

(and back up) text-based design assets like CSS, HTML, and JavaScript, and it can also help track graphic and binary files as well. Once you get started, you'll find a lot of reasons to keep GitHub looking over your project assets.

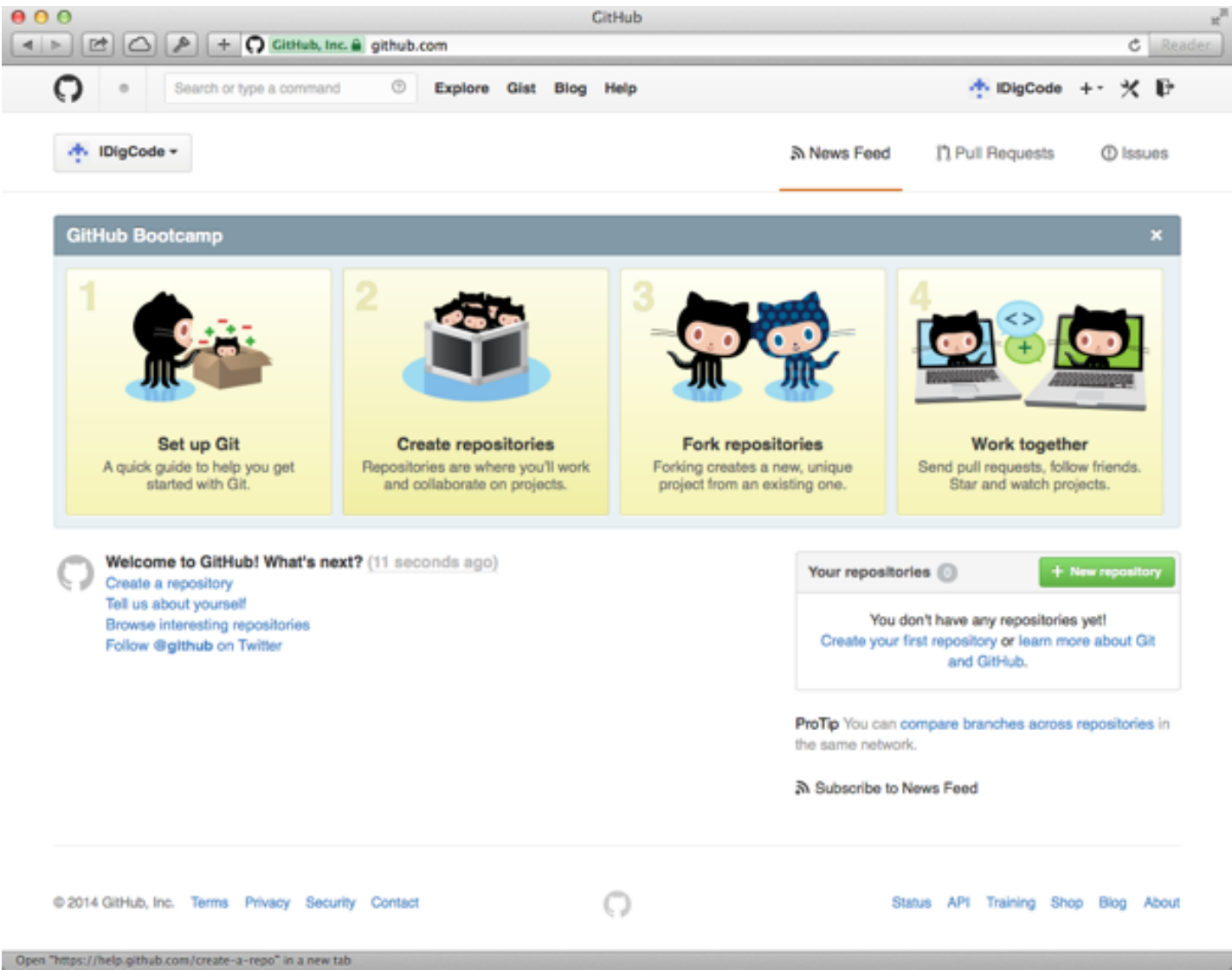
What is GitHub?

GitHub is basically a social network built around a hosted instance of Git, a popular version-control system created by computing legend Linus Torvalds. Linus has a history of tackling big problems; he also created the Linux operating system. Git is at its core a command-line application, but it also has handy and easy-to-use desktop apps that give you access to the core features of Git without having to go command line-native.

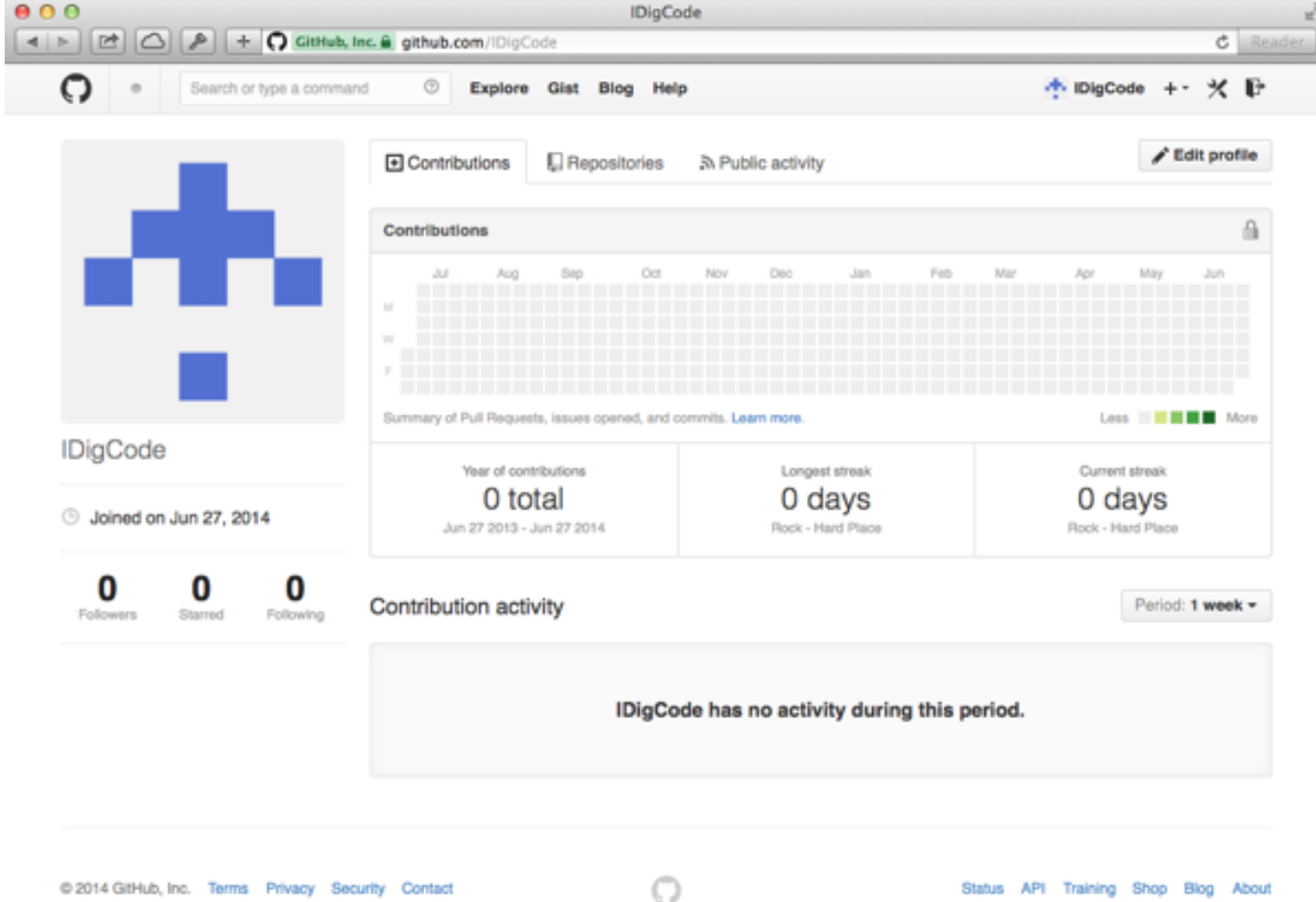
It's dead simple to use the GitHub desktop app to commit "snapshots" of your files to a cloud-based repository on GitHub as you work, and roll back to prior versions when you need to. Ready to go? Let's get started.

