Senseye - Take Home iOS Coding Assignment

Assignment

For this assignment, you’re going to build a simple iOS application that contains a *feature flagging system* that returns a customized nickname for a given user!

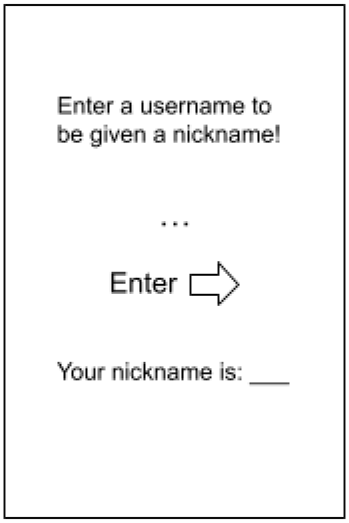
Feature flags are a critical component of customizing our application’s behavior for different clients. If you are not familiar with feature flagging systems, they are a method that allows you to update or modify certain parts of your app without releasing a new version update. For example, a boolean feature flag can be used to toggle a button on/off for specific users of the app. Typically, feature flagging systems talk to an external API or server to configure apps on the fly, but for simplicity, all the code will be contained in one project here.

UI

The app contains a single screen with a simple text entry field, an enter button, and a label underneath.

Once the user enters a username into the field and selects the enter button, a text field with the customized nickname will appear underneath.

Below is an example of the UI, but feel free to make any modifications or updates you feel are good for UX!



Enter a username to
be given a nickname!

...

Enter ⇨

Your nickname is: ___

Feature Flag Configuration

Create a separate class to handle our internal “Feature Flagging” system. The main goal of this class is to return a simple string nickname based on the inputted unique username.

This class should be initialized with the submitted username, and maintain a private variable nickname that has a custom getter function to return our customized nickname.

For example, if we define the below mapping of usernames to nicknames, we would expect the nickname to be returned for the given username.

Username	**Nickname**
wsmith1	Fresh Prince
mAndM	Slim Shady
jdoe21	Jam Handsome

For this task, nicknames are arbitrarily assigned by the app and need not have a meaningful relation to the username.

Testing

Run your project in Xcode, and verify the app displays the correct nickname for each inputted username!

Unit tests are not required, but write about how you would build them to verify correct behavior.

Tip: What classes would unit tests be particularly useful for? How would you ensure essential functions are working appropriately?

Submission

Please send us a link to your Github for the project, or a zipped folder of the uncompiled code! Attach a few screenshots and a small summary of your solution as well.

Tip: Think of this summary as a Pull Request description - what things would be useful for a reviewer to understand about your solution? In general, we want to learn how you approach this exercise. Particularly in how you lay out the ui/ux elements, handle edge cases, and the quality of your code.

Extra Credit

Develop a second feature flag that can be used to return a fully capitalized nickname. When this is disabled, return the default configured nickname.

Below are two examples of how the feature flags would be used together to return a final nickname:

1) Username for which shouldCapitalizeNickname is enabled:
   username -> Jdoe21
   nickname (for given username) -> Jam Handsome
   shouldCapitalizeNickname (for given username) -> true

   Outputted nickname -> JAM HANDSOME

2) Username for which shouldCapitalizeNickname is disabled:
   username -> mAndM
   nickname (for given username) -> Slim Shady
   shouldCapitalizeNickname (for given username) -> false

   Outputted nickname -> Slim Shady

PS:
This area is not required, the clarity and code quality in the previous section are more than enough to work with. If you have extra time and are excited, this is a way you can extend the exercise!