Courses > 310 Build Robust and Production Quality Applications > Week 5 > Assignment: Set up Continuous Delivery with Circle CI

Assignment: Set up Continuous Delivery with Circle CI

For this assignment, you are going to enable Continuous Delivery with Circle CI.

Here are the steps:

- create a `staging` branch locally if you don't have it yet.
- in Circle CI, follow `Project settings` for your project and select `Heroku Deployment` under Continuous Deployment
- configure the Heroku API key. (Get it from your Heroku account page)
- associate Heroku SSH key with your Circle account so Circle can have the authority to deploy to Heroku on your behalf.
- create a `circle.yml` file in your projects root directory and make sure your adjust `production_app_name` & `staging_app_name` to your own app name.

Here is an example of a `circle.yml` file that you can use:

```yaml
machine:
  ruby:
    version: 2.1.5
deployment:
  production:
    branch: master
    commands:
      - heroku maintenance:on --app production_app_name
      - heroku pg:backups capture --app production_app_name
      - git push git@heroku.com:production_app_name.git $CIRCLE_SHA1:refs/heads/master
      - heroku run rake db:migrate --app production_app_name
      - heroku maintenance:off --app production_app_name
  staging:
    branch: staging
    commands:
      - heroku maintenance:on --app staging_app_name
      - git push git@heroku.com:staging_app_name.git $CIRCLE_SHA1:refs/heads/master
      - heroku run rake db:migrate --app staging_app_name
      - heroku maintenance:off --app staging_app_name
```

Note: you should change the Ruby version to the version you're using for this project.

This code should be pretty self explanatory - this allows Circle to monitor your `staging` branch and deploy to the staging server, and monitor your `master` branch to deploy to the production server. It'll run migrations for you and for the production server, it also automatically backs up the database before a deploy.

Notifications Options

By default we will be notified by email on every status change of our builds. But receiving an email every time a build fails or passes is cumbersome and there are also good alternatives.

For Mac users - there is a free application CCMenu by ThoughtWorks which we install and it will stick to our upper bar. In order to make it fully run - we will need to generate a CircleCI API key (Project settings -> API keys) and http address of build. See the setup info guide here. A great feature of this app is that it can play sounds based on build's status and we are able to customize it to our needs.

For Chrome users - there is an addon called CircleCI Monitor that takes a couple of seconds to install and works out of the box.

Another great feature most CIs offer are Embedded Status Badges. These are especially useful for open source projects. They look nice in a project's README and also help to provide information about whether the current build is passing or failing.

You marked this topic or exercise as completed.

< Continuous Delivery

Attend local meetups; Start the process of looking for jobs >