

Account Creation

Gmail : New account creation

(for people creating burner accounts only... otherwise feel free to use your normal e-mail account.)

Visit <http://mail.google.com/mail/signup>.

Fill out the form that appears.

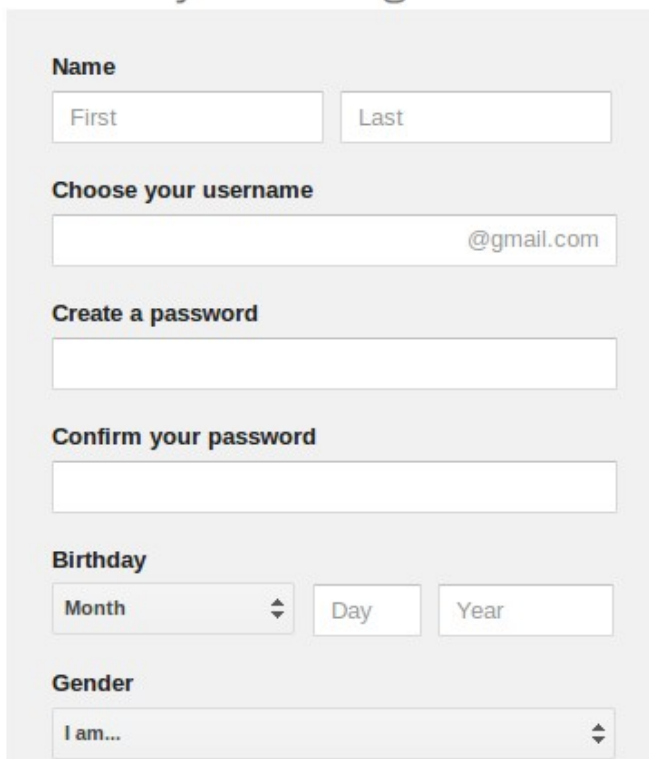
1. Put in an arbitrary name.
2. Username should be “ficlopBurnerX”, where X is a number. Eg, ficlopBurner1, ficlopBurner2, etc.
3. For password, use the same string as you used for username, but reverse the order of the words. Eg

username: ficlopBurner1

password: 1BurnerFiclop

4. Put in an arbitrary birthday and gender.
5. Leave all other fields blank.
6. Complete the image captcha and hit “Next step”.

Create your Google Account



The screenshot shows the 'Create your Google Account' form. It includes fields for Name (First and Last), Choose your username (with a @gmail.com suffix), Create a password, Confirm your password, Birthday (Month, Day, and Year), and Gender (I am... dropdown).

Name		
First	Last	
Choose your username		
[Username]@gmail.com		
Create a password		
[Password]		
Confirm your password		
[Confirm Password]		
Birthday		
Month	Day	Year
Gender		
I am... [Dropdown]		

Github: New account creation

If you are using a burner account, skip ahead to the “Github: Burner Account” section.

1. Visit <https://github.com/>. If you are not signed in as a different user, the page will appear as shown: Create a username, provide a valid e-mail address, and choose a password. Then hit “Sign up for Github”

GitHub

Search GitHub

ExploreFeaturesEnterprisePricing

Sign upSign in

Where software is built

Powerful collaboration, code review, and code management for open source and private projects. Public projects are always free.

Private plans start at \$7/mo.

Pick a username

Your email

Create a password

Use at least one lowercase letter, one numeral, and seven characters.

Sign up for GitHub

By clicking "Sign up for GitHub", you agree to our [terms of service](#) and [privacy policy](#). We will send you account related emails occasionally.

2. A page asking what payment plan you want to use will load. The free option is already chosen, so just click “Finish Sign up”.

