**Heroku Deployment**

1. **Install GIT**
2. **If nodejs is installed, run the command:**

**Npm install –g heroku-cli**

1. **Ready the project with GIT repository**
2. **Run the command:**

heroku create

1. **Run the command:**

git push heroku master

1. **The application is now deployed. Ensure that at least one instance of the app is running:**

heroku ps:scale web=1

1. **Now visit the app at the URL generated by its app name. As a handy shortcut, you can open the website as follows:**

heroku open

1. **View information about your running app using one of the**[**logging commands**](https://devcenter.heroku.com/articles/logging)**, heroku logs --tail:**

heroku logs --tail

1. **Defifning a** [**Procfile**](https://devcenter.heroku.com/articles/procfile)
2. **Right now, your app is running on a single web**[**dyno**](https://devcenter.heroku.com/articles/dynos)**. Think of a dyno as a lightweight container that runs the command specified in the Procfile.**

**You can check how many dynos are running using the ps command:**

heroku ps

1. **Scaling an application on Heroku is equivalent to changing the number of dynos that are running. Scale the number of web dynos to zero:**

heroku ps:scale web=0 //change value to scale dyno

1. **Declaring App dependencies:**

composer update

1. **Push Local Changes:**

$ composer require alrik11es/cowsayphp

$ composer update

First, add the modified files to the local git repository:

git add .

git commit -m "Demo"

Now deploy, just as you did previously:

git push heroku master

Finally, check that everything is working:

heroku open cowsay

1. **In this step you will provision one of these logging add-ons, Papertrail.**

heroku addons:create papertrail

1. **The add-on is now deployed and configured for your application. You can list add-ons for your app like so:**

heroku addons

1. **To see this particular add-on in action, visit your application’s Heroku URL a few times. Each visit will generate more log messages, which should now get routed to the papertrail add-on. Visit the papertrail console to see the log messages:**

heroku addons:open papertrail

1. **You can run a command, typically scripts and applications that are part of your app, in a**[**one-off dyno**](https://devcenter.heroku.com/articles/one-off-dynos)**using the heroku run command. It can also be used to launch an interactive PHP shell attached to your local terminal for experimenting in your app’s environment:**

heroku run "php -a" //type <quit> to stop

1. **To get a real feel for how dynos work, you can create another one-off dyno and run the bashcommand, which opens up a shell on that dyno. You can then execute commands there.**

heroku run bash

1. **Defrining Config Vars**

heroku config:set TIMES=20

1. **View the config vars that are set using heroku config:**

heroku config

1. **Add the database:**

heroku addons:create heroku-postgresql:hobby-dev

**Deploy Laravel App with Heroku**

1. **Create a laravel app , it’s repository, and create heroku account.**
2. **App must be in GitHub**
3. **Create a procfile:**

echo web: vendor/bin/heroku-php-apache2 public/ > Procfile

git add .

git commit -m "Procfile for Heroku"

1. **Create new Application**

heroku create

1. **Setting a Laravel Encription key:**

heroku config:set APP\_KEY=…

1. **Pushing to heroku**

git push heroku master

1. **Open the web app:**

heroku open

1. **Best Practices:**

In order to ensure that errors, should they arise, are visible in heroku logs, the corresponding configuration in config/app.php must be changed:

'log' => env('APP\_LOG', 'errorlog'),

Do not forget to add, commit, and, if appropriate, push this change: