[Lesson 7.4: Deploying with Heroku](#h.v50txl8329sa)

[Gem of the Day](#h.wolxsv5ssfdc)

[Objectives](#h.se1hbkpokrwc)

[What is Heroku and how does it fit into our app architecture?](#h.sqr1k5yn7y83)

[Our First Heroku App](#h.axpqmd3s3h3y)

[More Heroku Commands](#h.ukj4u04azj24)

[How to Delete a Heroku App](#h.w9j7ohtn6cjo)

[Instructions for Deploying Doggy Daycare](#h.t5hg1ztf8wbw)

[Dropbox Setup](#h.vq8uuco7lqqr)

[Heroku Deployment](#h.5p8xlg2liosw)

[Homework](#h.aj2zbata8t07)

# 

# Lesson 7.4: Deploying with Heroku

Share out any cool features students added to their Google Maps for Bus Me.

## Gem of the Day

Dashing - cool dashboards! <https://github.com/Shopify/dashing>

## Objectives

* Understand how Heroku and Git are connected
* Understand all steps to successfully push an app to Heroku
* Know how to delete an app on Heroku (only 5 free)

## What is Heroku and how does it fit into our app architecture?

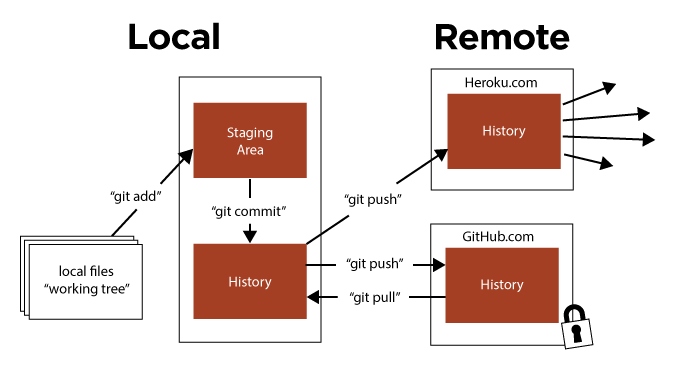
Heroku is a scalable cloud platform service for hosting your web applications.

* **Easy Deployment** - You can instantly deploy your app from the command line using a git push command.
* **Isolation** - Heroku uses “containers” called dynos that allow multiple isolated operating systems/processes to run on a shared host. A process in one container cannot see anything about another process in another container.
* **Scalability** - You can scale your app to handle more concurrent requests by adding more dynos or to handle more information at a faster speed (more RAM/CPU) with bigger dynos.
* **Full Logging and Visibility** - easy access to all logging output from every component of your app and each process (dyno)

Competitors to Heroku include:

* Amazon AWS
* Google App Engine (no Rails though)
* DigitalOcean (cheaper but more work)
* Linode (mid-range on price)
* and many others!

**How does Heroku fit in?**  Draw this on board:



## 

## Our First Heroku App

Let’s push Bus Me to Heroku.. The app must have it’s own git repository - not combined with any other app.

What files do we probably need to change if we want to change the production db? (Answer: gemfile, database.yml)

Steps :

1. Change **gem file** to use postgres (gem 'pg') in production and gem ‘sqlite’ only in dev and test. NOTE that Heroku doesn’t like using different servers. Also need to add to production gem 'rails\_12factor'. bundle install. Commit.
   1. If you get a bundle error related to pg, run this bundle command which says to ignore production gems:  
      bundle install --without production
2. Update **config/database.yml** file to replace everything in production with:   
   production:  
    <<: \*default  
    adapter: postgresql
3. Commit.
4. Once your app is ready to be released to production (db’s are migrated, everything is working, you’ve committed all changes) continue with heroku login
5. heroku create (and note your randomly generated app name and website)
6. git push heroku master (show diagram again)
   1. Purposefully attempt heroku open now. Ask how many got “can’t find page” (rake routes) vs “Something went wrong” (no db migrate)
   2. Get everyone to where something went wrong - ask for guesses as to what went wrong
7. If you have a database, you will need to migrate your db on heroku with heroku run rake db:migrate
8. Set any environment variables you may have created using [Figaro](https://github.com/laserlemon/figaro#deployment):  
   figaro heroku:set -e production
9. (optional) Ensure that at least one instance of the app is running: heroku ps:scale web=1
10. Open (or refresh) your app with the nifty heroku open (opens your live app in the browser!)

### More Heroku Commands

heroku logs -t

heroku config:add TZ=America/Chicago

Each time you make changes to your project, commit them as usual. Then, when you’re ready to release them to production on Heroku, run git push heroku master.