Create a GitHub account

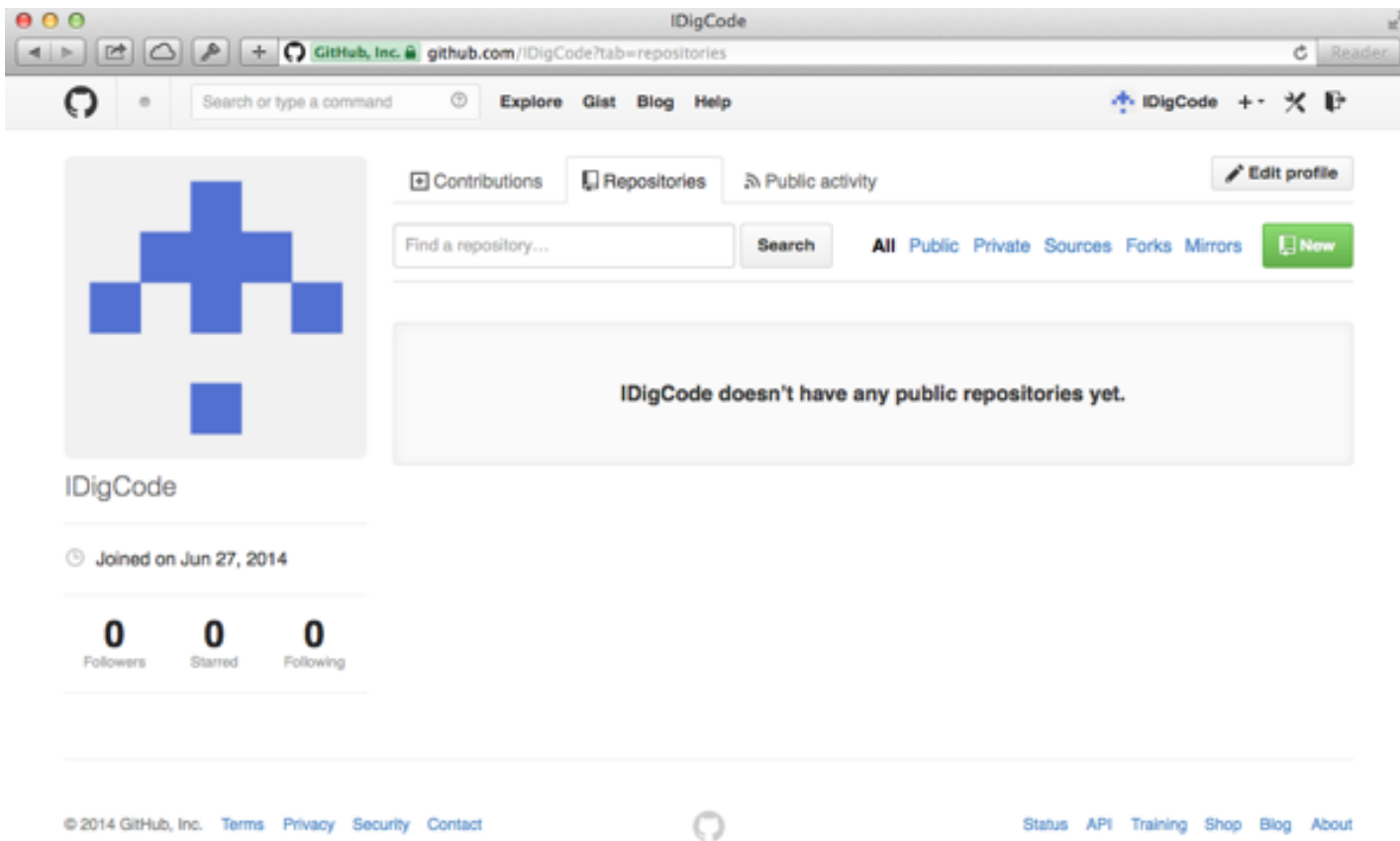
This part's easy. Just visit github.com (<http://github.com>), and create a new account. We'll use the free GitHub account. The key limitation with a free GitHub account is that your repositories are public, which means they can be found and followed by other GitHub members. No worries, though—if you need private repositories to keep your projects hush hush, GitHub offers paid plans allowing private repositories and other upgraded features. Make sure to note your email address and password while creating your account, as you'll need them when we get back to the desktop. After you've created your free account, you'll see the GitHub welcome page.



This is your new GitHub home page, so make a bookmark! You're currently in the News Feed view, which also shows the GitHub Bootcamp strip and many helpful links for learning more about Git and GitHub, in case you want to dig a bit deeper. To see your GitHub profile, click the name in the upper right of the browser window (in this case, the fictitious user IDigCode). You'll now have a slightly different view.



The current *Contributions* view gives an overview of your activity, and how you’re working across your repositories. To edit your profile and personalize it, you can use the **Edit profile** button in the upper right. Your repositories (i.e. the collection of files you’ll be managing via GitHub) can be found by clicking the *Repositories* tab, which shows that we don’t have any repositories just yet, either.

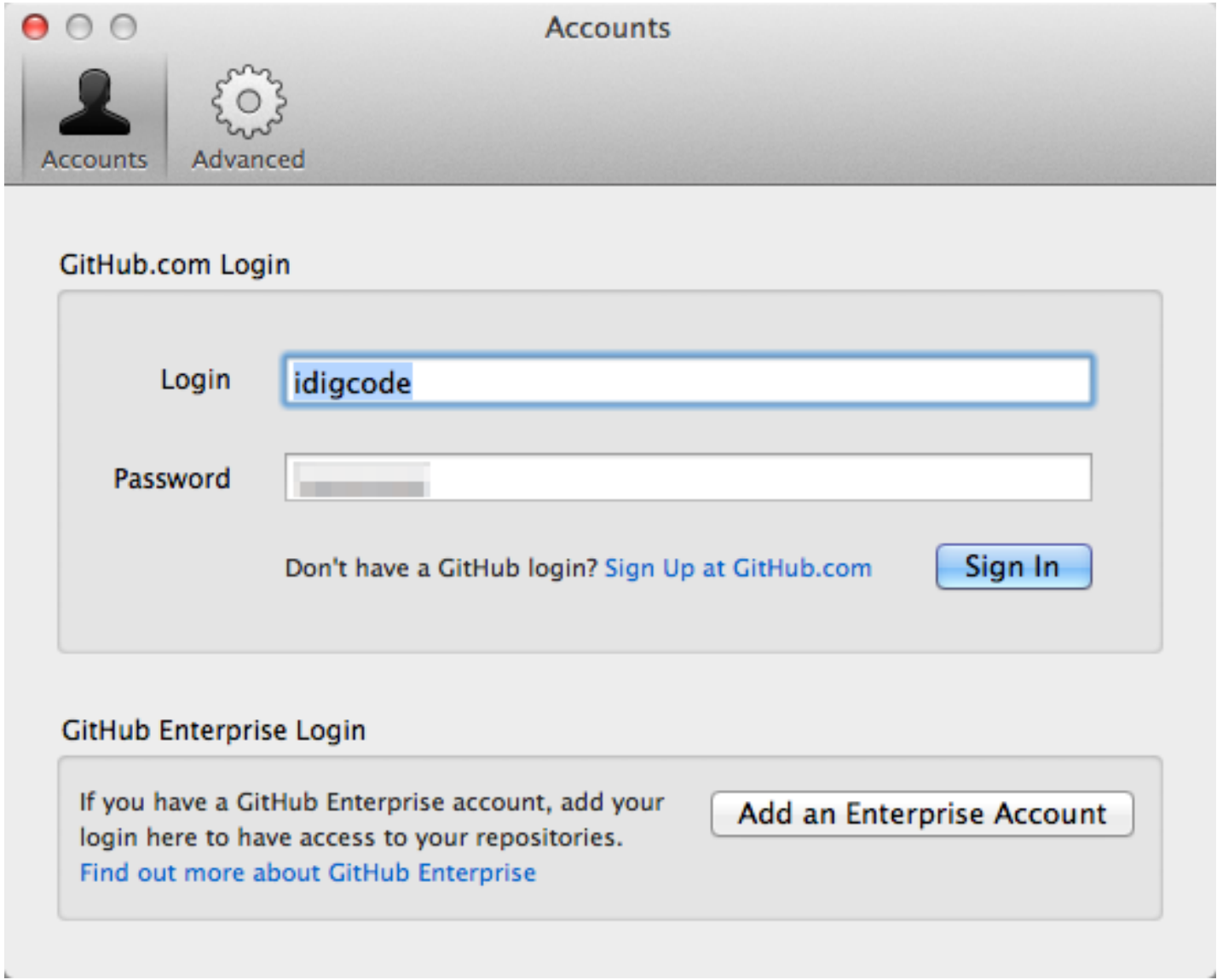


Let’s create a new repository on our desktop we can manage with GitHub.