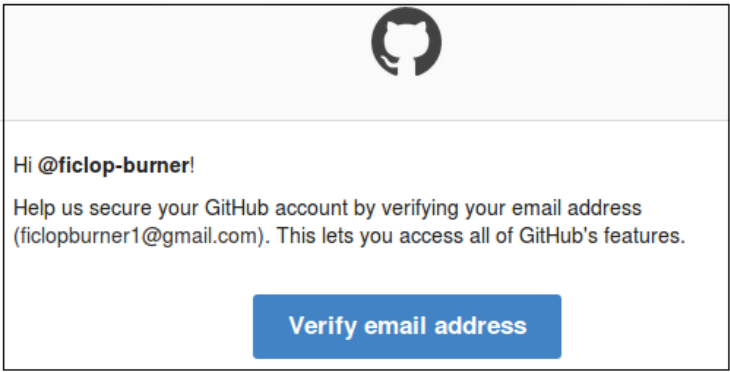
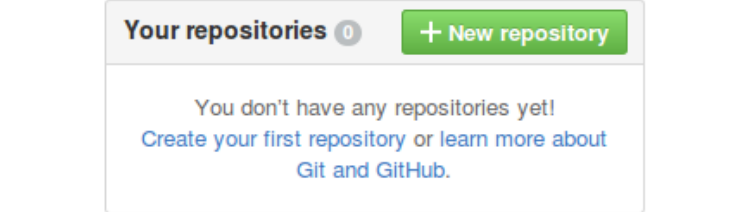
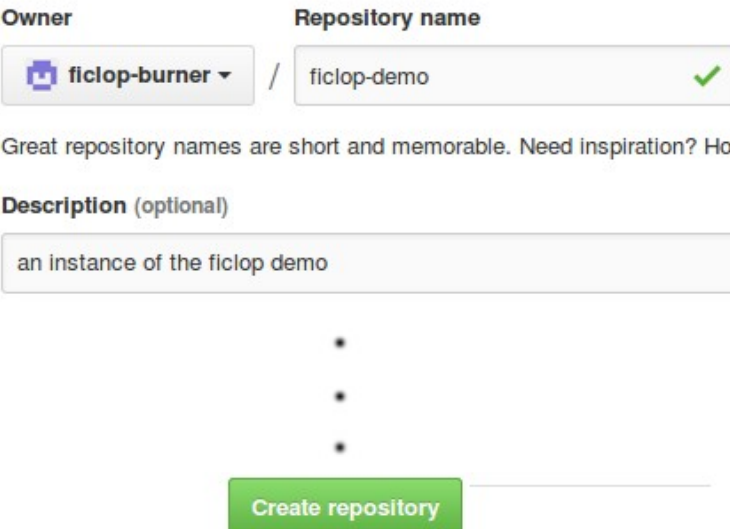
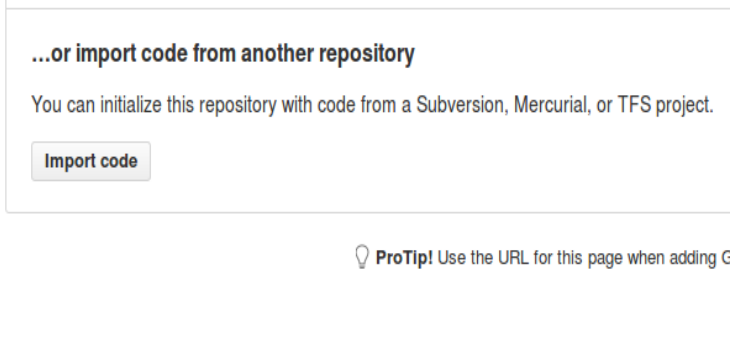
Choose your personal plan

Plan	Cost	Private repositories	
Large	\$50/month	50	Choose
Medium	\$22/month	20	Choose
Small	\$12/month	10	Choose
Micro	\$7/month	5	Choose
Free	\$0/month	0	Chosen

Charges to your account will be made in **US Dollars**. Converted prices are provided as a convenience and are only an *estimate* based on *current* exchange rates. Local prices will change as the exchange rate fluctuates. Don't worry, you can cancel or upgrade at any time.

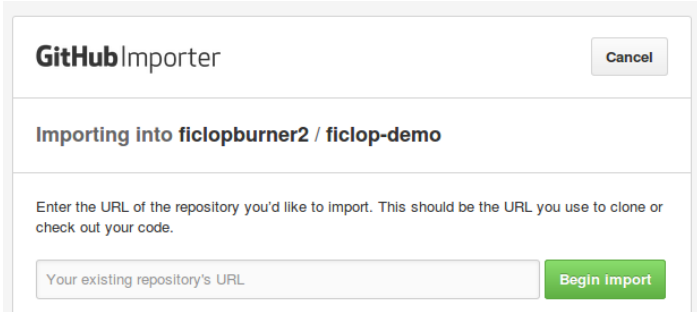
- ☐ **Help me set up an organization next**
Organizations are separate from personal accounts and are best suited for businesses who need to manage permissions for many employees.
[Learn more about organizations.](#)

