Heroku Flow

**Create Pipeline For Git Repository**

1. Login to Heroku
2. In top left, click on “**Personal apps**”
   1. Select “**grizzlycoastmedia**” (note this may already be selected)
3. Create Production App and Add to Pipeline
   1. Click on “**Apps**”
   2. In top right, click “**New**”
      1. Select “**Create new app**”
   3. Type in app name (optional, but easier to find app afterward).
      1. Example: “lumavieskincare” for “lumavieskincare.com”
   4. Click “**Create App**”
   5. Click on “**New Pipeline**”
   6. Click on “**Create Pipeline**” (name if needed)
      1. Example: “lumavieskincare” for “lumavieskincare.com”
4. Create Stage App and Add to Pipeline
   1. Click on “**grizzlycoastmedia**” in top left
   2. Click on “**Apps**”
   3. In top right, click “**New**”
      1. Select “**Create new app**”
   4. Type in app name (optional, but easier to find app afterward).
      1. Example: “stage-lumavieskincare” for “lumavieskincare.com”
   5. Click “**Create App**”
   6. Click on “**grizzlycoastmedia**” in top left
   7. Click on “**Apps**”
   8. Click on production app
      1. In this case it would be “lumavieskincare”
   9. Click on the pipeline for this app (multiple app icon next to “grizzlycoastmedia” in top left
   10. Click “**Add app**” next to “staging”
   11. Type in stage app name and select it
       1. In this case it would be “stage-lumavieskincare”
5. Connect to GitHub:
   1. From the pipeline for these apps, select the production app
      1. In this case “lumavieskincare”
   2. Click “**Deploy**”
   3. Beside “Deployment method”, click “GitHub Connect to GitHub”
   4. Beside “Connect to GitHub”, select “**grizzlycoastmedia**” from the dropdown
   5. Type in repository name and click “**Search**”
   6. Beside the correct repository, click “**Connect**”
   7. Beside “Automatic deploys”, click “**Enable Automatic Deploys**”
6. Add Review to Pipeline
   1. From the pipeline for these apps, there should now be a “Review Apps” section.
   2. Click “**Enable Review Apps…”**
   3. Click “**Create an app.json File…”**
   4. Scroll down new page and click “**Commit to Repo**”
   5. Click on “**Create new review apps for new pull requests automatically**” checkbox
   6. Click “**Enable**”
7. Tell Heroku how to use this app
   1. For php, add an composer.json file with only these braces “{}” in the file to repository root directory
   2. For php, if there isn’t an index file in root, add Procfile to repository root directory telling it where the entry point is
      1. web: vendor/bin/heroku-php-apache2 public\_html/

**Adding Work to Existing Pipeline**

1. Create Git branch off master:

**git checkout -b {yourBranchName}**

1. Create remote git branch for pushing:

**git push –set-upstream origin {yourBranchName}**

1. Create Heroku app for branch:

**heroku create**

1. Push code to Heroku app:

**git push heroku {yourBranchName}:master**

1. You can now view the page:

**heroku open**

1. Make changes to branch and commit/push to git for version control, as well as heroku app to view changes.
2. Task is ready for dev QA using the link from “heroku open” command
3. When the task is finished QA, go to github and select the repository
4. Click on **{yourBranchName}**
5. Click on “**Compare & pull request**”
6. Check changes are as expected and click “**Create pull request**”
7. Once review/testing is finished, move app to either stage or production by clicking the dropdown on the app and selecting “**Move app to** **{environment}”**