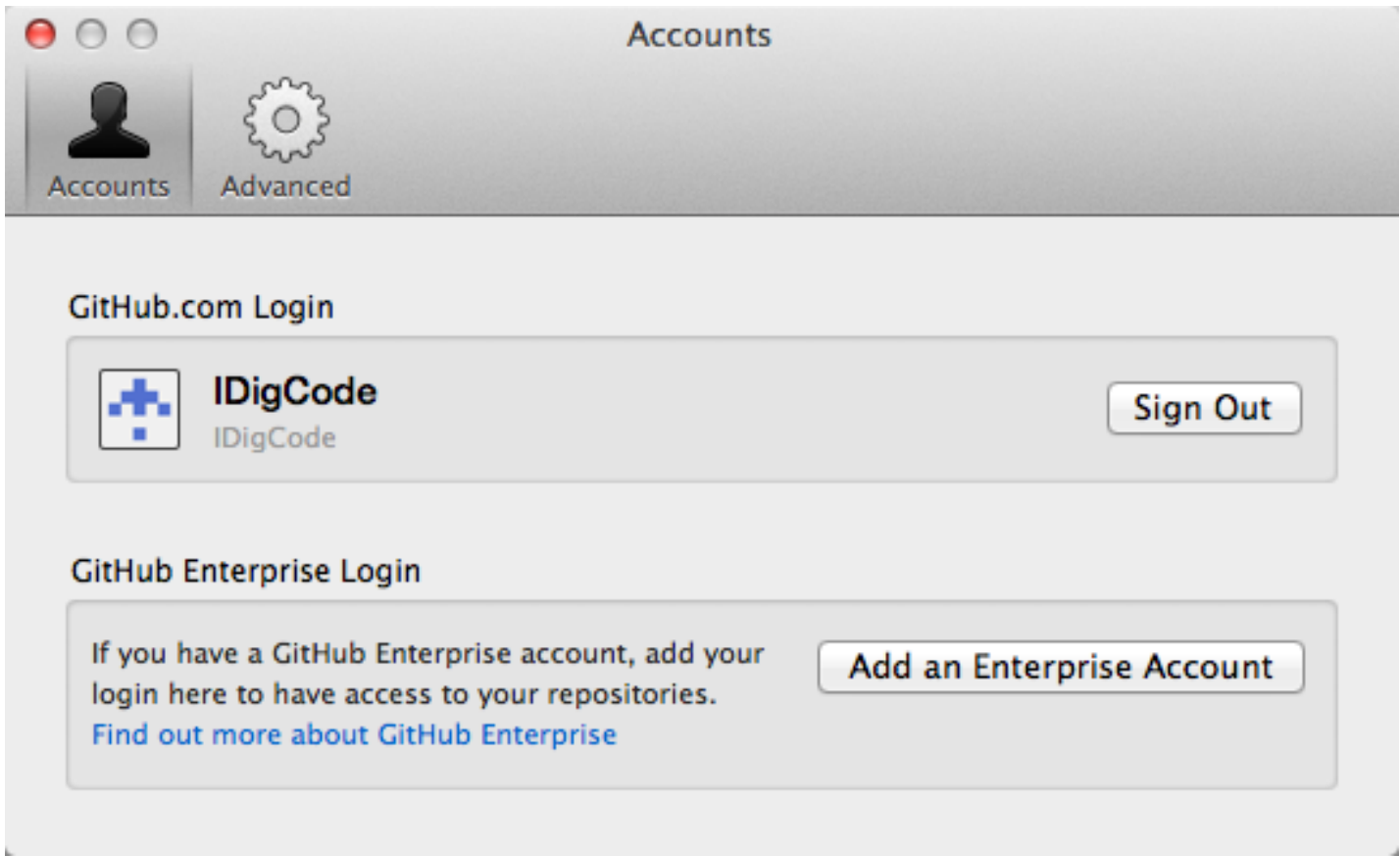
Installing the GitHub app

Visit either windows.github.com (windows.github.com) or mac.github.com (mac.github.com) and download the appropriate desktop client for your system. Although we’re using the Mac version of the desktop client (and GitHub in a Mac-based browser), the two versions of the app look quite different, but function in the same way. If you’re on Windows, here’s a movie from James Williamson’s course *GitHub for Web Designers* that [explains the Windows app and how it differs from the Mac GitHub app](http://www.lynda.com/GitHub-tutorials/Exploring-Windows-client/162276/173464-4.html) (<http://www.lynda.com/GitHub-tutorials/Exploring-Windows-client/162276/173464-4.html>) in more detail.

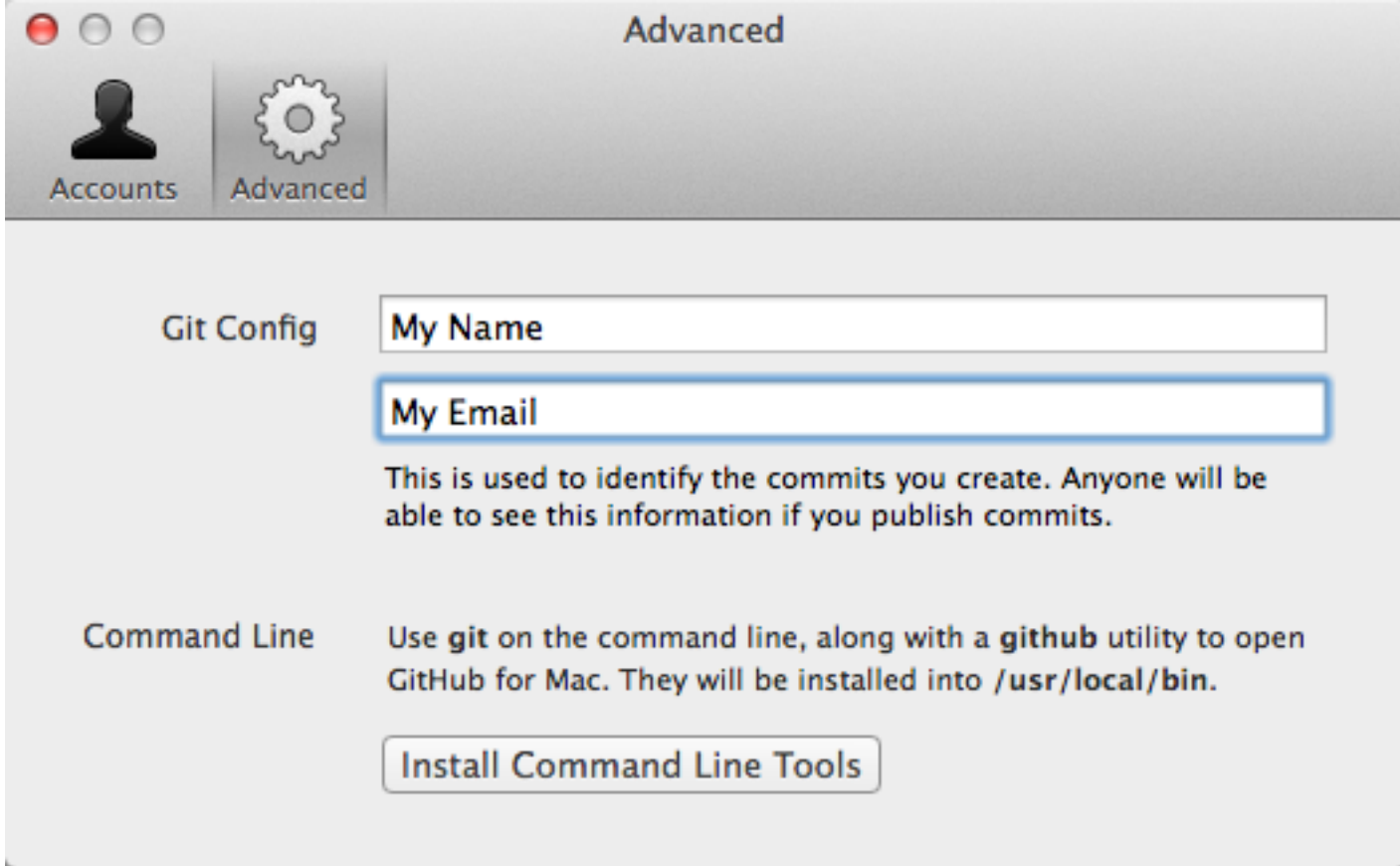
Once you've installed the app, go to its **Preferences** and enter your GitHub login information.



Click **Sign In** and you should be set, with your GitHub profile shown prominently in the resulting screen.



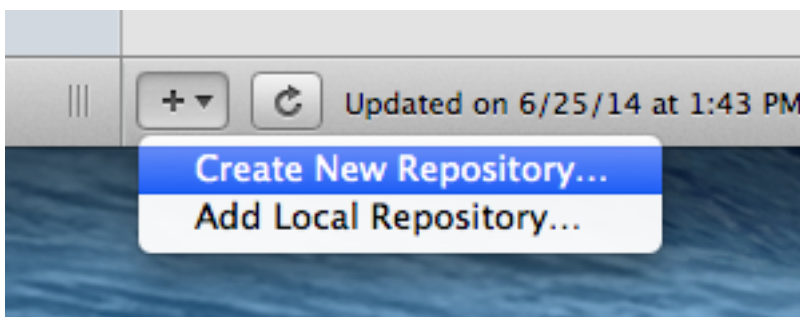
In the **Advanced** tab of the **Preferences**, set your name and email address, so that GitHub can note who did what as it stores your committed changes.



