projects    **articles**    about

# How to Push a Folder to Github Pages

Published 15 December 2017

Github pages allow you to host your static applications from their CDN. This is a nice and very convenient way to serve documentation, example application, or just some parts of your code. However, it requires you to have a separate branch `gh-pages`, where you need to put `index.html` to the top level (so, no nested directories). Manual approach to this is pretty annoying, and here I would like to share an easy way to upload new versions in one command. There are other ways (for example, using subtree), of course, I just find this one easier, since I have to add only one command for deployment.

## Always create the branch from scratch

> This approach works if you don't need history in your deployment branch – you'll always have only one commit in this branch

In a nutshell, the solution is to create a new branch from scratch every time, commit files as a first commit, and then rewrite remote branch with `--force` flag.

Let's assume you have a `dist` folder, which is generated by building your application, and it is ignored in your `.gitignore` file. Let's assume `npm run build` builds your application, feel free to replace to your command.

> We remove the whole `dist` folder. It is not necessary, but in our case we need at least to remove `dist/.git` folder, so we don't deal with updates

```
  # First, we need to remove old `dist` fold
2 rm -rf dist
3
4 # Build new `dist` folder
5 npm run build
```

Now we need to create a new git repository in this folder and commit all files. This will be the first and the last commit in our deployment branch:

```
   # go to the build directory
2  cd dist
3
4  # create new git repository
5  git init
6
7  # stage everything
8  git add .
9
10 # add our single commit
11 git commit -m "Initial commit"
```

The last step is to add remote to our git repo and push it to needed page. I'd use remote from my equalizer application, and `gh-pages` as a deployment branch:

```
  # add remote origin
2 git remote add origin git@github.com:Bloomca/equalizer.git
3
4 # push to the remote gh-pages branch with force
5 git push --force origin master:gh-pages
```

That's it! Now, the single thing we need to do – unite all of it into one command and add it as an option in our scripts in package.json. So, we need to connect all of them into one command and add it to your list (in my case, it is `package.json`):

```
  rm -rf dist && npm run build && \
2 cd dist && git init && git add . \
3 && git commit -m "Initial commit" && \
4 git remote add origin git@github.com:Bloomca/equalizer.git && \
5 git push --force origin master:gh-pages
```

And now we are done! I can use just `npm run deploy` and the latest version will be on gh-pages almost immediately.

# Related Posts

- 23 Feb 2019 » Alternatives to JSX

- 22 Dec 2018 » Node.js Fundamentals: Web Server Without Dependencies

- 03 Dec 2018 » Small Websites Are Dying

---

Subscribe to get my new articles!

email address

SUBSCRIBE

**Github**　　**Twitter**　　**seva.zaikov@gmail.com**　　**RSS**

© Seva Zaikov, 2021 (source)