**GE Renewable - GitHub Document**

**Step 1: GitHub Access**

* Get SSO ID from GE team
* Get GitHub access from your authorized person (Project Owner)

**Step 2: Git Installation**

**Windows:**

* Download *Git* from [Git for Windows](http://msysgit.github.io/) and install it.

**Linux:**

* [Open a terminal window](https://help.ubuntu.com/community/UsingTheTerminal)**. Copy & paste the following** into the terminal
* You may be prompted to enter your password

sudo apt-get update

sudo apt-get upgrade

sudo apt-get install git

## Step 3: Adding a new SSH key to your GitHub account

To configure your GitHub account to use your new (or existing) SSH key, you'll also need to add it to your GitHub account.

Before adding a new SSH key to your GitHub account, you should have:

**To Checked for existing SSH keys:**

* Open Git Bash.
* Enter ls -al ~/.ssh to see if existing SSH keys are present:
* ls -al ~/.ssh
* # Lists the files in your .ssh directory, if they exist

By default, the filenames of the public keys are one of the following:

* *id\_rsa.pub*
* *id\_rsa*

If you have those files open that id\_rsa.pub with any editor and copy the content

**To Generate the new SSH Keys:**

* Open Git Bash.
* Paste the text below, substituting in your GitHub email address.

ssh-keygen -t rsa -b 4096 -C "*your\_email@example.com*"

* When you're prompted to "Enter a file in which to save the key," press Enter. This accepts the default file location.

Enter a file in which to save the key (/c/Users/*you*/.ssh/id\_rsa):*[Press enter]*

Enter passphrase (empty for no passphrase): *[Type a passphrase]*

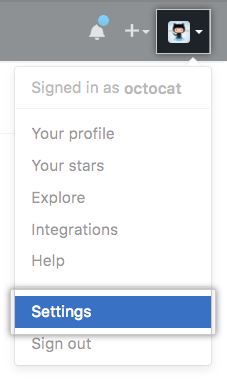
Enter same passphrase again: *[Type passphrase again]*

* Copy the SSH key to your clipboard.

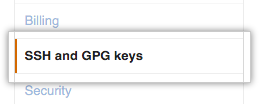
clip < ~/.ssh/id\_rsa.pub

# Copies the contents of the id\_rsa.pub file to your clipboard

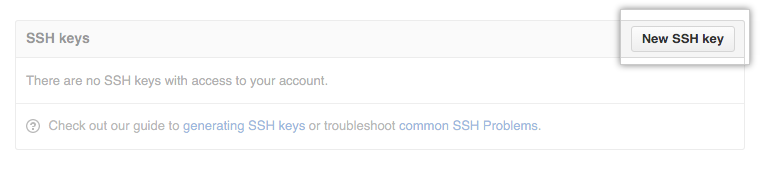
* Go the Github - In the upper-right corner of any page, click your profile photo, then click **Settings**.



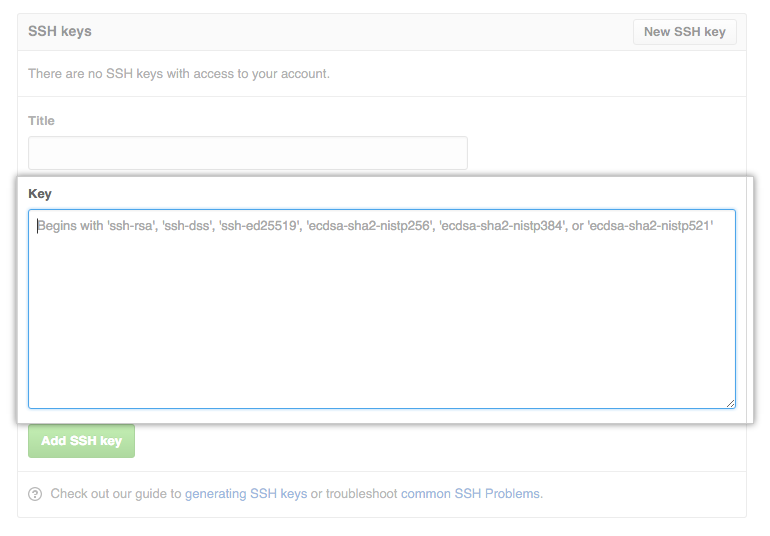
* In the user settings sidebar, click **SSH and GPG keys**.



