CSCI E-97 Assignment 4
David Killeffer <rayden7@gmail.com>

Mobile Application Store Authentication Service API Results Overview

Date: 2013-11-20
Author: David Killeffer <rayden7@gmail.com>
Reviewer(s): Jonathan Nichols <jn42887@gmail.com>

Comments from peer design review and optionally the functional review

Jon and I collaborated over email to share each others' design documents and offer feedback. The first revision of my design document got the following comments from Jon:

> *Looks great! I have a few comments, most of them very minor:*
>
> *1. For #2 of the Introduction, in the body, you said you would explain changes made to the CollectionServiceAPI to accommodate...etc. You should mention that there also changes necessary to the ProductAPI.*
>
> *2. In this section of the Requirements:*
>
>> *"The AuthenticationServiceAPI will allow for Users to log into the system, be granted Permissions and Roles, allow administrators to define those Roles and Permissions as well as Services, and allows for a robust way of ensuring that only allowed users carry out certain actions in the Mobile Application Store."*
>
> *I think there are some inconsistencies in the "will allow", "allow", and "allows". Also, perhaps use a different word for "allowed users" since you use the word "allow" so many times already.*
>
> *3. In the requirements, you could explain whether users can have different roles/permissions depending on the credentials that they use. This is an unclear part of the assignment, so it's something I explained a lot.*
>
> *4. In the restricted methods, you pass in the AccessToken object. I was a bit unclear on this, but based on some posts in the discussion forum, it sounds like you don't actually pass the AccessToken object. Instead, you pass around a hashed id that links to the AccessToken. I'm not positive on this, but it's something to be aware of.*
>
> *5. I like the idea of housing Permissions within Services. That makes a lot of sense, and it's something I didn't do.*
>
> *6. I'm not familiar with "salt," although perhaps that's a common term for more experienced programmers. If it isn't, perhaps explain what you mean by salt in the document.*
>
> *7. I like the Sequence and Activity Diagrams. They do a good job of explaining things.*
>
> *8. I may have missed it, but did you ever explicitly state how the design meets the requirements? I know that's something you reminded me to include, so perhaps you already have it in there somewhere.*
>
> *Otherwise, really good. I'm sure it'll be another quality project.*

-Jon

Changes made to my design and how they continue to support the requirements
Although the original requirements called for using a "factory method, where appropriate", I did not find that I needed to use a factory pattern to effectively create the authentication objects. Also, the nature of my design (having multiple tiered inheritance of abstract classes for the authentication items) made it challenging to have a pure "factory" method to create objects in a meaningful way that would be different from just using the class constructors. I also refactored how I was passing AccessTokens; after also reading the various posts on the discussion forums and getting Jon's feedback, it became clear that I should only be passing the tokenID rather than the entire AccessToken object, so I refactored that.

Did the design document make the implementation easier?

The design document did help to make the implementation somewhat easier, though I still did run into some interesting challenges in certain areas, like password hashing (for which I used a utility class to help with that). I leveraged the Iterator pattern that we used in Assignment 3 to create the RoleIterator for this Assignment. The sequence diagram did definitely help to get a better understanding of the order of operations and made clearer what properties and attributes objects needed to communicate with each other. Over the course of the implementation it became clear that I needed some private helper methods to decompose some methods to be simpler in the AuthenticationServiceAPI.

How could the design have been better, clearer, or made the implementation easier?

I think my design document is pretty clear about the intended architecture.

Did the design review help improve your design?

Jon's feedback was definitely constructive, and I did try to go back and make the design document a bit clearer according to his recommendations. The design review process is good, not only for getting direct feedback, but also for being able to see what other people's designs and ideas are when interpreting the exact same set of requirements.