



Courses > 310 Build Robust and Production Quality Applications > Week 5 > Understand staging server and production server

Understand staging server and production server

This is a good tutorial on why you need staging server environment and how to set it up with Heroku - focus on the explanations, and don't follow the steps there (it's a bit vague and can be confusing for the apps names that they refer to), we'll take care of this later.

<https://devcenter.heroku.com/articles/multiple-environments>

A few things I'd like to add:

1) Make staging server's configuration and data as close to the production server as possible - after all, the purpose of having the staging server is to be able to test out new features in a "production like" environment.

If your production app is already launched, you can periodically take database dumps from the production server to populate the staging server. You could automate this as well with either a replication service, or if you are on Heroku, use Heroku's follower DB (comes with a cost) <https://devcenter.heroku.com/articles/heroku-postgres-follower-databases>

2) You want to be mindful on email sending on staging - staging server will actually send out emails. You want to make sure that you do not spam your users while testing out features on the staging server. This can be achieved by adding a conditional in your mailer to see if the environment is staging, and if it is, set email recipients to administrators (such as yourself). This way, you can still test and verify that emails are sent out, but you don't spam users.

3) Your staging server can (and typically should) have a different config itself - you just need to create a `staging.rb` file in `config/environments` directory, and with it you can specify the exact settings you want on there (for example, turning on asset pipeline or not, sending out email or not, charging credit card for real or not, etc). On your Heroku staging server, make sure to set `RACK_ENV` and `RAILS_ENV` to `staging`, then it'll use that config instead. For Heroku specifically, you also need to add the `rails_12factor` gem to the staging environment for the asset pipeline to work on Heroku.

You marked this topic or exercise as completed.