Finish sign up

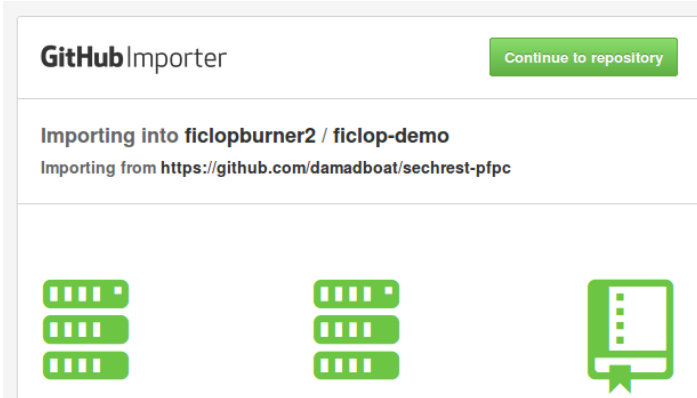
<p>3. You should have received a verification e-mail from GitHub at the e-mail address you provided. Log in, open the e-mail from GitHub, and hit “Verify e-mail address”.</p>	 <p>The screenshot shows the GitHub email verification interface. At the top is the GitHub logo. Below it, the text reads: "Hi @ficlop-burner! Help us secure your GitHub account by verifying your email address (ficlopburner1@gmail.com). This lets you access all of GitHub's features." At the bottom is a blue button labeled "Verify email address".</p>
<p>4. After verifying your GitHub account, you can create a new repository. Return to https://github.com/; after you have logged in, it should appear as shown below. Hit the “+ New repository” on the right.</p>	 <p>The screenshot shows the "Your repositories" section on GitHub. It displays "Your repositories 0" and a green button labeled "+ New repository". Below this, a message states: "You don't have any repositories yet! Create your first repository or learn more about Git and GitHub."</p>
<p>5. Give the new repository a name and a brief description. After that hit “Create repository”; we'll be cloning another repository to get you started, so we can skip repository initialization.</p>	 <p>The screenshot shows the "Create repository" form. It has two main sections: "Owner" with a dropdown menu showing "ficlop-burner" and "Repository name" with a text input field containing "ficlop-demo" and a green checkmark. Below these is a text area for "Description (optional)" containing "an instance of the ficlop demo". At the bottom is a green button labeled "Create repository".</p>
<p>6. Your new repository is currently empty. We're going to duplicate an existing repository to get you started... normally with git, you would create a new branch from the base repository, but for this demo, we'll be giving you your own repository.</p> <p>Hit the “Import code” button at the bottom of the page.</p>	 <p>The screenshot shows the "Import code" section of a new repository. It has a heading "...or import code from another repository" and a subtext: "You can initialize this repository with code from a Subversion, Mercurial, or TFS project." Below this is a button labeled "Import code". At the bottom right, there is a "ProTip!" icon and text: "ProTip! Use the URL for this page when adding C".</p>

7. When the “GitHub Importer” page loads, fill in <https://github.com/ficlop-master/ficlop-demo-master> <https://github.com/damadboat/sechrest-pfpc>

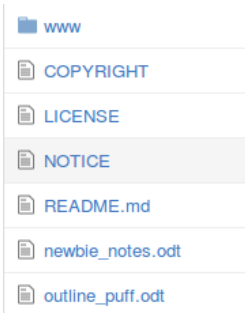
and click “Begin Import”.



8. If the import process is successful, after a few moments the page will appear as shown. Click “Continue to repository” to see what was imported.

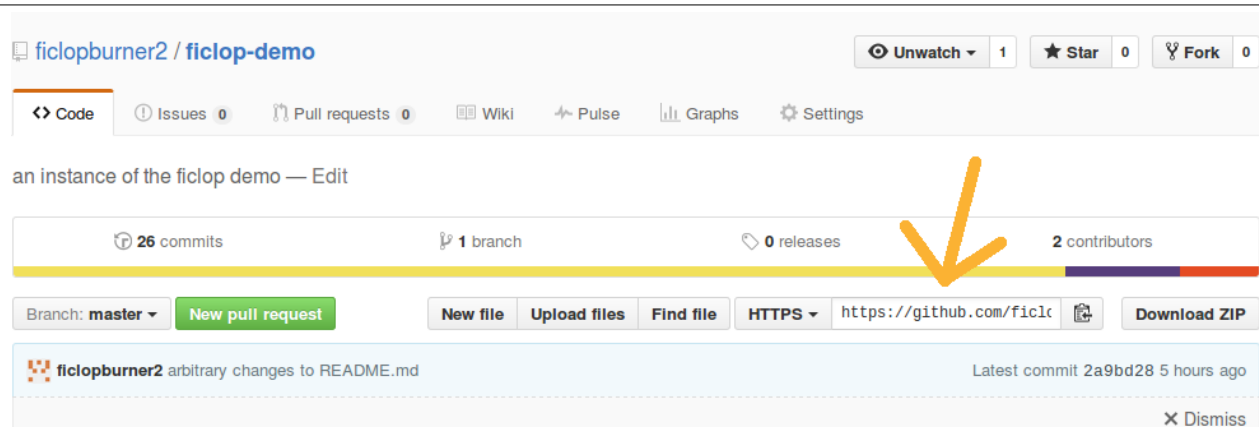


9. If the import process was successful, your repository will appear as shown.




10. We have successfully created a GitHub account. Before we leave, however, please make note of the git repository address associated with this repository; this is important for some of the services we'll be using later. It is recoverable on the main page of your repository, as shown below.

It is usually the same as the URL of your repository, with a “.git” extension added to the end.



Github: Burner Account

If your using one of the burner accounts, visit <https://github.com/>, and instead of creating a new Github account, click “Sign in”. This will take you to the page shown. Login using the credentials given to you by the event organizer.



