

Project Writeup

Over the course of this semester I had to learn several different things in order to get to the point where I am now. Not everything I learned was major or felt significant, however, learning some of these more basic skills will have a major impact on my future in the computer field. Some of the skills I acquired were the use of GitHub, mapping data flow through the use of diagrams, and effective communication in working with a team.

Learning GitHub, while basic to most at this point, was a major step for me as I had never really thought of the benefits of using repositories because all of my programming courses up until this point were of small scale and did not benefit as much from GitHub. Everyone on my team who did not know how to use GitHub was given the rundown on how to use it for our specific repository by Dan and Will, this along with googling the commands was sufficient in figuring out how to use it. In this course it was significant because it allowed multiple team members to work on different areas of the code and get smaller pieces working separately in order to prepare them to be merged together. This is important because if someone made a mistake on a major piece of code they could simply revert back to the working version.

The second thing I would like to highlight is the mapping of data flow using diagrams such as the overarching UML as well as smaller details such as sequence diagrams. Both of these will be presented as examples at the end of this report. I was in charge of keeping our team's vision organized in a main UML. Each time we met as a group to discuss what we wanted to implement I would scribble down notes and organize them into the document we used as our roadmap. The overall UML was important because it would let the team know what we wanted

to do and the rough data flow to help determine what classes interacted with what. In addition to the main UML I also made several sequence diagrams, this was learned from the professor's examples as well as a sequence diagram generator website that had an excellent "instructions" page and gave tooltips for all the different icons and arrows used in a sequence diagram. This made learning more about the diagrams very easy.

Teamwork and communication are necessary when it comes to working in a team, each person should have roles that let them contribute effectively. Over the course of this project I was not in a coding role, I decided to take on the role of co-project manager with both Jonel and Michelle. For this task, we decided on using Trello because it was a clean and efficient way to task out everyone and keep track of what each person would be working on. With this job, we would find out who was comfortable with what parts of the project and everyone ended up taking a position that they enjoyed.

For use cases, I started off our group by making the "Create Account" use case. This use case was used as the template for everyone else's use cases so that we would have them uniform to make it easier to follow. I also did the use case for "Delete Account". I would also go on to do sequence diagrams of these in order to illustrate the data flow we needed in order to make them work. I had also done a little bit more research on design patterns after our lectures on them, using the website geeksforgeeks.org was a major help. In fact, that site is where much of my information came from for this class. As for what is next, it is just a matter of putting a presentation together and ultimately displaying what we worked on this semester.

Name: Create Account

Description A use case that allows a visiting user to create a login account for the whereterbottle app.

Goal Create an account with Whereterbottle

Preconditions (List the state(s) the system can be in before this use case starts)

1. Launched app for the first time
2. Signed out of app

Basic Course (Describe the "normal" processing path, aka, the Happy Path)

1. Use case begins when a user presses a "Create Account" button
2. System opens form for user to input information
3. User enters required information to register
4. System validates the entered Create Account information
5. The values for the created account are stored in the user's Account
6. Use case ends when the user has successfully created an account

B. Username already exists in system

Post conditions (List the state(s) the system can be in when this use case ends)

1. Success - The user entered data that is stored in the user's Account.
2. The account was not created - The user entered invalid information or aborted the account creation process

Actors (List of actors that participate in the use case)

User
System

Notes

Alternate Course A: Description of the alternate course

Condition: User aborts account creation

1. At any point during the Create Account process, the user can choose to cancel the account creation. At this point the user is notified that the account creation process has been terminated.
2. Use case ends when the user aborts the account creation process

Alternate Course B: Description of the alternate course

Condition: User enters invalid account information

1. Use case begins when a user presses a "Create Account" button
2. System opens form for user to input information
3. User enters required information to register
4. System could not validate the entered Create Account information
5. The user is informed of invalid field(s)
6. User re-enters account information
7. If system could not validate information the flow is repeated until valid information is entered
8. Use case ends when the user has successfully created an account or user aborts account creation process

Invalid Create Account information:

A. Missing information items

Name: Create Account

Description A use case that allows a user with a valid Whereterbottle account to delete their current account for the whereterbottle app.

