

Project 3

Students' Choice of Native Apps

each milestone's deadline is noon

see cs164.net/expectations for each milestone's expectations

| Mon | Tue | Wed | Thu | Fri |
|-----|---------------------------------|-----|----------------|-----|
| | 4/10 Proposal | | | |
| | 4/17 Design Doc, Style Guide | | | |
| | 4/24 Alpha | | | |
| | | | 5/3 Release | |

Help

Help is available throughout the week at <http://help.cs164.net/>,
and we'll do our best to respond within 24 hours. But do turn first to your partner with bugs!

Academic Honesty

All work that you do toward fulfillment of this course's expectations must be the work of you and your partner. Collaboration with anyone other than the partner with whom you begin the semester is not permitted unless one of the course's heads approves a change of partner in writing. Partners must contribute equitably to each milestone: you may not implement most or all of some project's milestone and submit it on behalf of your two-person team.

Viewing or copying another individual's work (even if left by a printer, stored in an executable directory, or otherwise exposed) or lifting material from a book, website, or other source—even in part—and presenting it as your own constitutes academic dishonesty, as does showing or giving your work, even in part, to another student or soliciting the work of another individual. Similarly is dual submission academic dishonesty: you may not submit the same or similar work to this course that you have submitted or will submit to another. Nor may you provide or make available solutions to projects to individuals who take or may take this course in the future. Moreover, submission of any work that you intend to use outside of the course (*e.g.*, for a job) must be approved by the staff.

You may read and comment upon classmates' code toward fulfillment of projects' code reviews but only for classmates whose code is assigned to you by the course's staff for review. You may integrate ideas and techniques that you glean from your reviews of classmates' code and from classmates' reviews of your code into your own work, so long as you attribute those ideas and techniques back to your classmates, as with comments in your own code. As for classmates beyond your own partner and those with whom you're involved in reviews, you may discuss projects, including designs, but you may not share code. In other words, you may communicate with those classmates in English, but you may not communicate in PHP, JavaScript, or Objective-C. If in doubt as to the appropriateness of some discussion, contact the course's heads.

You may turn to the Web for instruction beyond the course's lectures and labs, for references, and for solutions to technical difficulties, but not for outright solutions to projects or portions thereof. However, failure to cite (as with comments) the origin of any code or technique that you do discover outside of the course's lectures and labs (even while respecting these constraints) and then integrate into your own work may be considered academic dishonesty.

All forms of academic dishonesty are dealt with harshly. If the course refers some matter to the Administrative Board and the outcome for some student is *Admonish*, *Probation*, *Requirement to Withdraw*, or *Recommendation to Dismiss*, the course reserves the right to impose local sanctions on top of that outcome for that student that may include, but not be limited to, a failing grade for work submitted or for the course itself.

Bitbucket.

- Okay, for the sake of discussion, we again need to call you or your partner Alice and the other of you Bob. Assume the same identities that you assumed on for Project 3, then read on!**
- Only Alice should perform this step.**

Log into your Bitbucket account and create a new, private repo as follows:

- Select **Repositories > create repository** at top-right.
- Ensure that **Create new repository** is highlighted (in dark blue).
- Input a value of **project3** under **Name**.
- Ensure that **Private** is checked.
- Ensure that **Git** is selected under **Repository type**.
- Check both **Issue tracking** and **Wiki** under **Project management**.
- Select **Objective-C** under **Language**.
- Input a value for **Description** and/or **Website** if you'd like.
- Click **Create repository**.

You should then find yourself at a page whose URL is <https://bitbucket.org/alice/project3>, where *alice* is your actual Bitbucket username. Click the **Admin** tab at top, then click **Access management** at left. In the text field under **Users (1)**, input your partner's Bitbucket username, then click **Admin** at right. Next, input **cs164** into that same text field, then click **Admin** at right. Finally, input your TF's username (which can be found at <https://www.cs164.net/Staff>), then click **Admin** at right. Your partner and CS164's staff, including your TF, should now have access to your repository. Your partner can confirm as much by visiting <https://bitbucket.org/alice/project3>, where *alice* is your actual Bitbucket username.

- Only Alice should perform this step too.**

Launch Xcode and select **File > New > Project...** When prompted to choose a template, select **Application** under **iOS** (if not selected already), then select your preferred template. (If unsure which to choose, **Single View Application** is probably a nice starting point, even if you intend to have multiple views.) Then click **Next**.

On the screen that appears: input **project3** for **Product Name**; input **edu.harvard.college** for **Company Identifier**; leave **Class Prefix** blank (it's okay if you see a placeholder of **XYZ** in gray); select **iPhone** next to **Device Family** (unless you'd prefer to select **iPad** or **Universal**); leave **Use Storyboards** unchecked (unless you'd like to use them); leave **Use Core Data** unchecked (unless you'd like to use it); check **Use Automatic Reference Counting**; check **Include Unit Tests**; then click **Next**. Choose a location for the project when prompted, check the box to create a local git repository, then click **Create**.

