CP469- iPhone Application Programming

Winter 2015

Assignment 1 (Category 1)

Due: Wednesday Januart 21, 11:30PM

If you do the assignments on the lab's Macs, you should save your work on your own USB stick. Make sure you do not leave any work on the Macs

You will build a iPhone app that provides a number of questions (about yourself) and the corresponding answers, together with appropriate images. A video demo of the app can be viewed at Youtube. You app's behaviour should match that of the demo. Your app should be named using the class convention (read the Submission part at the end of this document.)

Steps:

Add the following user interface elements to the storyboard:

One UIImageView object

Two labels (one for questions, one for answers)

Two buttons (for showing questions and answers)

From the Button Fun project (Chapter 3 of the text), you already know how to add labels and buttons to the View.

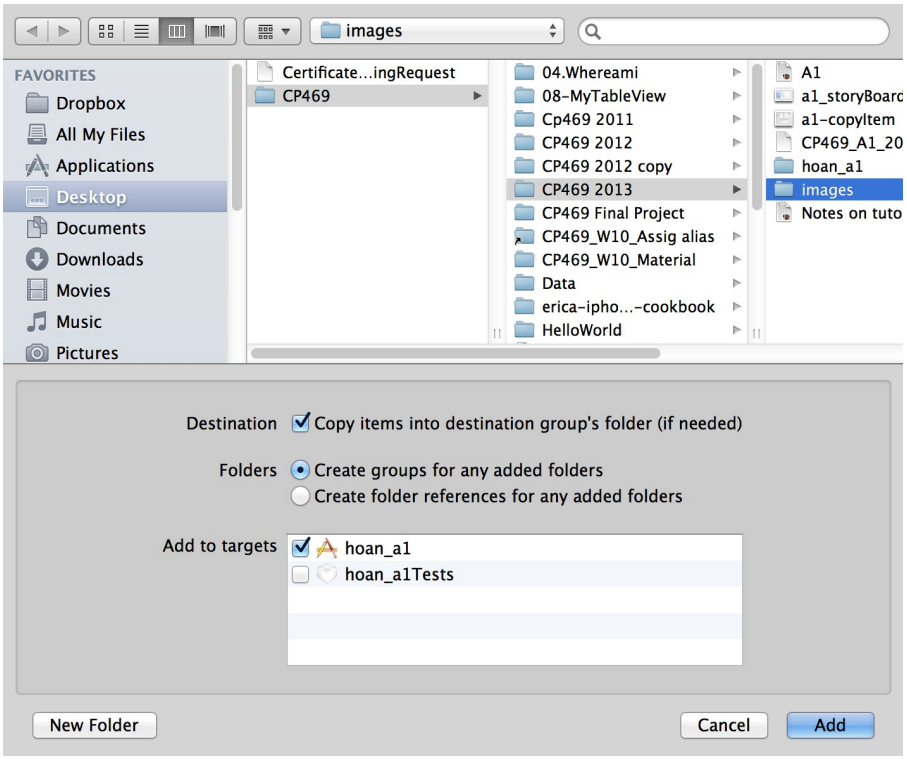
Now, let's talk about adding images to the project. The simplest ways is to put all images in a folder in the Mac. Let's call this folder "images".

In the navigator area, highlight the folder "supporting files"

In the File menu, choose "add files to..."

Navigate to the folder "images"

Make sure that there is a check on the check box "Copy items into destination folders..." If you don't do this, the images won't be inside of your project directory and when you submit your assignment the resources will not be included in the submission. See figure below



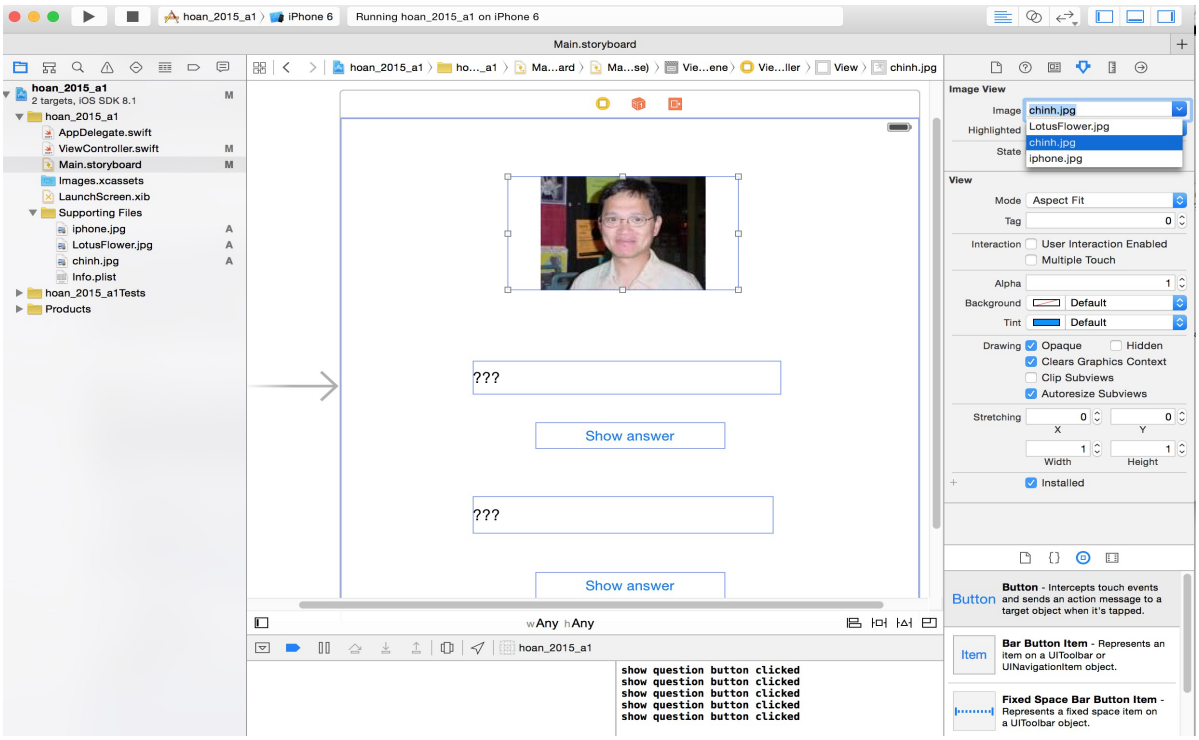
Double-click on the folder "images". The folder and all images inside it are added under the "supporting files" folder. Your navigator area should look like below