Sign in to GitHub

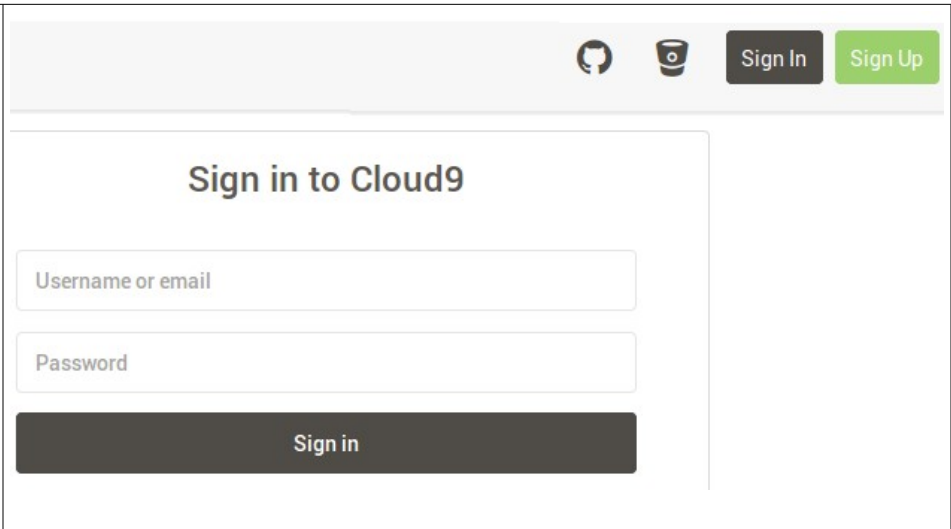
Username or email address

Password [Forgot password?](#)

Sign in

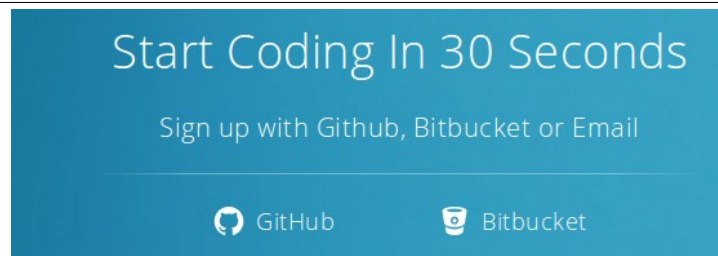
Cloud9: New Account Creation

1. Visit cloud9's login page (<https://c9.io/login>) and click “Sign Up”.



The image shows the Cloud9 login page. At the top right, there are icons for GitHub and Bitbucket, and two buttons: "Sign In" (dark grey) and "Sign Up" (green). The main heading is "Sign in to Cloud9". Below it are two input fields: "Username or email" and "Password". At the bottom is a dark grey "Sign in" button.

2. After the page shown loads, click the button marked “GitHub”.



The image shows a blue banner with the text "Start Coding In 30 Seconds" in white. Below it, in smaller white text, is "Sign up with Github, Bitbucket or Email". At the bottom are two buttons: "GitHub" (with the GitHub logo) and "Bitbucket" (with the Bitbucket logo).

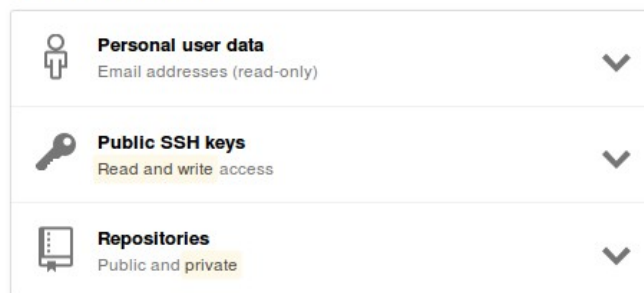
3. If you are logged into your Github account, Github will open a page asking you to authorize cloud9's permission request. Click “Authorize application.”

If you are not logged into your Github account, login and try again.




Authorize application

Cloud9 by @c9 would like permission to access your account

Review permissions



The image shows the "Review permissions" section. It contains three items, each with an icon, a title, a description, and a dropdown arrow:

-  **Personal user data**
Email addresses (read-only)
-  **Public SSH keys**
Read and write access
-  **Repositories**
Public and private

Authorize application

4. After giving permission to Cloud9, Cloud9 may ask you to provide your e-mail address in a pop-up window. If so, provide it.

Oh, one more thing...

Your e-mail address

Proceed

Why do we need this?

In order for us to send you notifications of your workspaces we need your email address. Unfortunately your email address isn't provided by the service you selected. Please enter it to continue.

5. To create a new Cloud9 workspace with our GitHub project as a template, click “Create a new workspace.”



ficlopburner2

Workspaces

Shared With Me

Repositories

YOUR CLOUD9 SUBSCRIPTIONS

Free

Upgrade

Workspaces



Create a new workspace

6. Fill out the form that loads as you see fit. In the “Clone from Git or Mercurial” field, specify the Github account that you created in the previous section.