Select **File > Source Control > Repositories...**, and Organizer should open. Select **Remotes** under **project3** in Organizer's lefthand menu, then click **Add Remote** toward the window's bottom. On

the screen that appears, input **origin** for **Remote Name**, and input **https://alice@bitbucket.org/alice/project3.git** for **Location**, where **alice** is your actual Bitbucket username, then click **Create**. A folder called **origin** should then appear in Organizer. With **origin** selected, input your Bitbucket password toward the window's bottom. Then close Organizer's window.

Return to Xcode itself and, with **project3.xcodeproj** in the foreground again, select **File > Source Code > Push...** To the right of **Remote** should be **origin/master**, and a green light should indicate that the repository is online. (If not, best to retry all these steps.) Click **Push**. If informed that your push was successful, visit

`https://bitbucket.org/alice/project3/src`

with a browser, where `alice` is your actual Bitbucket username. You should see your code!

As for Project 2, anytime you'd like to commit some change(s) to your local repository (on your own hard drive), select **File > Source Control > Commit...** and follow the prompts. Anytime you'd like to push some change(s) to Bitbucket (perhaps for your partner's sake), select **File > Source Control > Push...** as you just did. And anytime you'd like to pull some change(s) from Bitbucket (as from your partner), select **File > Source Control > Pull...** Alternatively, you can use a command line by opening **Applications > Utilities > Terminal**, navigating your way to your `project3/` directory (as via `cd`), and using `git` as usual.¹

Okay, now Bob should perform this step.

Launch Terminal, as via **Applications > Utilities > Terminal**. Navigate your way to wherever you'd like to store Project 3's code (e.g., `~/Documents/`), as via `cd`. Then execute

```
git clone https://bob@bitbucket.org/alice/project3.git
```

where `bob` is your actual Bitbucket username and `alice` is Alice's Bitbucket username, inputting your Bitbucket password if prompted. (In theory, you can clone repositories in Xcode itself, but cloning appears to be buggy in Xcode 4.3.2, at least for HTTPS.)

Once done cloning, launch Xcode, as via **Applications > Xcode**. If prompted to choose a template for a new project, click **Cancel**. Then select **File > Open...**, and navigate your way to your **project3** directory. Select **project3.xcodeproj**, then click **Open**. You should see the codebase that Alice pushed to Bitbucket!

Henceforth, anytime you'd like to commit some change(s) to your local repository (on your own hard drive), select **File > Source Control > Commit...** and follow the prompts. Anytime you'd like to push some change(s) to Bitbucket (perhaps for your partner's sake), select

¹ Recall that you created an SSH key pair for Bitbucket inside of the CS50 Appliance for Project 0. If you'd like to use `git` at the command line, you'll want to copy that key pair (e.g., `~/.ssh/id_rsa` and `~/.ssh/id_rsa.pub`) from the appliance to your Mac's `~/.ssh/` directory, or you'll want to create a new key pair on your Mac (as with `ssh-keygen`) and then add the newly created public key to your Bitbucket account.

File > Source Control > Push... as you just did. And anytime you'd like to pull some change(s) from Bitbucket (as from your partner), select **File > Source Control > Pull...** Alternatively, you can use a command line by opening **Applications > Utilities > Terminal**, navigating your way to your **project3** directory (as via `cd`), and using `git` as usual.²

Specification.

- Your challenge for this project is to implement your choice of native iOS apps. So long as your project draws upon this course's lessons, the nature of your project is entirely up to you, albeit subject to the staff's approval. We only require that your app meet some technical requirements.

Technical Requirements.

- Your app must be implemented with iOS SDK 5.1 (and Xcode 4.3.2).³
- Your app's UI should be sized for an iPhone or iPod touch (*i.e.*, 320 × 480 points). However, if you or your partner owns an iPad and would prefer to optimize your app for it (*i.e.*, 768 × 1024 points), you may, so long as you inform your TF prior to this project's deadline.
- Your app must use Automatic Reference Counting (ARC).
- You must implement unit tests for your UI (*i.e.*, application tests) and any models (*i.e.*, logic tests).
- You must use `git` and Bitbucket for version control.
- Your app must work within the iPhone (or iPad) 5.1 Simulator; you need not test it on actual hardware. However, if you or your partner owns an iPad, iPhone, or iPod touch, and you'd like to install your app on it, see <https://manual.cs50.net/iOS> for instructions.

² *Ibid.*

³ Confer with your TF before your alpha's submission if you must, for technical reasons, use an earlier SDK.