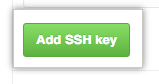
* Click **New SSH key** or **Add SSH key**



* In the "Title" field, add a descriptive label for the new key. For example, you might call this key "Test Key".
* Paste your key into the "Key" field.



* Click **Add SSH key**.

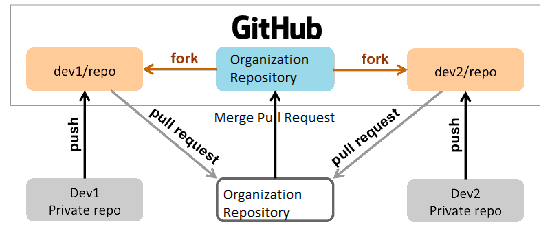


## Step 3: Fork a repo & Setting in local

* A fork is a copy of a repository. Forking a repository allows you to freely experiment with changes without affecting the original project.
* Fork the repository.
* Make the code on your own repository.
* Submit a pull request to the project owner.

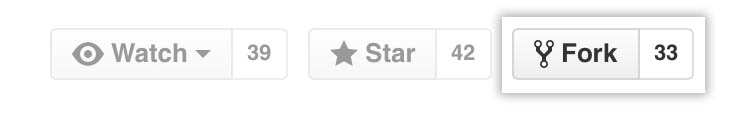
If the project owner likes your work/code, they might pull your code into the organization repository!

**Work Flow:**



Forking a repository is a simple two-step process

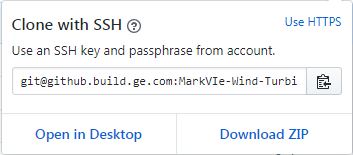
* On GitHub, navigate to the Ge-Renewable/Sample repository.
* In the top-right corner of the page, click **Fork**.



#### Create a local clone of your fork repository

* On GitHub, navigate to **your fork** repository
* Under the repository name, click **Clone or download**.
* In the Clone with SSH section, click to copy the clone URL for the repository.

Clone or download button



* Open Git Bash/Terminal
* Type git clone, and then paste the URL you copied in.

git clone https://github.com/*YOUR-SSOID*/EdgeControl

* Press **Enter**. Your local clone will be created.

git clone https://github.com/YOUR-SSOID/EdgeControl

Cloning into `Edge-Control`...

remote: Counting objects: 10, done.

remote: Compressing objects: 100% (8/8), done.

remove: Total 10 (delta 1), reused 10 (delta 1)

Unpacking objects: 100% (10/10), done.

Now, you have a local copy of your fork of the EdgeControl(Sample) repository!

#### **Step 4: Configure Git to sync your fork with the organization repository**

* Open Git Bash
* To go to your home directory, type just cd with no other text.
* To list the files and folders in your current directory, type ls.
* To go into one of your listed directories, type cd your\_listed\_directory.
* To go up one directory, type cd ...
* Type git remote -v and press **Enter**. You'll see the current configured remote repository for

your fork.

git remote -v

origin https://github.com/YOUR\_USERNAME/YOUR\_FORK.git (fetch)

origin https://github.com/YOUR\_USERNAME/YOUR\_FORK.git (push)

* Type git remote add upstream, and then paste the URL you copied, and press **Enter**.
* To verify the new upstream repository, you've specified for your fork, type git remote -v. You should see the URL for your fork as origin, and the URL for the organization repository as upstream.

git remote -v

origin https://github.com/YOUR\_USERNAME/YOUR\_FORK.git (fetch)

origin https://github.com/YOUR\_USERNAME/YOUR\_FORK.git (push)

upstream https://github.com/ORIGINAL\_OWNER/ORIGINAL\_REPOSITORY.git (fetch)

upstream https://github.com/ORIGINAL\_OWNER/ORIGINAL\_REPOSITORY.git (push)

* Now push your code to your forked repository

git push origin master

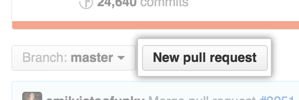
Now, you can keep your fork synced with the upstream repository with a few Git commands. For more information, see "[Syncing a fork](https://help.github.com/articles/syncing-a-fork)."