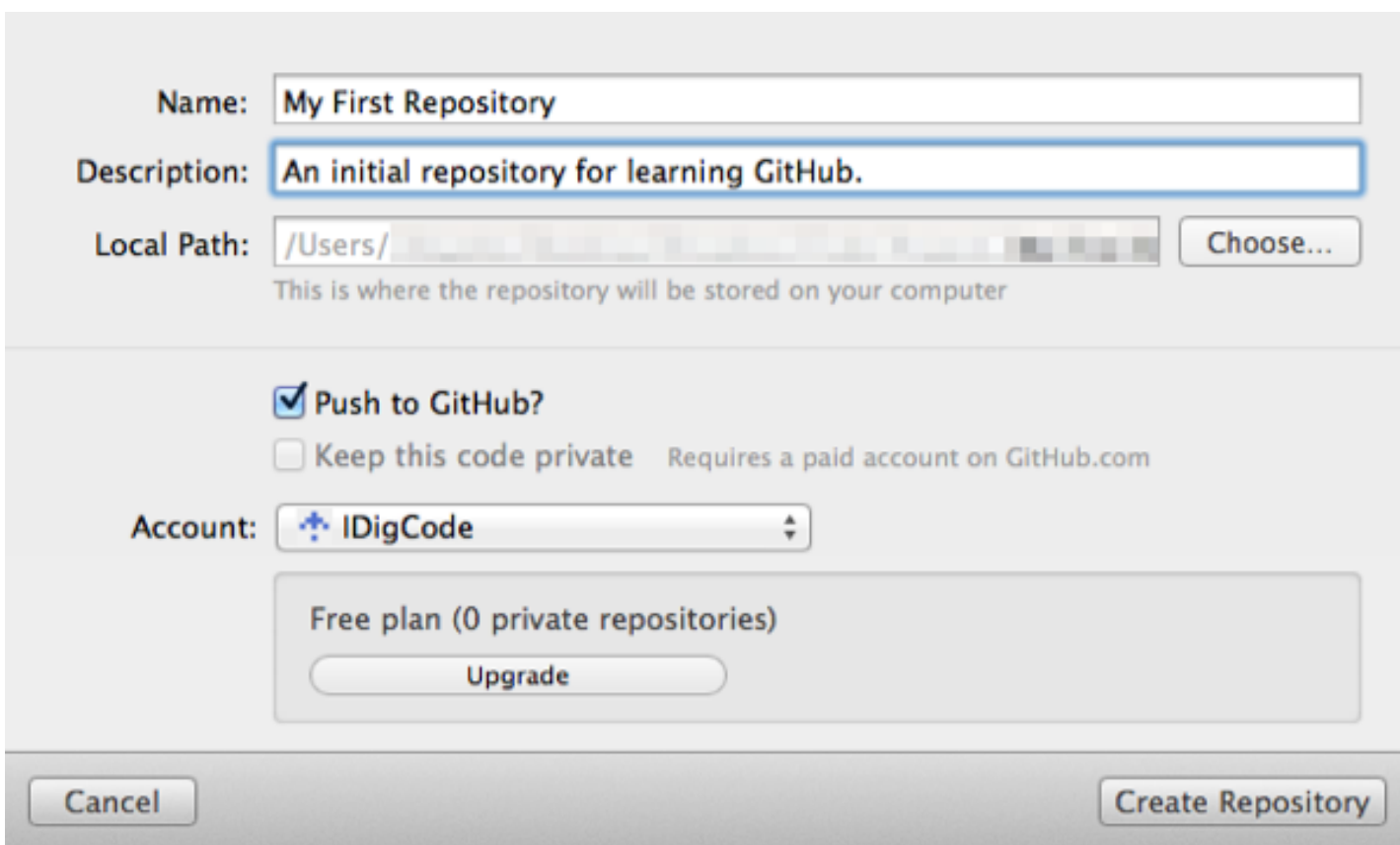
Close the **Preferences** dialog and let's create your first repository.

Creating your first repository

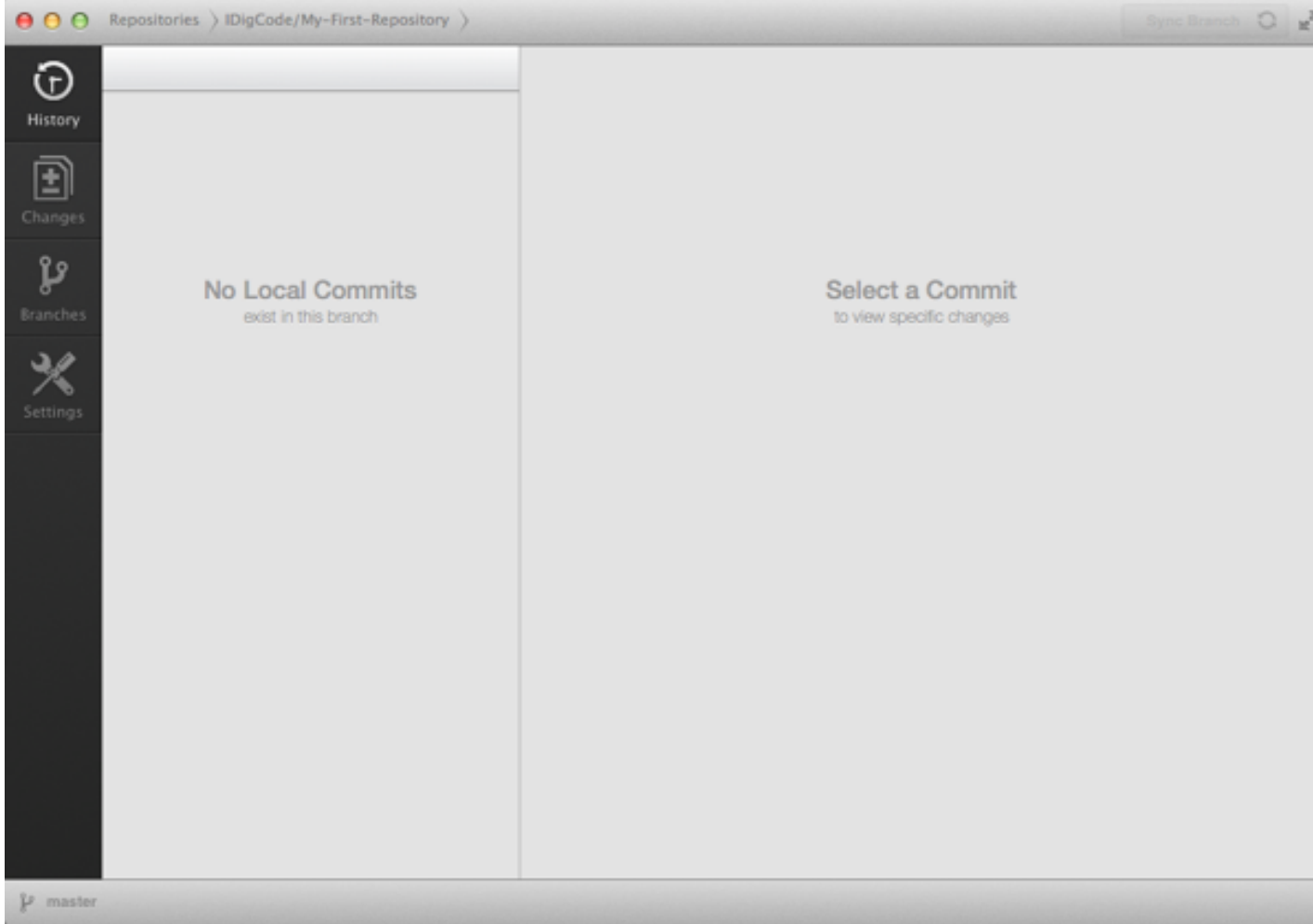
Choose the **Create New Repository...** menu item from the plus-sign button at the bottom of the GitHub app. Side note: the **Add Local Repository...** option lets you create a GitHub repository for an existing directory of files on your system, which is handy if you've already got files on your local system you'd like to create a repository for.



Enter a name for your new repository—in our example case, *My First Repository*. You can also give it a friendly description so you and others recognize it more easily. Be sure to check the Push to GitHub option, as this will create the repository both locally and on the GitHub service. You also have the option to upgrade to a paid GitHub account, should you want to use the greyed-out checkbox, **Keep this code private**. Finally, choose the local directory for your files in the **Local Path** field; this is where you'll eventually put all your project asset files, so Git can manage them.