Create a new workspace

Owner

ficlopburner2

Workspace name

ficlop-demo

Description

an instance of the ficlop demo

Hosted workspace

Remote SSH Workspace



Private

This is a workspace for your eyes only



Public

This will create a workspace for everybody to see

Clone from Git or Mercurial URL (optional)

https://github.com/ficlopburner2/ficlop-demo

Choose a template



Custom



HTML5



Node.js



Meteor



PHP, Apache & ...



Python

django

Django



Ruby



C++



Wordpress

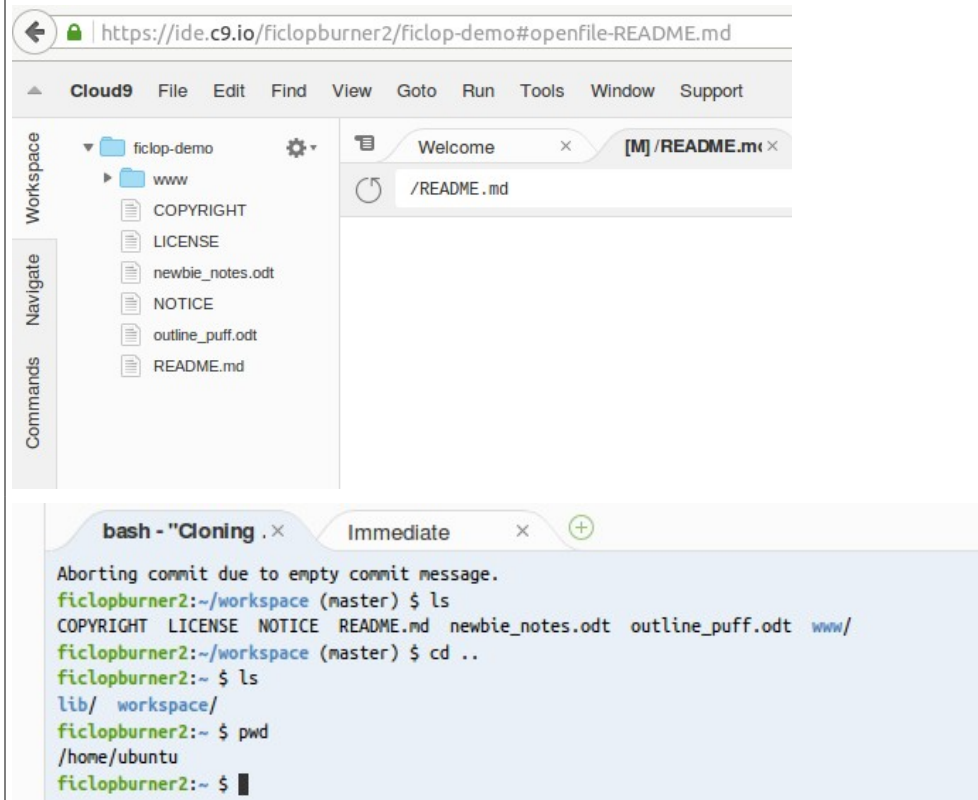


Rails Tutorial

Create workspace

7. If the workspace creation process is successful, Cloud9 will open the newly created project. Shown here is the workspace navigator tab for that project. The directory structure should appear as it does in GitHub.

8. Also of interest is Cloud9's "bash" terminal, shown here. This is a typical bash terminal for an Ubuntu system.



You may now edit this source freely. However, like any project that uses Git, changing your Cloud9 project will not affect the contents of your GitHub account unless you use Git to commit those changes. Here we will only describe the bare essentials to committing your Cloud9 changes back into GitHub; Google can suggest many more "full-blooded" Git tutorials if you want more information.

1. To specify to Git that a file has changed, use the following command in Cloud9's bash terminal:

```
git add {changed files}
```

You can use wildcards or regular expressions(?) when specifying files.

2. To commit any changes you've made to the local Git repository (NOT GitHub!), use the following command in Cloud9's bash terminal:

```
git commit -m "A message describing what changes there are in the commit you're about to add."
```

Git requires some kind of message associated with each commit. You may skip the "-m" flag in the command, but if you do you must fill one in via 'nano' before Git will accept your commit.

