation	High

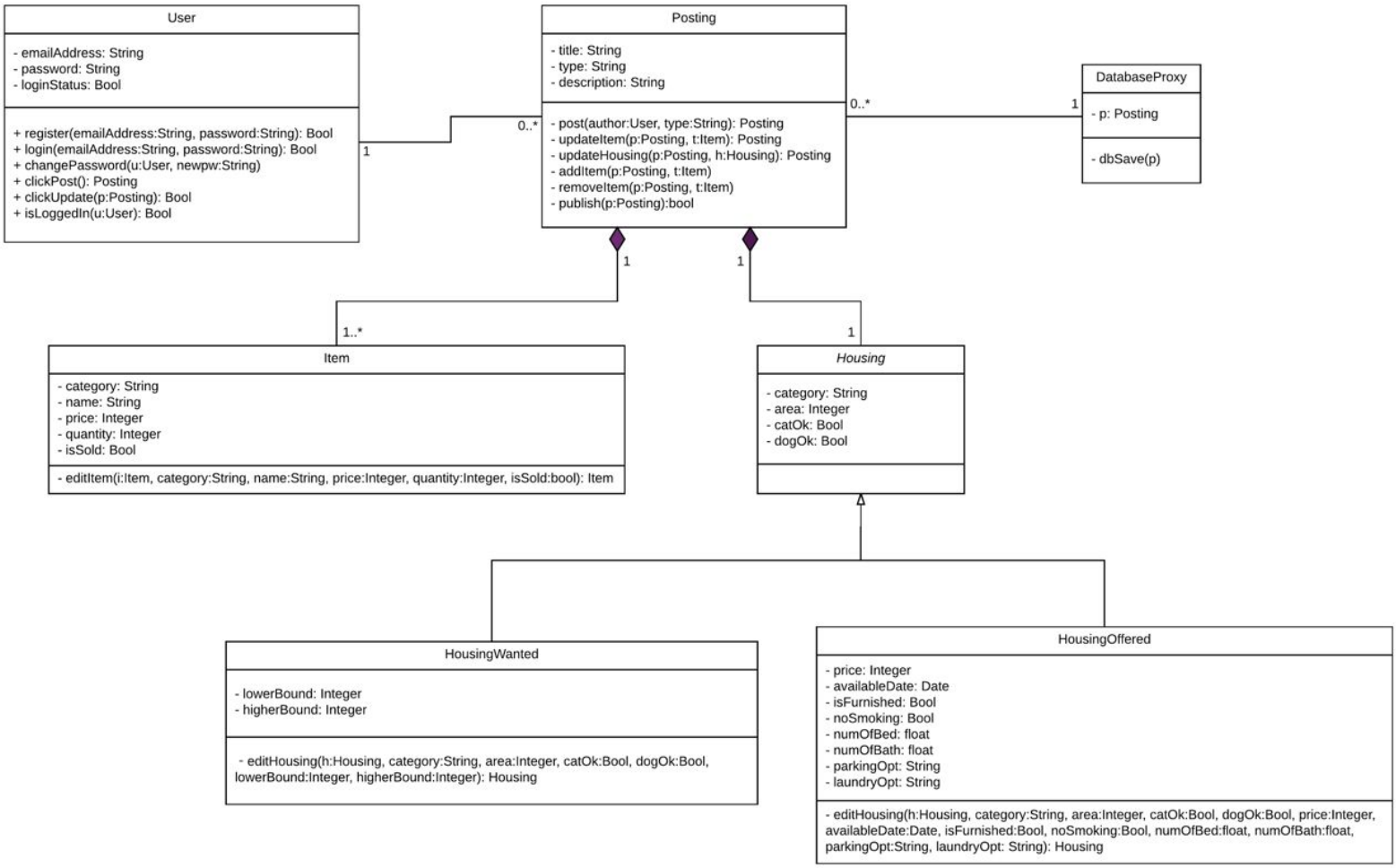
2. List the features were not implemented from Part 2 (table with ID and title).