Goal Delete an account with Whereterbottle

Preconditions (List the state(s) the system can be in before this use case starts)

1. Currently Signed into a valid Wherete bottle account

Basic Course (Describe the "normal" processing path, aka, the Happy Path)

1. Use case begins when a user presses "Delete Account" button while logged into a valid Wherete bottle account
2. System displays form requesting password
3. User enters valid Wherete bottle password for current account
4. System validates the password and display confirmation message
5. User confirms they want to delete their current Wherete bottle account
6. Use case ends when the user has successfully deleted their account

Post conditions (List the state(s) the system can be in when this use case ends)

1. Success - The user successfully deleted the current account.
2. The account was not deleted - The user entered invalid password or aborted the account deletion process

Actors (List of actors that participate in the use case)

User
System

Notes

Alternate Course A: Description of the alternate course

Condition: User enters invalid password

1. Use case begins when a user presses "Delete Account" button while logged into a valid Wherete bottle account
2. System displays form requesting password
3. User enters invalid Wherete bottle password for current account
4. System cannot validate the password and display error message
5. User must reenter a valid password otherwise repeat step 4
6. Use case ends when the user has successfully deleted their account or aborts action

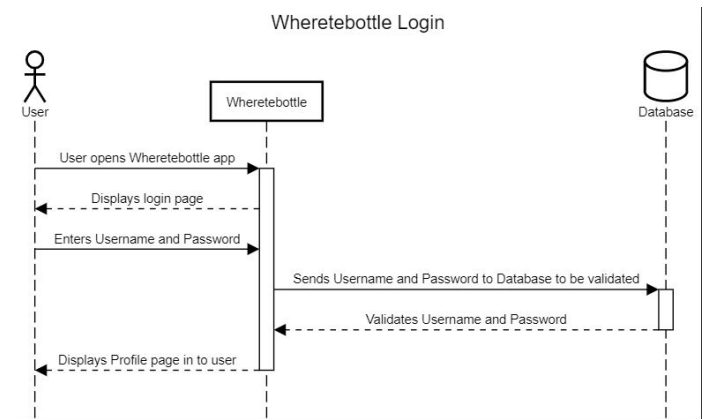
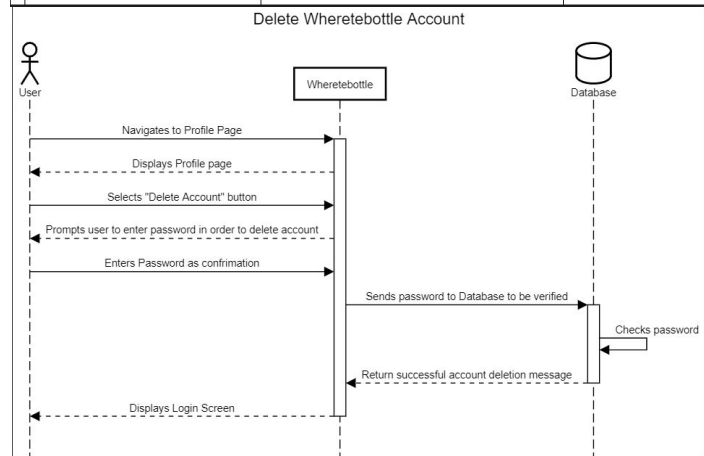
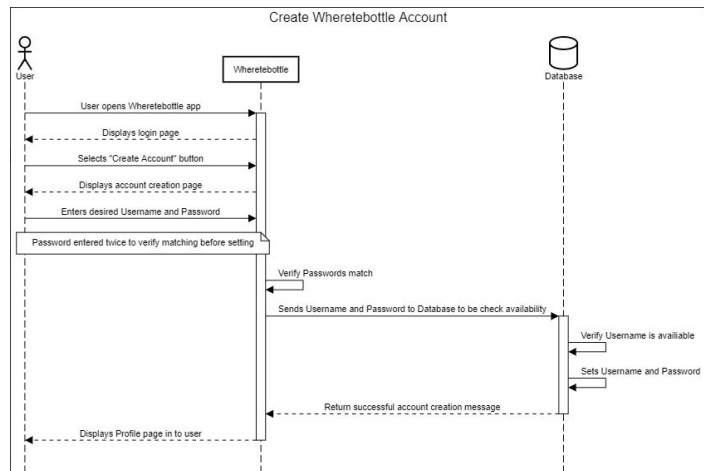
Alternate Course B: Description of the alternate course