3. To push the changes in your local Git repository out to GitHub, use the following command in Cloud9's bash terminal:

```
git push
```

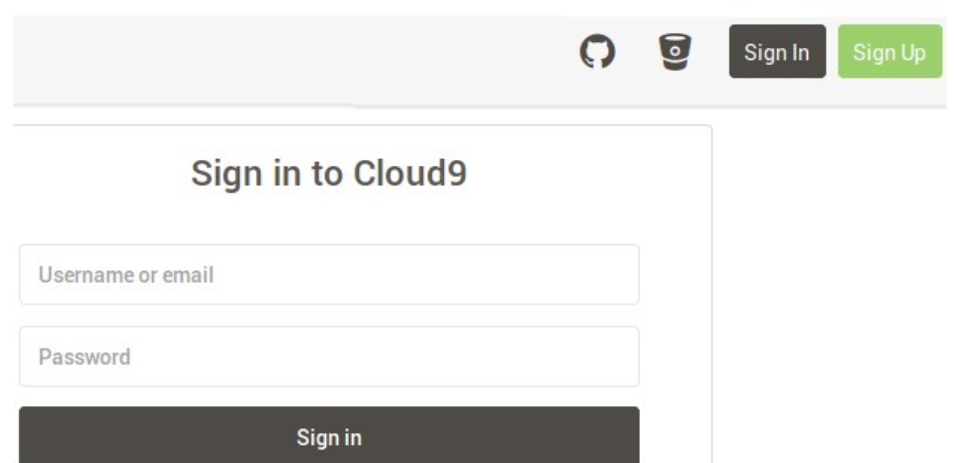
This command requires you to provide your GitHub credentials.

When using the builtin Cloud9 terminal, the whole process, if it is successful, will look as shown below:

```
ficlopburner2:~/workspace (master) $ git add README.md
ficlopburner2:~/workspace (master) $ git commit -m "arbitrary changes to README.md"
[master 2a9bd28] arbitrary changes to README.md
1 file changed, 2 insertions(+), 2 deletions(-)
ficlopburner2:~/workspace (master) $ git push
Username for 'https://github.com': ficlopburner2
Password for 'https://ficlopburner2@github.com':
Counting objects: 5, done.
Delta compression using up to 8 threads.
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 306 bytes | 0 bytes/s, done.
Total 3 (delta 2), reused 0 (delta 0)
To https://github.com/ficlopburner2/ficlop-demo
fc411c9..2a9bd28  master -> master
ficlopburner2:~/workspace (master) $
```

Cloud9: Burner Account



If your using one of the burner accounts, visit <https://c9.io/login>, and instead of creating a new Cloud9 account, login using the credentials given to you by the event organizer.



The screenshot shows the Cloud9 login interface. At the top, there's a navigation bar with the GitHub logo, the Cloud9 logo, and two buttons: 'Sign In' (dark grey) and 'Sign Up' (green). Below this, the main heading is 'Sign in to Cloud9'. Underneath the heading is a login form with two input fields: 'Username or email' and 'Password'. At the bottom of the form is a dark grey button labeled 'Sign in'.

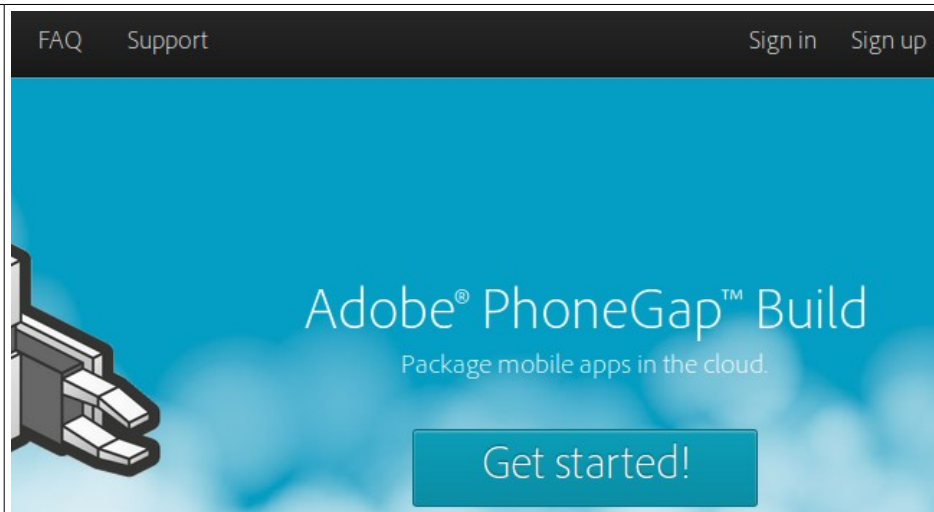
Phonegap-Build: New Account Creation

Phonegap-build is a service supported by Adobe, and we need an Adobe account before we can use it.

