All development can be done by cloning the github repo onto your device and editing it normally.
A normal workflow I use is:

1. Clone github repo
2. Checkout the branch that I want to use
    a. You can do this on the command line or with github desktop
3. Make changes
4. Make git commits as you go
    a. I use VS code's helpful committing, can also use the CLI
5. When you have made all the commits you want to test, push your changes to github

Once you're ready to test:

Step 1:

Get Greyson to open up the EC2 instance. This can be opened for 4 hours at a time, just let me know and I can get it going for you most times during the day.

Step 2:

Once you have the IP address from Greyson, use this command to ssh into the server:

```
sudo ssh -i [path of augur instance.pem] ubuntu@[ip]
```

Make sure to replace with the path to your instance and the ip

The password it asks for is the admin password to YOUR device

Step 3:

Move into your folder via cd

Move into your augur clone via cd

Step 4:

Make sure you have the correct git branch checked out via:

```
git rev-parse --abbrev-ref HEAD
```

If you do not have the correct branch checked out, run:

```
git checkout [branch-name]
```

Step 5:

Run "git pull" to pull your commits

Step 6:

Get into the python virtual environment via:

```
source $HOME/.virtualenvs/augur_env/bin/activate
```

Step 7:

Setup augur, you can ctrl-c once it asks you about databases via:

```
make dev-install
```

Step 8:

Setup the docs via:

```
make docs
```

Step 9:

On your PERSONAL computer, run this command to pull down the newly created html files:

```
sudo scp -r -i [path to augur key] ubuntu@[server ip]:/home/ubuntu/[your folder name]/augur/docs/build/html [source dest (for current folder, just ./)]
```

Make sure to replace the path to the key, the server IP, your folder name, and the destination where you want the files to go in your personal computer

Step 10:

Open one of the html files in the folder you downloaded. Then you should be able to navigate to your changes and see them

Tom's suggestion: make versions of all the commands that put things where you want them in your laptop, then you only have to change the IP every time