Condition: User cancels delete request

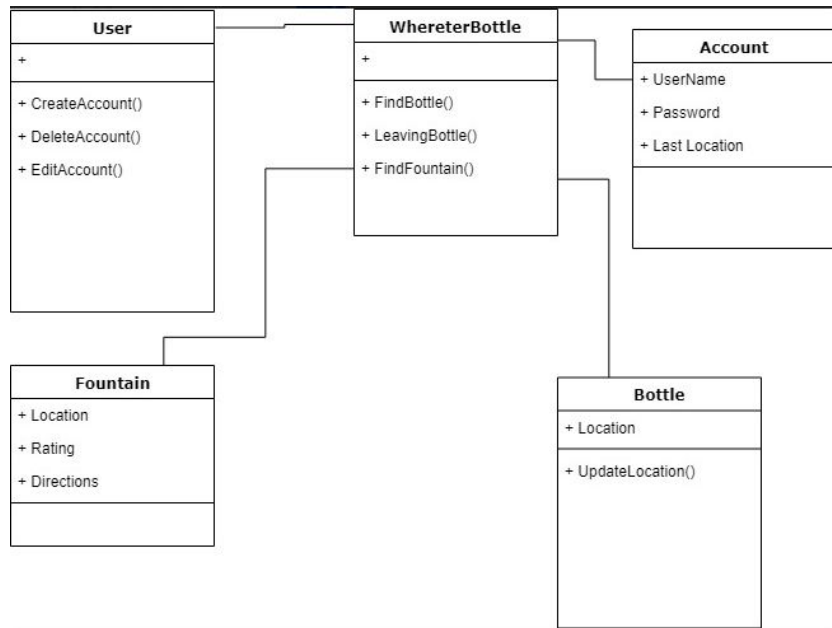
1. At any point during the Delete Account process, the user can choose to cancel the account deletion. At this point the user is notified that the account deletion process has been terminated
2. Use case ends when the user aborts account deletion process

Invalid Delete Account information:

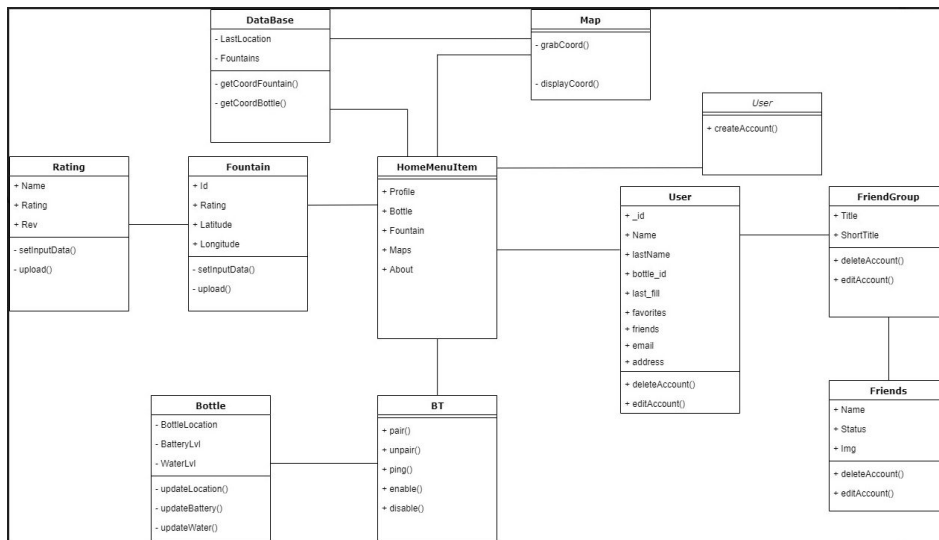
- A. Missing information items
- B. Password does not match account password in system
- C.



Early Design



End Design



Helpful Links

<https://www.geeksforgeeks.org/list-useful-github-commands/>

<https://www.geeksforgeeks.org/software-design-patterns/>

<https://sequencediagram.org/>