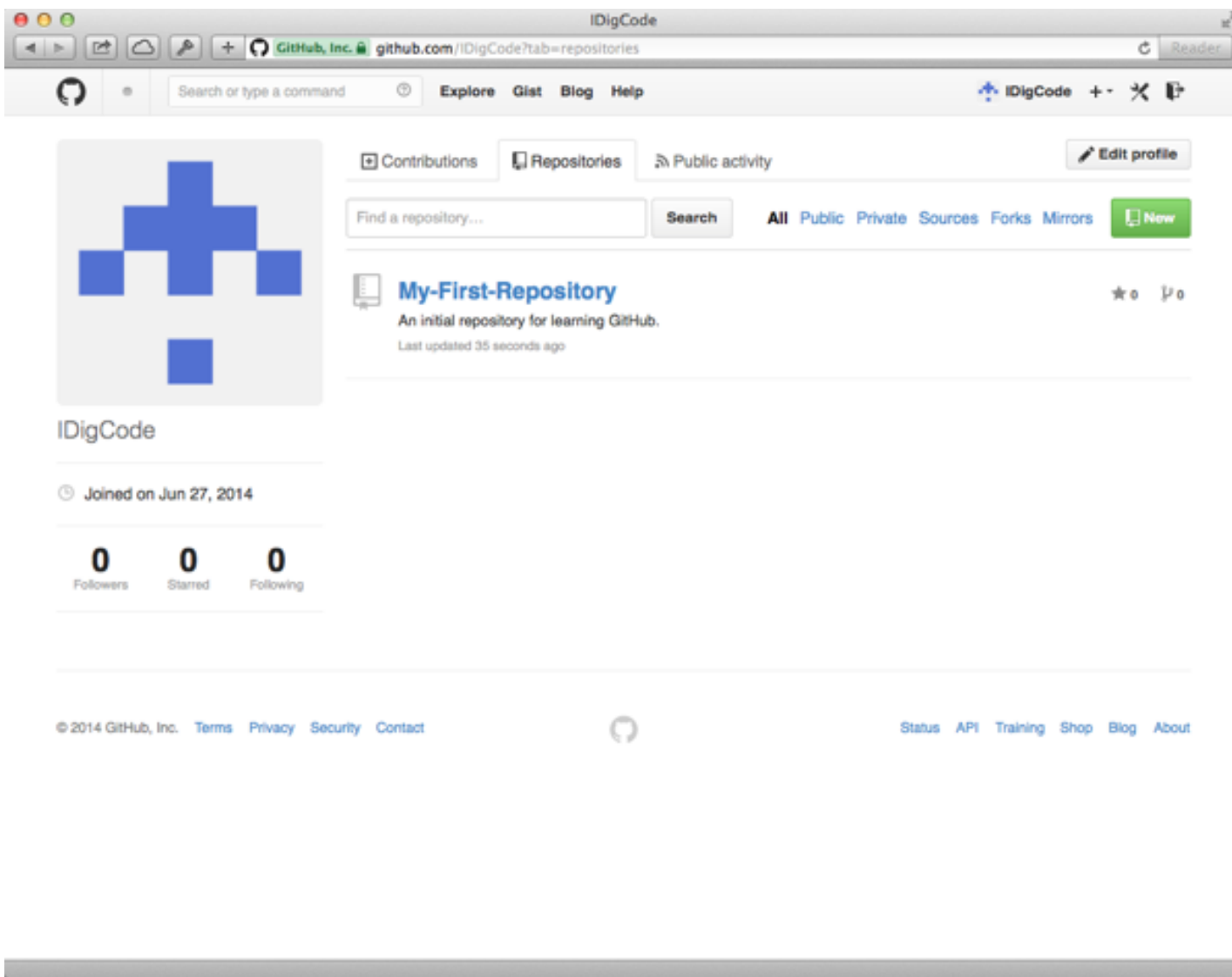
All set? Click the **Create Repository** button when you're done, and you'll get a very different view of the GitHub app. Success!



This is the *History* view of the app, which shows all the changes you’ve made to this project repository—currently empty as you haven’t made any.

Adding your first file to the repository

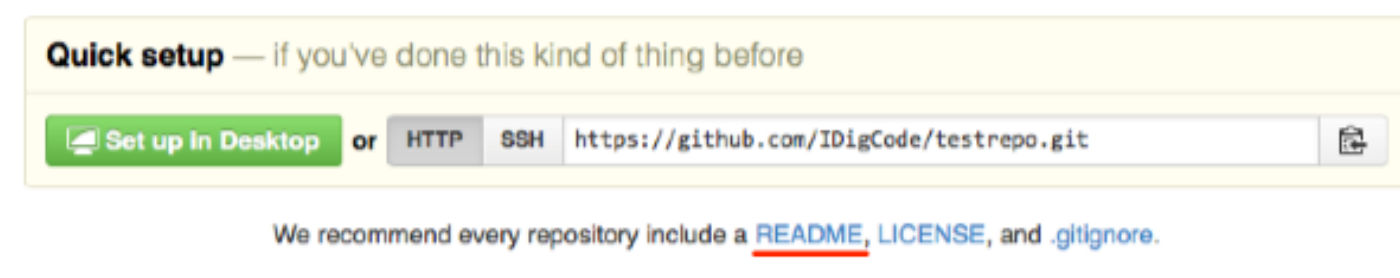
Let’s hop over to your GitHub account in the web browser and see what’s changed there. Refresh (or click on) the **Repositories** tab, and you should see your new Repository listed. Score!



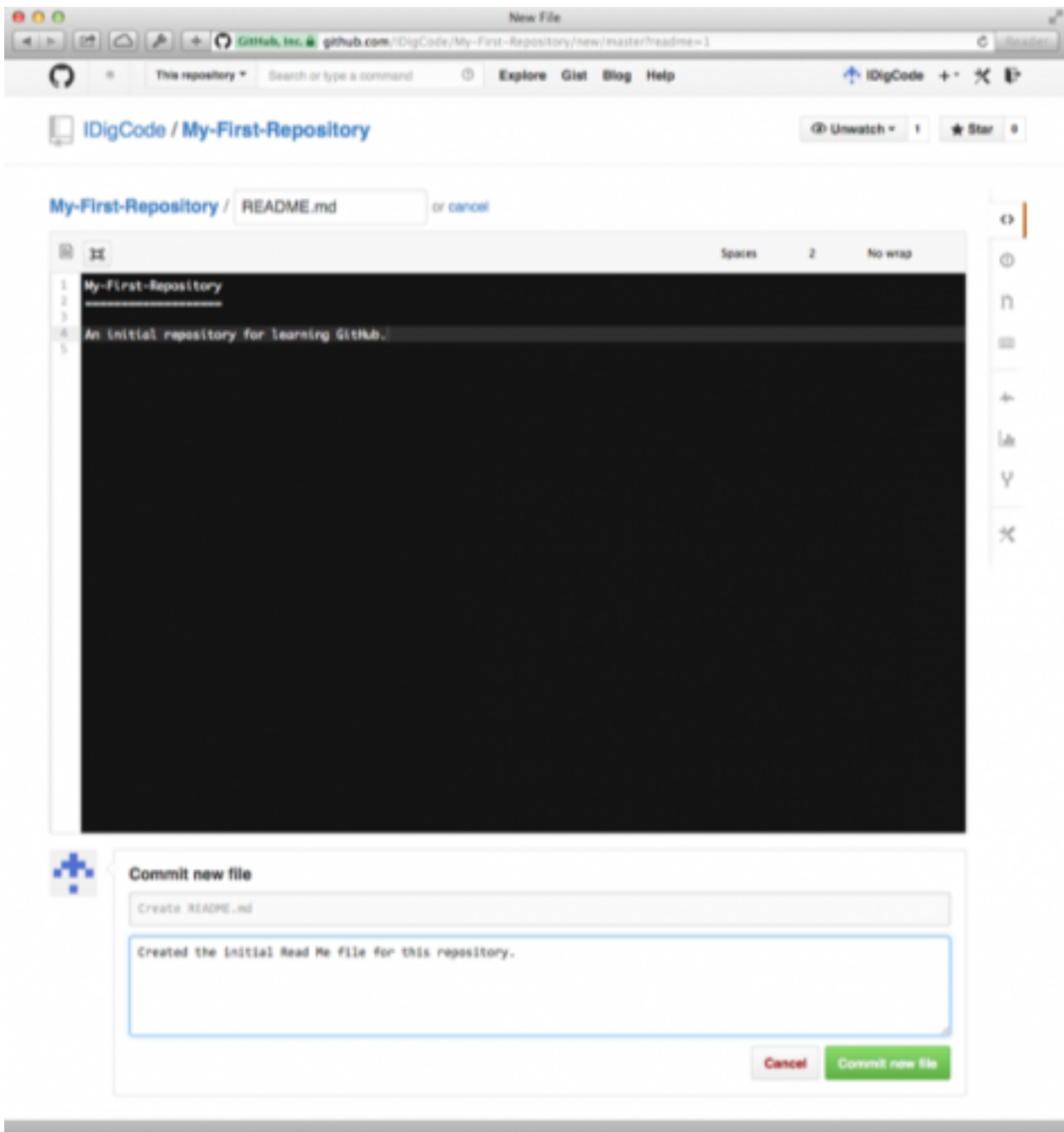
From this view you can get a listing of all the repositories you’re managing with GitHub. Click the title of your new repository to go into the detailed *Repository* view. You’ll notice there’s just no files there now. Let’s change that.