User Requirements			
ID	**Description**	**Topic Area**	**Priority**
US-06	User can mark posting and add it to his favourite list while viewing postings.	User Operation	High
US-07	User can remove marked postings from his favourite list.	User Operation	High
US-08	User can update his postings.	User Operation	High
US-09	User can delete the postings that he posted.	User Operation	High

US-10	As a system admin, I want to see a dashboard of weekly number of postings so that I can monitor the usefulness of the system.	Web Statistics	Medium

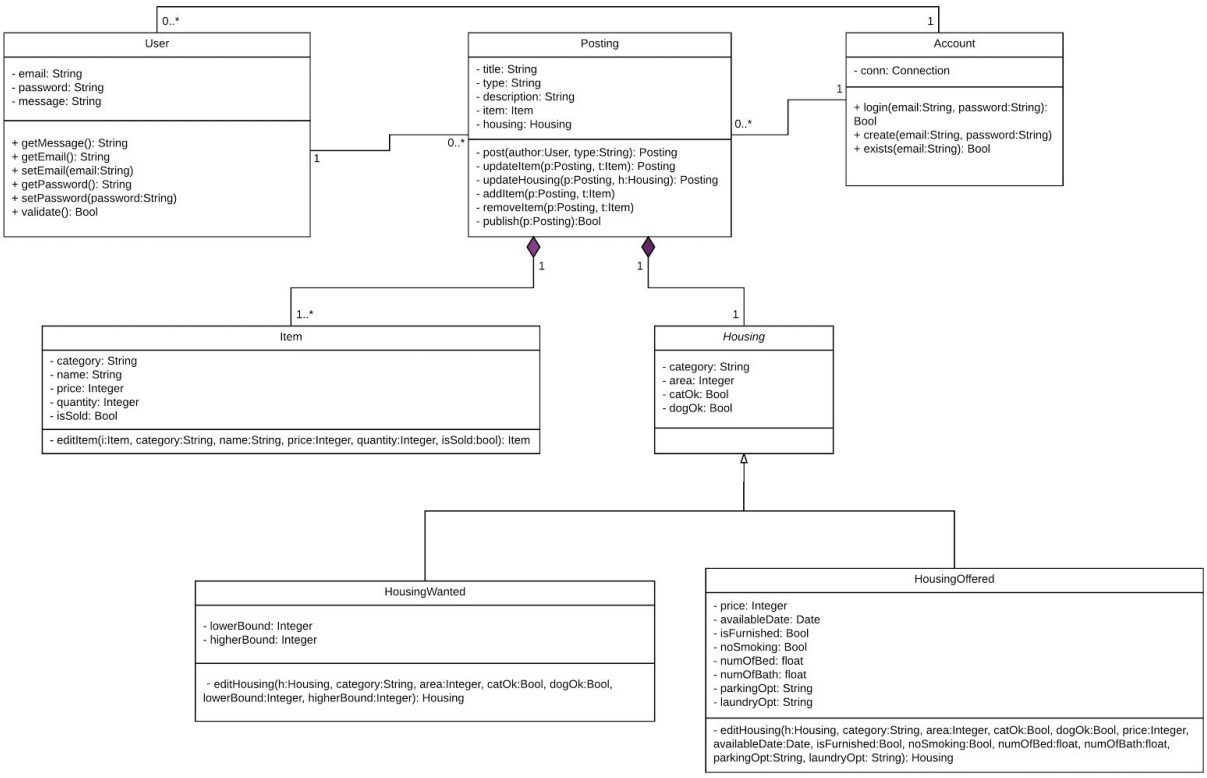
3. Show your Part 2 class diagram and your final class diagram.

1) Our original class diagram:



2) Modified class diagram

CLASS DIAGRAM



The original DatabaseProxy class was replaced by a class renamed as Account, which sets up the database connection and is ensured to have only one instance (singleton). We've also moved some of the operations from User class to Account since they appear more coupled with database manipulation. Besides, we've added the 1-to-many connection between Account and User class due to its usage of method calls.