If you want to rename your app, it’s easiest to do from the command line. See [these instructions](https://devcenter.heroku.com/articles/renaming-apps).

This is a great tutorial - use it with their files, or use it as a general guide with your own app:

<https://devcenter.heroku.com/articles/getting-started-with-ruby>

## How to Delete a Heroku App

Go to heroku.com and log in. You will see your dashboard. Click on the app you no longer want. Click on Settings, scroll down, and then click on Delete app and continue with the instructions provided to confirm the delete.

## Instructions for Deploying Doggy Daycare

We used Paperclip for Doggy Daycare, which means we need to set up a service to host our images. You can use Amazon S3, Dropbox, or some other solutions. These instructions will cover using Dropbox. You can learn more about both options in the Paperclip readme, or specifically for [Dropbox](https://github.com/janko-m/paperclip-dropbox) and [Amazon S3](http://www.rubydoc.info/gems/paperclip/Paperclip/Storage/S3).

### Dropbox Setup

Before we can use Dropbox to host our photos, we need to set up a Dropbox Platform App. First, decide whether you want to use Full Dropbox access (public folder) or a special App Folder. Read about the differences [here](https://github.com/janko-m/paperclip-dropbox/wiki/Access-types). The rest of these instructions are for doing the App Folder.

1. Go to <https://www.dropbox.com/developers/apps/create>
2. Select Dropbox API App, select ‘yes’, give it a name, and submit.
3. Add gems to gemfile, and bundle (or bundle install --without production):

gem "paperclip-dropbox", ">= 1.1.7"

gem 'figaro'

group :production do

gem 'pg'

gem 'rails\_12factor'

end

group :development, :test do

gem 'sqlite3'

...

1. Create a new file called dropbox.yml in your config folder, and inside it, put the below content but replace app\_key and app\_secret (as well as make sure app\_folder is the access type):

app\_key: "..."  
app\_secret: "..."  
access\_token: "..."  
access\_token\_secret: "..."  
user\_id: "..."  
access\_type: "app\_folder"

1. Now we need to do a rake task to save our app key and secret which will authorize our app and provide the other 3 missing fields for the above file. Type this command in command line, but replace with your app key and secret, not in quotes:

rake dropbox:authorize APP\_KEY=your\_app\_key APP\_SECRET=your\_app\_secret ACCESS\_TYPE=app\_folder

1. Once you hit enter, it will tell you to go to a specific url to authorize your app. Do this, then hit “y”. It will spit back your access token, secret, and user\_id. Fill out the rest of **config/dropbox.yml** with that info.
2. Now, run figaro install to get your application.yml file. Copy everything in dropbox.yml over to your application.yml. Then, make your dropbox.yml file look like this - this will make it easier to push to production using Figaro’s capabilities:

app\_key: <%= ENV['app\_key'] %>

app\_secret: <%= ENV['app\_secret'] %>

access\_token: <%= ENV['access\_token'] %>

access\_token\_secret: <%= ENV['access\_token\_secret'] %>

user\_id: <%= ENV['user\_id'] %>

access\_type: <%= ENV['access\_type'] %>

1. Once your secrets are no longer in dropbox.yml, you are free to commit your changes!
2. Now, in both our **dog** and **products** **models**, we need to append some code to our **has\_attached\_file** validation (put a comma after the default file location item, hit enter, then paste this):

:storage => :dropbox,  
 :dropbox\_credentials => Rails.root.join("config/dropbox.yml")

1. It’s probably safest to rake db:drop and then both re-migrate and seed. You’re welcome to attempt skipping this step though in production you will kind of have to do this anyway since the database itself doesn’t port to production.
2. Run your server to test that you can add a photo to a dog or product. So cool. Now commit your changes.

### Heroku Deployment

1. Update **config/database.yml** file to replace everything in production with:   
   production:  
    <<: \*default  
    adapter: postgresql
2. Commit.
3. Once your app is ready to be released to production (db’s are migrated, everything is working, you’ve committed all changes) continue with heroku login
4. heroku create (and note your randomly generated app name and website)
5. git push heroku master
6. Migrate your db on heroku with heroku run rake db:migrate, then seed it with heroku run rake db:seed.
7. Set any environment variables you may have created using [Figaro](https://github.com/laserlemon/figaro#deployment):   
   figaro heroku:set -e production
8. Open (or refresh) your app with the nifty heroku open (opens your live app in the browser!)

## Homework

Try to deploy another one of your apps - especially a special project that you’ve been working on. Clean up your apps and add them to your portfolio page!