Committing changes – from GitHub

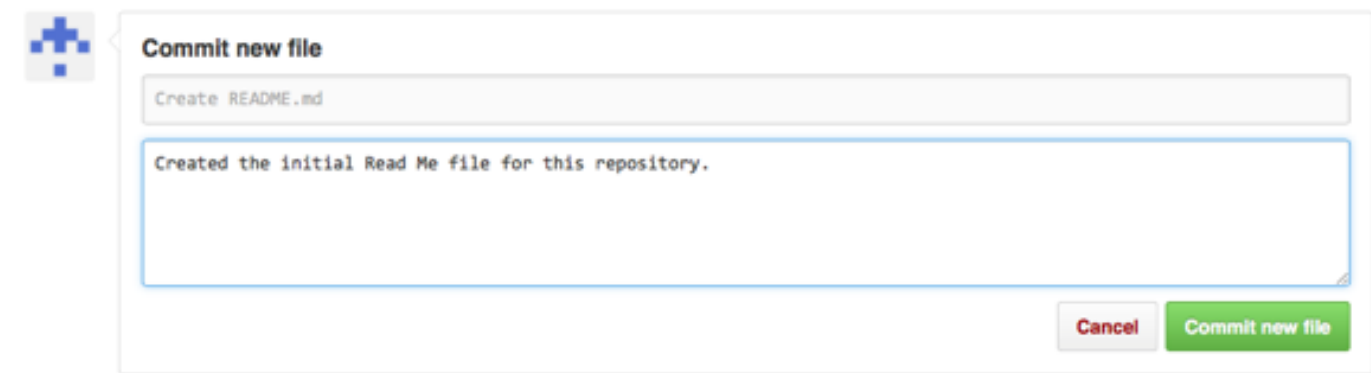
The README file is what’s generally shown by default in this view of a repository, and standard to have in your GitHub project. Fortunately there’s a convenient set of links in a yellow box to help you create a new one. We’ll create a new README file and edit it both in GitHub and on the desktop to illustrate the workflow of using GitHub day to day. Click the **README.md** link underlined in the screenshot below.



You should now see the GitHub online editor (yes, you can even edit text-based files directly on GitHub), with the contents of your new README.md file within it, created from your repository’s title and description.



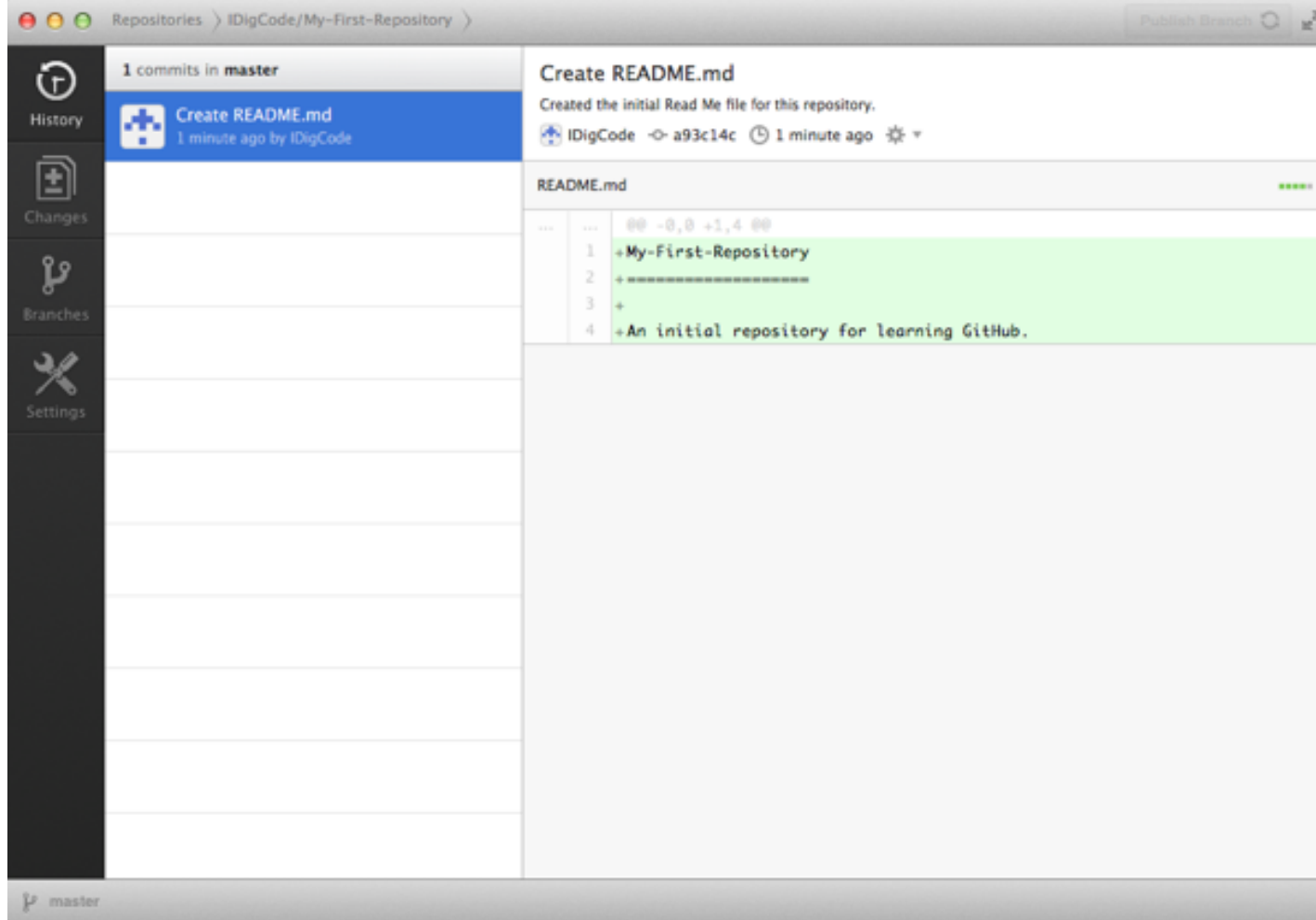
The form at the bottom is the critical point to focus on. But first, edit your README.md file as you see fit, and when you’re ready, head down to that area near the green button and let’s take a peek.



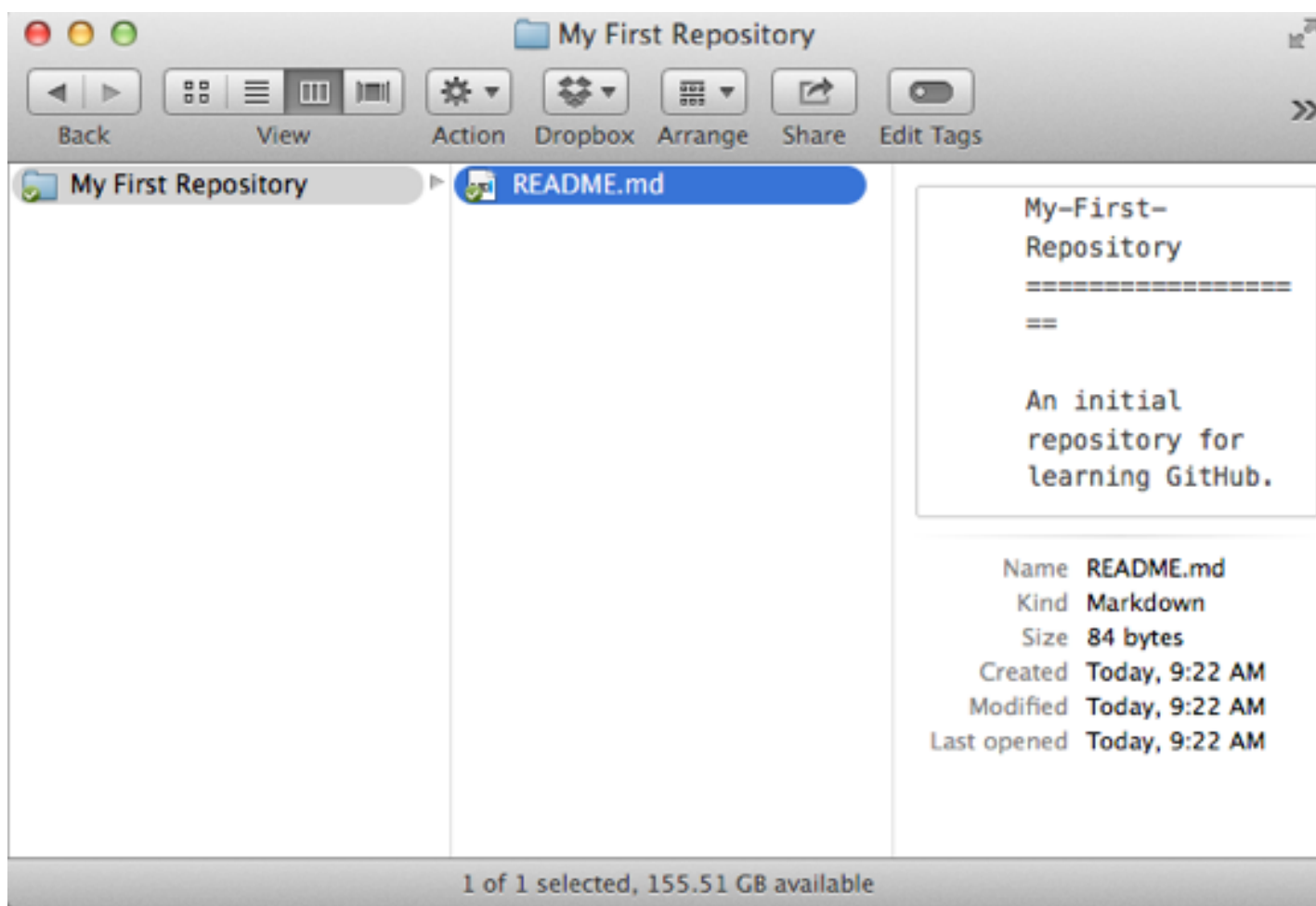
The first field is a summary of the changes; *Create README.md* is the default text (and it’s rather appropriate), but this could be *Changed italic to bold* or *Updated CSS styles with new colors* in the course of a project. If you have more details to add on the changes you’ve made to your file, enter those in the second field, then click the big **Commit new file** button. This will not only add the new file to the repository but register those changes as a version you can roll back to, if needed. The repository view now shows the README.md file’s contents by default, like a project home page.