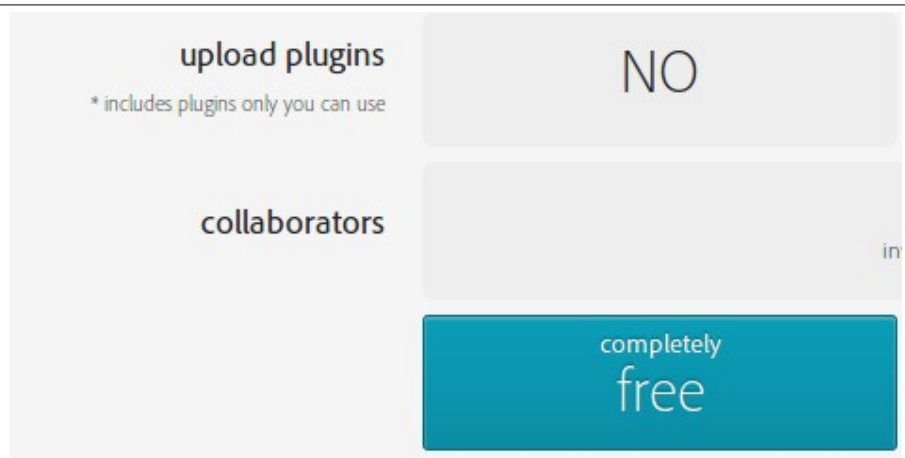
1. Visit the Adobe Account creation page at https://accounts.adobe.com/ and click “Get an Adobe ID”.	<div><p>FOR YOUR PROTECTION, PLEASE VERIFY YOUR IDENTITY.</p><div>Email address</div><div>Password</div><div>Forgot password?</div><div>SIGN IN</div><div>Not a member yet? Get an Adobe ID</div></div>
2. Fill out the form and click “Sign Up”.	<div><div>UserMcUser</div><div>ficlopburner2@gmail.com</div><div>.....</div><div>United States</div><div>Date of birth</div><div>January11999?</div><div><input type="checkbox"/> Stay informed about Adobe products and services. Learn more.</div><div>By clicking "Sign up" I agree that I have read and accepted the Terms of Use and Privacy Policy.</div><div><div><div><div></div></div><div>I'm not a robot</div></div><div> reCAPTCHA Privacy - Terms</div></div><div>SIGN UP</div></div>
3. Adobe requires you to confirm the e-mail address you provided before using Phonegap build. You should receive the e-mail momentarily after clicking “Sign Up”.	

With an Adobe account, we can now access the Phonegap-Build service

1. Visit Phonegap-Build's page (<https://build.phonegap.com/>) and click "Sign up".

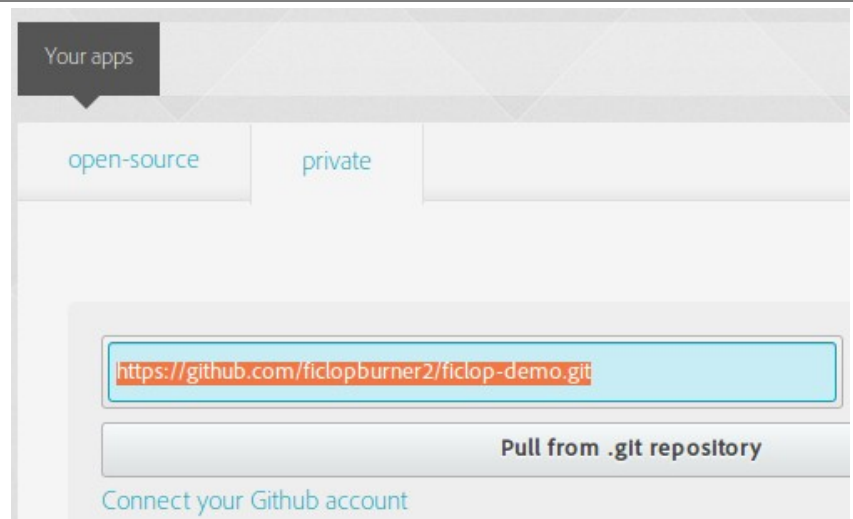


2. The page that loads asks you to pick a payment plan for your account. For our purposes, the free account is sufficient.



3. If you just created your Adobe account, you'll already be logged in. You should see your "Apps" page with no apps present, as shown. To download your app to Phonegap-build, fill in the git repository associated with your GitHub account and click Pull from .git repository.

Note that this is NOT the same thing as the URL of your GitHub account. See the end of the GitHub section for more details.



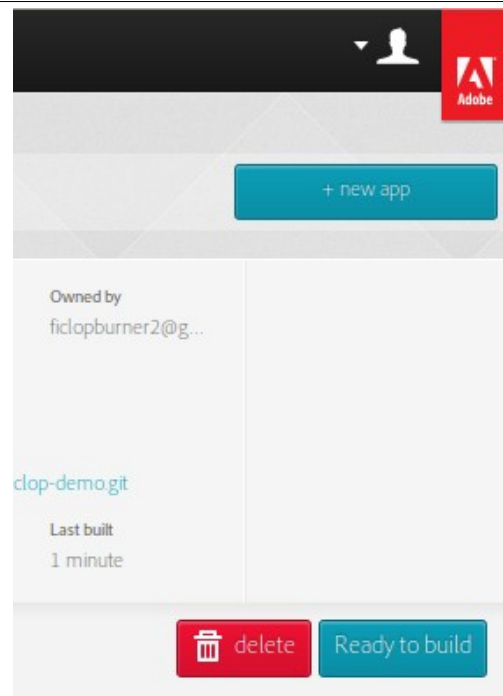
4. If the download was successful, you'll see a new "Hello World" application in your apps page, as shown.



