



CircleCI

[Improve this page](#)

Building, testing, and deploying your Jekyll-generated website can quickly be done with [CircleCI](#), a continuous integration & delivery tool. CircleCI supports [GitHub](#) and [Bitbucket](#), and you can get started for free using an open-source or private repository.

Follow Your Project on CircleCI

To start building your project on CircleCI, all you need to do is 'follow' your project from CircleCI's website:

1. Visit the 'Add Projects' page: <https://circleci.com/add-projects>
2. From the GitHub or Bitbucket tab on the left, choose a user or organization.
3. Find your project in the list and click 'Build project' on the right.
4. The first build will start on its own. You can start telling CircleCI how to build your project by creating a [circle.yml](#) file in the root of your repository.

Dependencies

The easiest way to manage dependencies for a Jekyll project (with or without CircleCI) is via a [Gemfile](#). You'd want to have Jekyll, any Jekyll plugins, [HTML Proofer](#), and any other gems that you are using in the `Gemfile`. Don't forget to version `Gemfile.lock` as well. Here's an example `Gemfile`:

```
source 'https://rubygems.org'

ruby '2.4.0'

gem 'jekyll'
gem 'html-proofer'
```

CircleCI detects when `Gemfile` is present is will automatically run `bundle install` for you in the `dependencies` phase.

Testing

The most basic test that can be run is simply seeing if `jekyll build` actually works. This is a blocker, a dependency if you will, for other tests you might run on the generate site. So we'll run Jekyll, via Bundler, in the `dependencies` phase.

```
dependencies:
  post:
    - bundle exec jekyll build
```

HTML Proofer

With your site built, it's useful to run tests to check for valid HTML, broken links, etc. There's a few tools out there but [HTML Proofer](#) is popular amongst Jekyll users. We'll run it in the `test` phase with a few preferred flags. Check out the `html-proofer` [README](#) for all available flags, or run `htmlproofer --help` locally.

```
test:
  post:
    - bundle exec htmlproofer ./_site --check-html --disable-externa
```

Complete Example circle.yml File

When you put it all together, here's an example of what that `circle.yml` file could look like:

```
machine:
  environment:
    NOKOGIRI_USE_SYSTEM_LIBRARIES: true # speeds up installation of

dependencies:
  post:
    - bundle exec jekyll build

test:
  post:
    - bundle exec htmlproofer ./_site --allow-hash-href --check-fav:

deployment:
  prod:
    branch: master
    commands:
      - rsync -va --delete ./_site username@my-website:/var/html
```

Questions?

This entire guide is open-source. Go ahead and [edit it](#) if you have a fix or [ask for help](#) if you run into trouble and need some help. CircleCI also has an [online community](#) for help.

GETTING STARTED

- Welcome
- Quick-start guide
- Installation
- Basic Usage
- Directory structure
- Configuration

YOUR CONTENT

- Front Matter
- Writing posts
- Working with drafts
- Creating pages
- Static Files
- Variables
- Collections
- Data Files
- Assets
- Blog migrations

CUSTOMIZATION

- Templates
- Includes
- Permalinks
- Pagination
- Plugins
- Themes
- Extras

DEPLOYMENT

- GitHub Pages
- Deployment methods
- Continuous Integration

MISCELLANEOUS

- Troubleshooting
- Sites using Jekyll
- Resources
- Upgrading from 0.x to 2.x
- Upgrading from 2.x to 3.x

META

- Contributing
- Maintaining Jekyll
- Code of Conduct
- History