Syncing remote changes with the GitHub app

Let’s switch back over to the GitHub app on the desktop and see what happened there. Click the **Sync Branch** button in the upper right of the app and you should see a new entry in your *History* reflecting what you just did up on the GitHub site (if not, click the **History** icon in the left-hand menu).



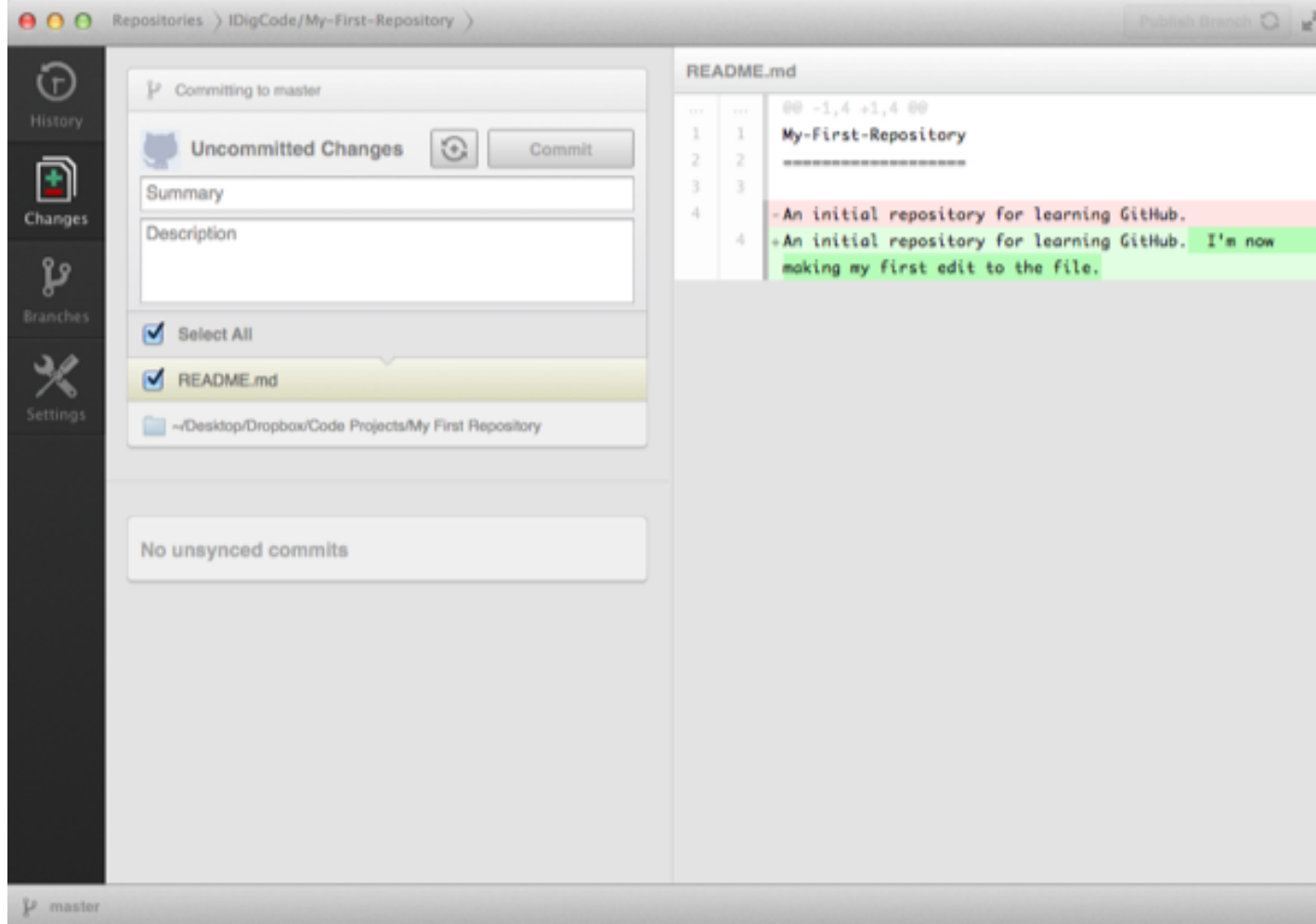
Open the repository directory in the Finder and you should also now see your new README.md file there. It's basically just a text file that supports the Markdown syntax (<http://daringfireball.net/projects/markdown/syntax>) for copy formatting and metadata.



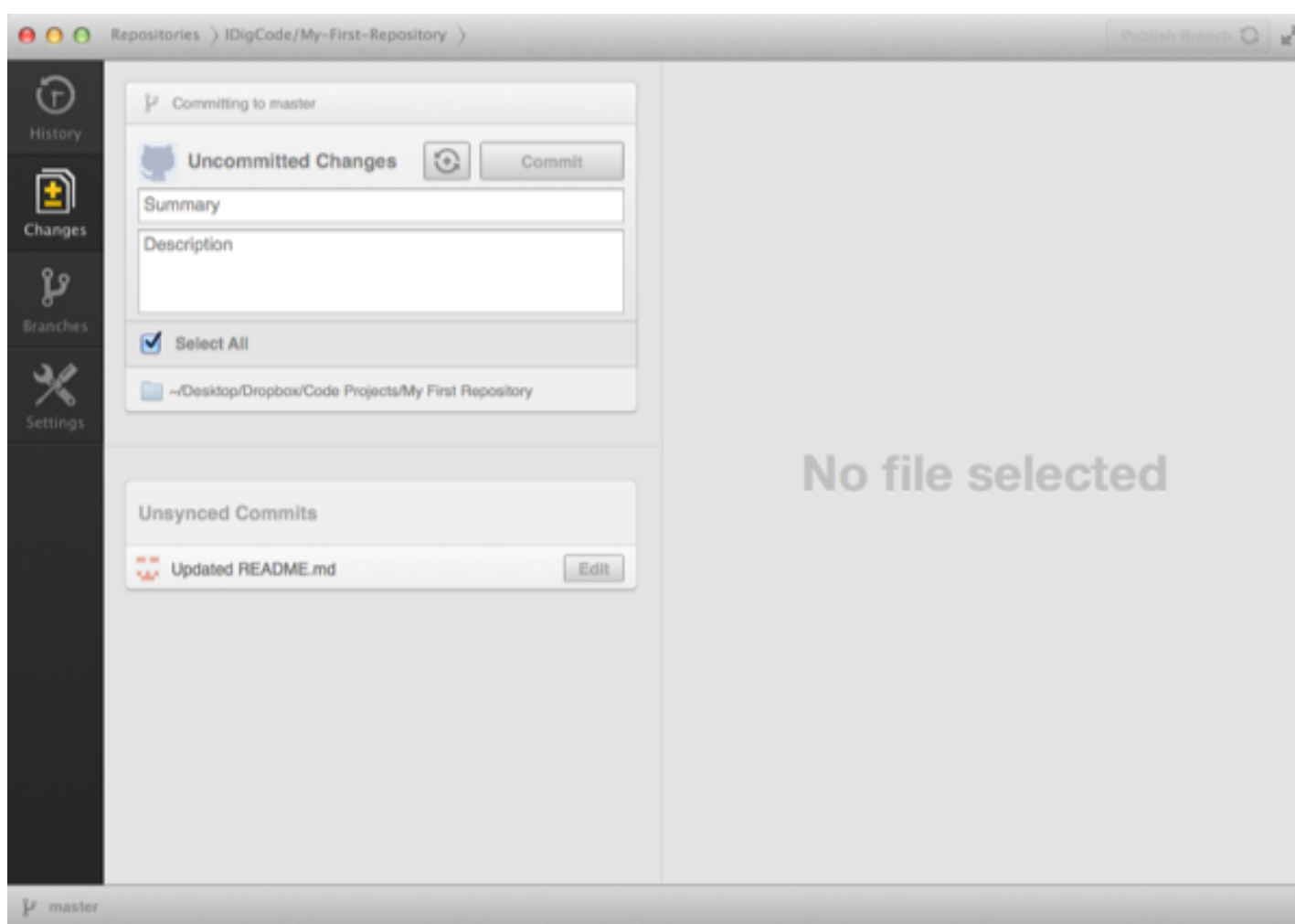
Congratulations! You just synched your local directory with the new file created in your remote repository.

Committing local changes to GitHub

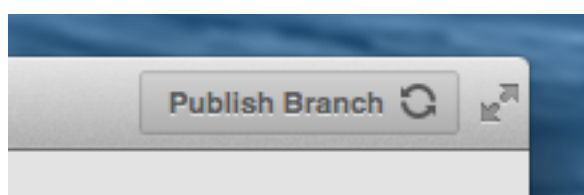
Let's take this a few steps further. Open the README.md file in a text editor (any should do), make a change to it, then save the file and return to the GitHub app and you should see new changes in the *Changes* view:



The red lines are the original lines in our README file, and the green lines are the edits we made to change it. This gives us a quick, easy view of what this set of changes entail without having to open the files directly. We can now commit this snapshot of the changes to our file by entering another *Summary* (and optionally, *Description*) of our changes, and hitting the big **Commit** button. But this time, when we do, something different happens: It shows up as a line underneath the **Unsynced Commits** section.