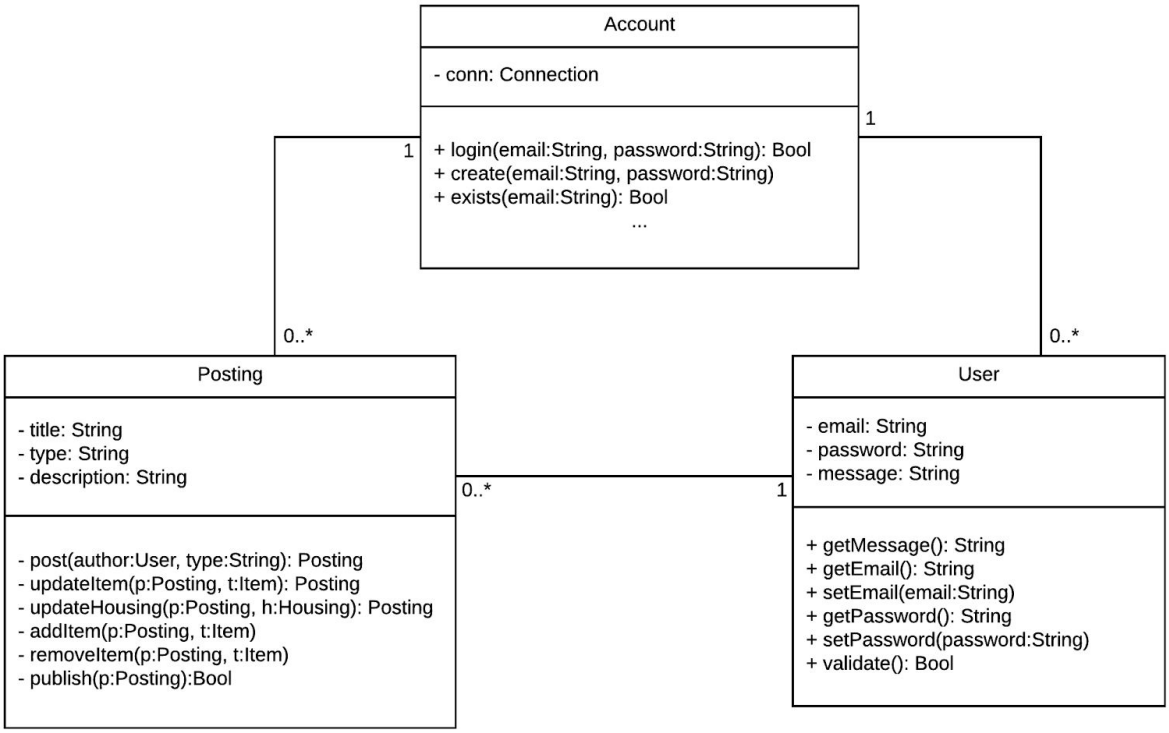
We've added more attributes like item and housing to the Posting class as we found they are also supposed to be included.

4. Did you make use of any design patterns in the implementation of your final prototype?

If so, how? Show the classes from your class diagram that implement each design pattern (each design pattern as a separate image in the .PDF).

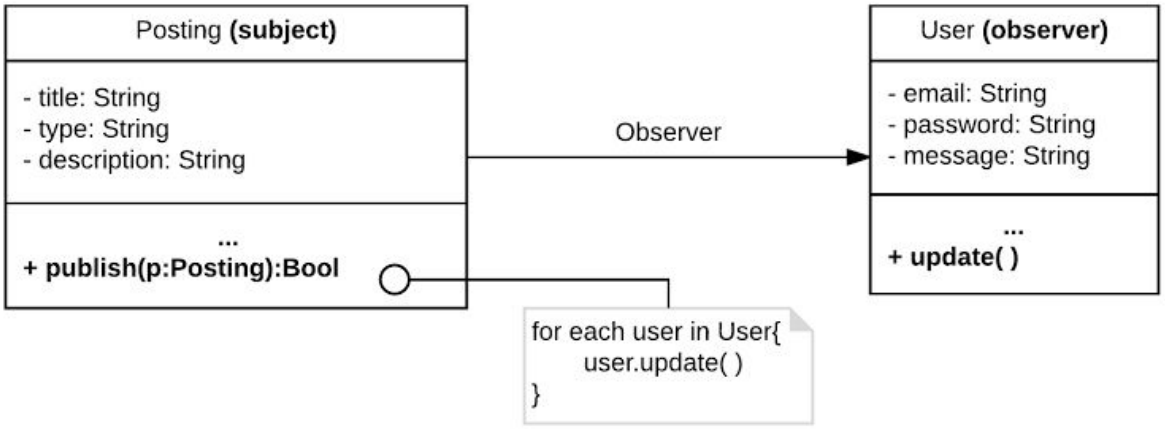
We used Singleton for this project, and could be further modified to use the Observer pattern.

The Singleton pattern was used in this project as it creates only one instance of Connection and thus ensure concurrency control to database.



If not, where could you make use of design patterns in your system? Show a class diagram of how you could implement each design pattern and compare how it would change from your current class diagram.

Our project has the publish mechanism for postings, where all the other online users would be able to get updates and see the latest posts. This could be adapted to the Observer design pattern and we were actually quite close. However, we didn't have the update() method in our observer class (which is the User class) even though we implemented the publish() method in our object class (which is the Posting class). Therefore, we could modify our class diagram to be the following so as to implement the Observer pattern.



5. What have you learned about the process of analysis and design now that you have stepped through the process to create, design and implement a system?

Before when we get a project, we always start coding without design and frame building. It's actually not a good habit for programming. Right now we have learnt several important things:

- Diagrams makes life really easier, especially for the partners and client. They can let people easily understand the goal and the activities of a system.
- Design patterns are useful and helpful for programmers, but we really have to practice a lot with it and understand the benefits that they have. Only reading material and studying the theory are not enough.
- The design of your system may be modified while implemented it.