Note: you may also add images one by one by clicking on each of them, and you may also put your images in a different folder.

Now, let's add an UI ImageView. Choose Library --> Objects --> Data View --> Image View

Drag the Image View object to the storyboard. You can resize the UIImageView object for your liking.

Now, you need to associate the Image View object to one of your own images, say, your portrait photo. In the storyboard, select the image view and set the image for the image view using the Attributes Inspector (click on the Image box, choose the image you want, see picture below).



Configure the image view's "Mode" to be "Aspect Fit" so that the entire image is scaled to fit inside the image view's bounds while preserving the aspect ratio of the image (click on the Mode box). If a different mode works better for your picture, set it accordingly. Explore the Mode box to see all different modes you may have.

Now, you may build the rest of the UI elements and write codes to complete the project. Your project should run without errors and warnings.

Connecting the UI elements to the codes

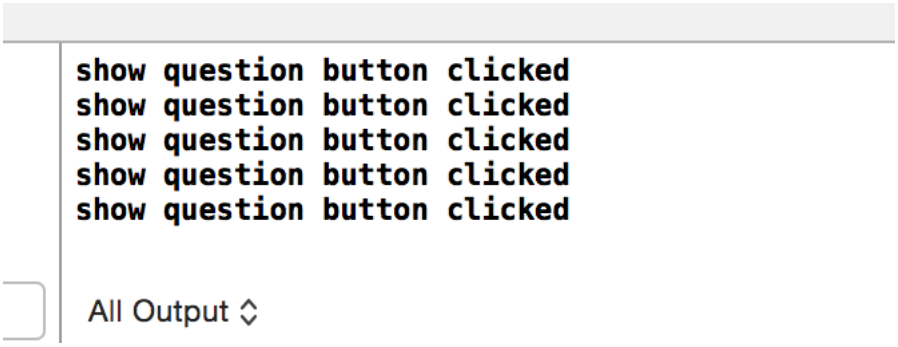
Create outlets for the Labels and Image View with the Assistant Editor. Create the actions for the two buttons also.

Test your target-actions

Write codes in the implementation file to log the method showQuestion, like below.

```
- (IBAction)showQuestion:(id)sender {
    println("show question button clicked");
}
```

Now, run the app, click on the Show Question button in the simulator, you should see a log in the debug area.



Where to place your initialization codes

The delegate method *viewDidLoad* is executed by the app after the view is loaded (logical, no?). You can do your initialization there.

```
    override func viewDidLoad() {
    super.viewDidLoad()
    // initialize your data here
}
```

Other requirements:
1. Images. Make your app display different images on different questions. The images should have some relations to the questions.

You will have to set the UIImageView object programmatically. This is done first by initializing an UIImage object to some existing image file (in jpg or some other format) and then set that UIImage object in UIImageView. For example:

```
var currentImage = UIImage(named: “myPhoto.jpg”) // set an UIImage
 imageView.image = currentImage // set UIImageView to UIImage
```

where imageView is the UIImageView object (the outlet of your UIImageView object in

the storyboard)
2. When the showQuestion button is pressed, the answerField should be empty or contains “???”

3. Test your app with different Iphone screen size. Make sure the UI elements are nicely placed in all sizes.

4. Your app should run without warnings and errors.

The submission. Your app should be named as required by the course outline, that is, if your name is hoang, then the app should be called hoan_a1. Highlight the app folder (it should have a name such as hoan_a1). Compress it by selecting it in the Finder and choose File->Compress. Alternately, you may right-click on the folder and choose Compress. Submit this zip file at my learningspace.

Recommended reading. In XCode’s developer documentation, read the iPhone development guide. This is a large document! Add it to your bookmarks.