This is significant, and means that you can work locally, saving individual snapshots of your work without syncing them to GitHub. Ready to sync them all with the master repository on GitHub? Click the **Publish Branch** button at the upper right of the GitHub app window, and you'll do exactly that—no matter how many committed changes you've made.

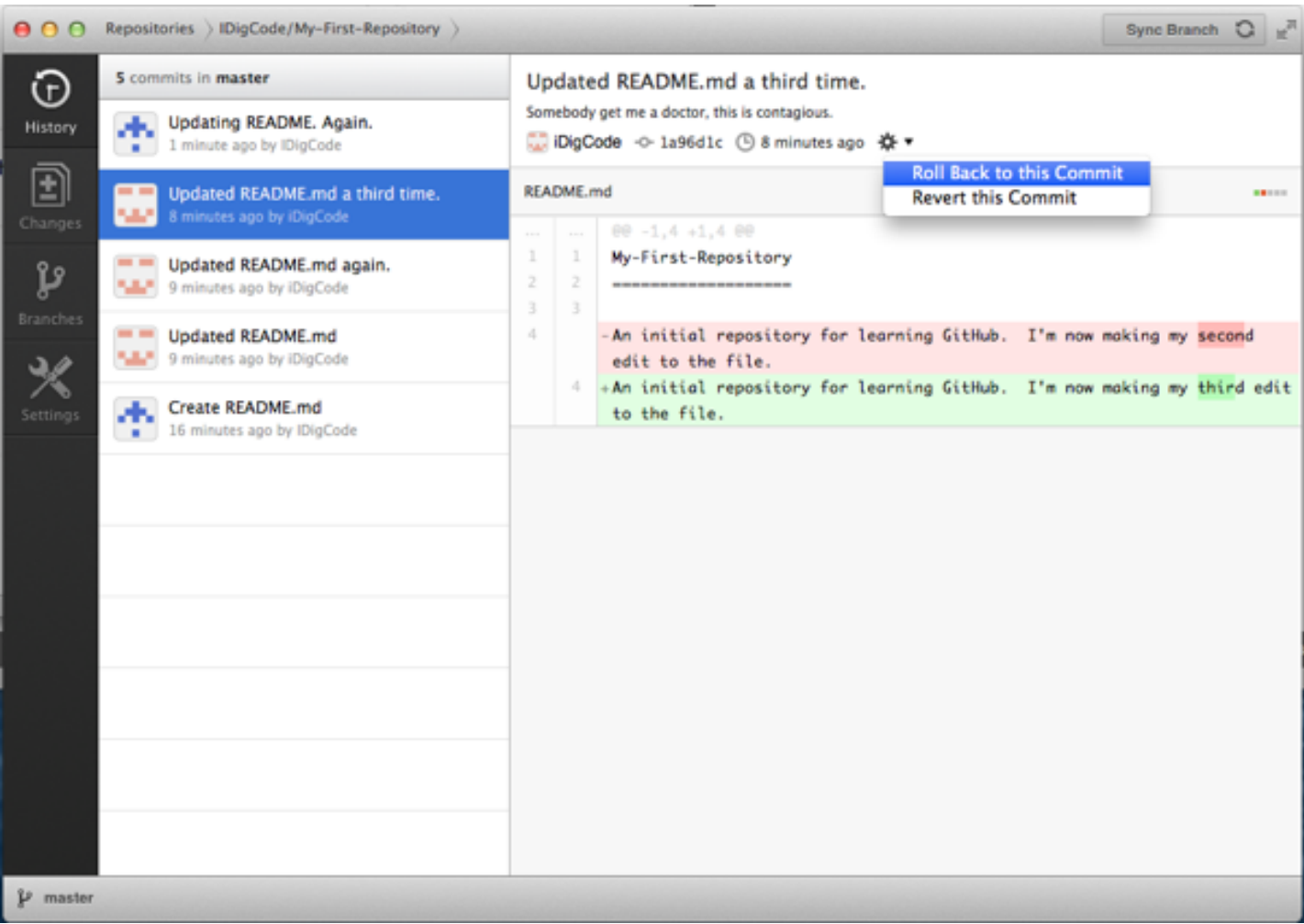


A regular workflow

Now that the repository's been created, use it! You can add files and directories to it in the Finder (or Windows desktop), edit them as you see fit, and commit snapshots of your progress to GitHub for a specific history of your project you can revert to at any time. It's common to leave the app running 24-7 so it can watch your repository directories for changes, and prompt you to commit them.

Rolling Back and Reverting to older versions

Now that you’re tracking your project assets, you’ll inevitably reach the point in which you need to step backwards through your progress and go back to an earlier commit or “save” of your file. Just go to the *History* tab of the GitHub app, find the version you want to roll back to, and click the small gear icon near the header, and you’ll have the option to roll back the project to that snapshot. Simple!



With the GitHub desktop apps, you can easily store and manage your project assets, and also share them with colleagues with the GitHub online service. But there’s a lot more underneath the hood of Git and GitHub to explore. Get a deeper overview of Git (and GitHub) in James Williamson’s course *GitHub for Web Designers* (<http://www.lynda.com/GitHub-tutorials/GitHub-Web-Designers/162276-2.html>) on lynda.com. Happy versioning!

Learn it from the experts.

With online video courses at lynda.com (<http://www.lynda.com/default.aspx>), you can reach your goals faster. Learn software, improve your skills, and get an inside look at how the professionals work.

See what you can learn (<http://www.lynda.com/default.aspx>)

Check out these popular Design courses



Illustrator CC Essential Training
([/Illustrator-tutorials/Illustrator-CC-Essential-Training/122469-2.html](#))

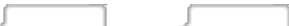


InDesign CC Essential Training
([/InDesign-tutorials/InDesign-CC-Essential-Training/120609-2.html](#))



Illustrator CC One-on-One: Fundamentals
([/Illustrator-tutorials/Illustrator-CC-One-One-Fundamentals/124100-2.html](#))

Share this article:



6

15

0

([http://www.addthis.com/bookmark.php?v=300&winname=addthis&pub=lyndadotcom&source=tbx-300&lng=en-us&s=linkedin&url=http%3A%2F%2Fwww.lynda.com%2Farticles%2Fgithub-rest-us&title=GitHub%20for%20the%20Rest%20of%20Us&ate=AT-lyndadotcom/-/-/54c14c5bf04f24e4/6&frommenu=1&uid=54c14c5bdb497f8f&ct=1&tt=0&captcha_provider=nucaptcha](http://www.addthis.com/bookmark.php?v=300&winname=addthis&pub=lyndadotcom&source=tbx20-300&lng=en-us&s=linkedin&url=http%3A%2F%2Fwww.lynda.com%2Farticles%2Fgithub-rest-us&title=GitHub%20for%20the%20Rest%20of%20Us&ate=AT-lyndadotcom/-/-/54c14c5bf04f24e4/6&frommenu=1&uid=54c14c5bdb497f8f&ct=1&tt=0&captcha_provider=nucaptcha))

(http://www.addthis.com/bookmark.php?v=300&winname=addthis&pub=lyndadotcom&source=tbx-300&lng=en-us&s=linkedin&url=http%3A%2F%2Fwww.lynda.com%2Farticles%2Fgithub-rest-us&title=GitHub%20for%20the%20Rest%20of%20Us&ate=AT-lyndadotcom/-/-/54c14c5bf04f24e4/7&frommenu=1&uid=54c14c5b21c03d3b&ct=1&tt=0&captcha_provider=nucaptcha)

22 (http://www.addthis.com/bookmark.php?v=300&winname=addthis&pub=lyndadotcom&source=sco-300&lng=en-us&s=linkedin&url=http%3A%2F%2Fwww.lynda.com%2Farticles%2Fgithub-rest-us&title=GitHub%20for%20the%20Rest%20of%20Us&ate=AT-lyndadotcom/-/-/54c143b09329e3ca/7/x&frommenu=1&uid=54c143b0ebca6716&ct=1&pre=http%3A%2F%2Fwww.lynda.com%2Fmember.aspx&tt=0&captcha_provider=nucaptcha)

g+1

12

g+1

12

Tags: Web Design (<http://www.lynda.com/articles/tag/web-design>), Scott Fegette (<http://www.lynda.com/articles/tag/scott-fegette>), Git (<http://www.lynda.com/articles/tag/git>), GitHub (<http://www.lynda.com/articles/tag/github>), VCS (<http://www.lynda.com/articles/tag/vcs>), Version Control (<http://www.lynda.com/articles/tag/version-control>)

Site map (<http://www.lynda.com/sitemap>) | Partner program (<https://www.lynda.com/partners>) | Privacy policy (<http://www.lynda.com/aboutus/otl-privacy.aspx>) | Website use policy (<http://www.lynda.com/aboutus/website-use-policy>)