We can also build this sample app and download it to a phone.

1. Your apps page also has a "Ready to Build" button that will kick off a build. Click it.

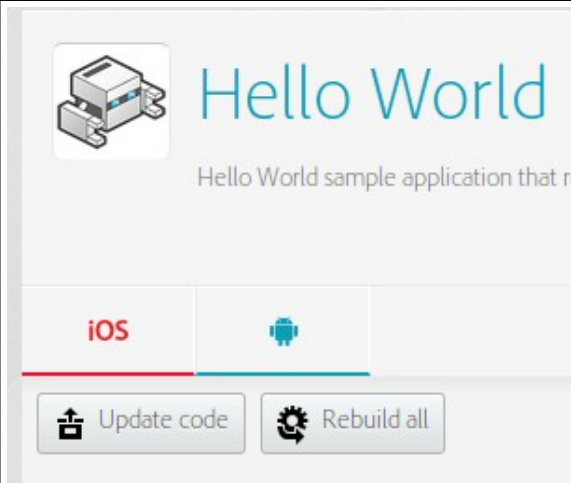
This button only appears if the app has never been built before. Subsequent builds use a slightly different presentation.



2. When your app starts building, status bars for iOS and Android will appear, and show a grey “In progress” animation; when the build succeeds or fails, they will stabilize; red indicates a failed build, blue a successful one, as shown.

Once at least one build has been completed, the “Update code” and “Rebuild all” buttons will appear.

- “Update code” will re-download your app code from GitHub; this must be done manually when you make changes to GitHub.
- “Rebuild All” will attempt to build your app for all supported phone operating systems (iOS and Android).



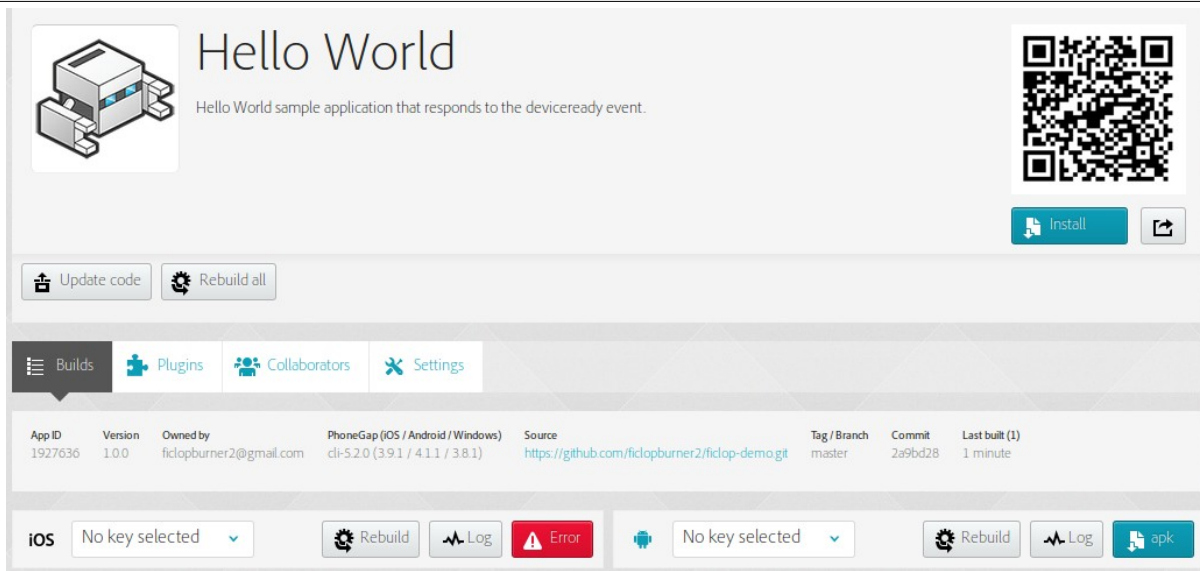
3. Once your app has been built at least once, you can get more detailed controls by clicking on the app name.



4. The detailed app screen is shown below. This page has several useful features.
- The settings tab allows you to customize the outermost features of the app (like name and description), and also allows you to delete the app (as we're only permitted one app, this is useful for sanity testing).
 - Along the bottom are three buttons. The first two are the “Rebuild” and “Log” buttons. These two buttons will attempt another build, and view the compiler log for the last build, respectively.

If the build was successful, the third button will be blue, and will queue your app to be downloaded. The downloaded file can be run as a native iOS or Android application.

If the build was not successful, the third button will be red, and will give you more information about what went wrong during the build.



5. Visit this page (<https://build.phonegap.com/apps>) on your phone and click the download link to download the app directly to your phone.

