CS 329E: Bulko
Programming Assignment 8:
Simple Animation

1 Problem Definition

Your goal in this assignment is to play with simple animation. There is only one View Controller in this project, as shown in Fig 1(a). In the View Controller, there is one image, “UT Tower”, that occupies the full screen. When you click on the “UT Tower” image, it will change to the image “UT” with animation, as shown in Fig 1(b). Then if you click on the “UT” image, it will change back to “UT Tower” with animation, and so on.

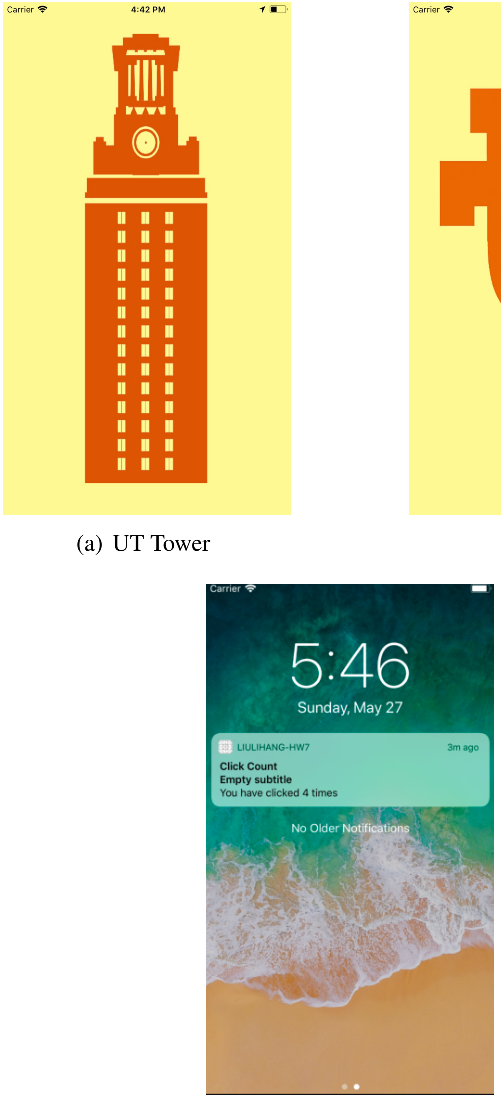
The two images you should use can be found in the directory as our class webpage. Simply go to our webpage, and replace “329E.html” with the following:

- UT image: “ut.png”
- UT tower image: “uttower.png”

You are also required to implement a local notification component. When you have clicked the images 4 times, a local notification with time interval of 8 seconds will be triggered as shown in Fig 1(c). Similar notifications saying ”8 times”, ”12 times”, etc. should be triggered after the appropriate number of clicks, all multiples of 4.

2 Detailed Instructions

- Create a Single View application project named <lastName><firstName>-HW8.
- Storyboard:
  - Set the background color of the main view of the ViewController to some color (choose one you like except white).
  - Add a button to the storyboard with appropriate layout constraints so that it is centered and occupies the full screen.
- View Controller: When the button is clicked, if the current image is image A, change it to image B with the following animations:
  - Step 1: Animation 1 with
    - ∗ duration: 1.0
    - ∗ delay: 0.0
    - ∗ options: `curveEaseOut`
    - ∗ animation handler: set the alpha of the button to 0.0



Click Count
Empty subtitle
You have clicked 4 times

(a) UT Tower

(b) UT

(c) iPhone Lock/Home Screen

Figure 1: Application demos

    - ∗ completion handler: Step 2 and 3.
  - Step 2: Set the image of the button to image B.
  - Step 3: Animation 2 with
    - ∗ duration: 1.0
    - ∗ delay: 0.0
    - ∗ options: `curveEaseIn`
    - ∗ animation handler: set the alpha of the button to 1.0
    - ∗ completion handler: `nil`

- Note that Step 2 and 3 should be embedded in the completion handler of Animation 1, so that the 3 steps are executed in order.

- When you click on the image again, it should switch back to image A. Subsequent clicks should cause the image to alternate between A and B.

- When you have clicked the images 4 times, a local notification will be triggered with the following properties:

  - time interval: 8s
  - title: `Click Count`
  - subtitle: make up something appropriate
  - body: `You have clicked 4 times`

  Immediately exit to the home screen. (In the simulator, you can quickly do this by typing Command-Shift-H.) You *must* do this: by default, push notifications from an app do not display while the app is in the foreground!

- After 8 seconds, the notification will show up.

3 Grading criteria

1. The background of ViewController is set to some non-white color. (10%)

2. Appropriate constraints are added to the button. (20%)

3. When the button is clicked, the image of the button changes from one image to another. (20%)

4. The animations are correctly defined and behave as expected. (20%)

5. The local notification works as expected. (30%)

6. **Note that if the app does not build and run, ZERO points will be given.**

7. The Coding Standard is followed. One point deducted for each violation.

4 General criteria

1. I will be looking for good documentation, descriptive variable names, clean logical structure, and adherence to all coding conventions expected of an experienced programmer, as well as those outlined in the Coding Standard document. There will be penalties for failure to meet these standards.

2. Your code must compile and run before submission.

3. Xcode will automatically generate standard headers to your .swift files. Add two lines to each Swift file so that the header includes the following:

   ```
   // Project: LastnameFirstname-HW8
   // EID: xxxxxx
   // Course: CS329E
   ```