Step 4: Creating Pull Request

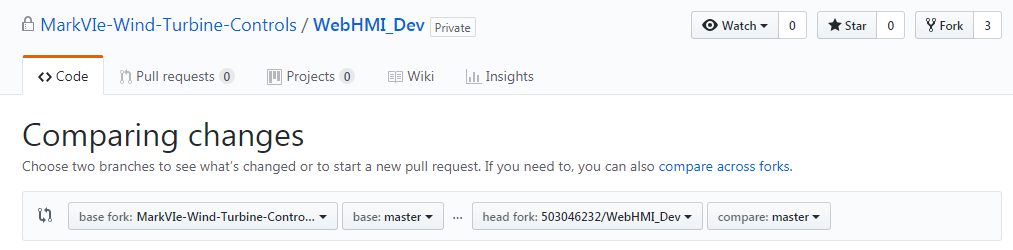
Pull requests let you tell others about changes you've pushed to a repository on GitHub.

Once a pull request is opened, you can discuss and review the potential changes with collaborators and add follow-up commits before the changes are merged into the repository.

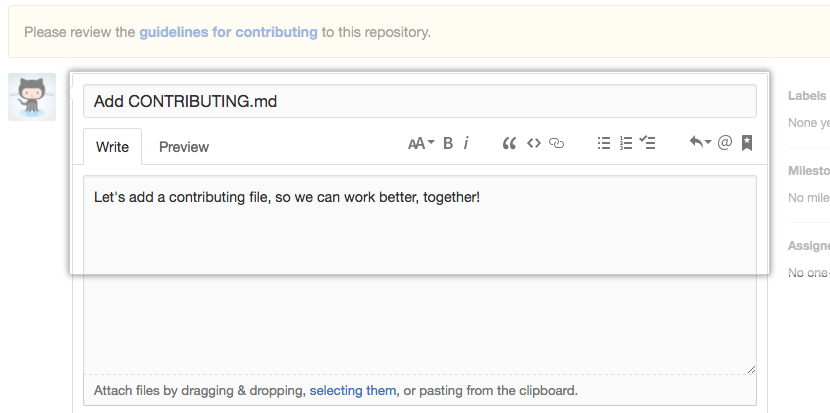
* On GitHub, navigate to the fork page of the repository.
* To the right of the page, click **New pull request**.



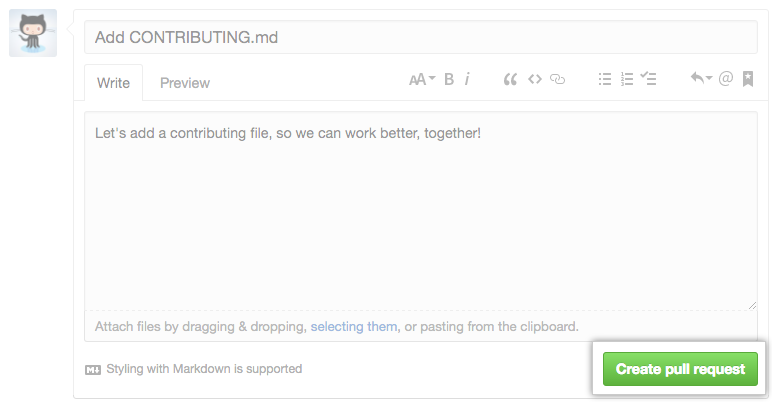
* Use the base branch dropdown menu to select the organization repository branch you'd like to merge your changes into, then use the compare branch drop-down menu to choose the topic fork repository base branch you made your changes in.



* Type a title and description for your pull request.

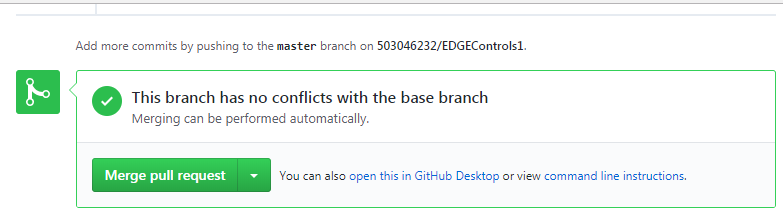


* Click **Create pull request**.



**If the project owner likes your work/code, they might merge your code into the organization repository!**

* As Team Lead/Project owner need to take code review and accept the pull-request.



* Once review Over, **Click Merge pull request Button** to merge you developer code to organization repository.