

Code Version Control with GITHUB and BITBUCKET

<https://github.com/>
<https://bitbucket.org/>

This tutorial will guide you to upload your project to github or bitbucket, and next to download your project to the production server

Step 1: Create an account

First you have to open an account with [github](#) or [bitbucket](#) and create a repository: call it "library"

Step 2: Create a local snapshot of your project

Now we come to some steps that are necessary each time you create a new *repository* (sometimes called a *repo* for short). First navigate to the root directory of the first app and initialize a new repository:

```
$ cd ~/public_html/library  
$ git init
```

The next step is to add all the project files to the repository using **git add -A**:

```
$ git add -A
```

This command adds all the files in the current directory apart from those that match the patterns in a special file called **.gitignore**. The **rails new** command automatically generates a **.gitignore** file appropriate to a Rails project, but you can add additional patterns as well

To tell Git you want to keep the changes, use the **commit** command:

```
$ git commit -m "Initialize repository"
```

Step 3: Upload your project

Now define which repository, github or bitbucket, you are going to use to upload your project.

```
$ git remote add origin https://bitbucket.org/jsilva/library.git  
OR  
$ git remote add origin https://github.com/jsilva/library.git  
$ git push -u origin --all # pushes up the repo and its refs for the first time
```

NOTE: Use only one repository at a time. Change "jsilva" for your personal login in github or bitbucket:

```
Username for 'https://github.com': jsilva  
Password for 'https://jsilva@github.com':
```

That's it! Your project has been uploaded to github (or bitbucket).

Step 4: (optional) working in a project branch

```
$ git checkout -b project_branch_name
```

Do your coding; when you are done merge with the master branch and upload again

```
$ git add -A
$ git commit -m "some info text about code changes"
$ git checkout master
$ git merge project_branch_name
$ git push
```

(Repeat this step as often as you wish, during the development of your project)

Step 5: Download your project to the production server

Login to the production server and go to the web directory (in this example it is public_html)

```
$ cd ~/public_html
```

Now download a copy of your project to a new directory called "library"

```
git clone git://github.com/jsilva/library.git library
```

Step 5: Version control

To learn more about version control follow the tutorials at

- <https://github.com/>
- <https://bitbucket.org/>