- Make a Mongo Atlas account and get a free Cluster
- Free is on the left. The provider should not matter too much unless we decide to head for certain implementations later.
- Create your cluster. It will take a few minutes to boot up.
- While this happens, go to Database Access on the left and create a user. This is the user and password that will need to be in your code (<DatabaseUser> and <password>). I used the autogenerated password, I wouldn't use one of your personal ones for obvious reasons. I also used the password authentication method and gave the user read and write privileges. Take note of the user and the password as you will need it in the code.
- Next, go to network access and add an IP. Click Allow Access From Anywhere, which will be 0.0.0.0/0. If you do not do this, the Heroku app will not be able to read or write.
- Go back to Clusters, and once the cluster is done creating go to Collections. It will ask you to load a sample dataset or add your own. Click Add My Own Data. Add and take note of your database name and give your collection a name. The name of this database will be called <databaseName> in the code.
- Go back to Clusters and click connect. Click Connect using MongoDB Compass. There will be a string that you can copy. This will be the string you add to the Database host in the code. It will look like this:
    - mongodb+srv://<user>:<password>@cluster0.hpurh.mongodb.net/test . There are a few changes that need to be made that look like the next line
    - mongodb+srv://<user>:<password>@ cluster0.hpurh.mongodb.net /<DatabaseName>?retryWrites=true&w=majority
    - Copy this string and replace user, password, the cluster info after the @ with your cluster, and DatabaseName to your information. This whole string will be called <Link> in the code
- In your code, in settings.py, in Database, replace that whole field with this, replacing the bracketed <items> with your respective information in single quotes

```
DATABASES = {
    'default': {
        'ENGINE': 'djongo',
        "CLIENT": {
           "name": <databaseName>,
           "host": <Link>
           "username": <DatabaseUser>,
           "password": <password>,
           "authMechanism": "SCRAM-SHA-1",
        },
    }
}
```

- Add the link to your Heroku App in the ALLOWED_HOSTS array in settings.py in single quotes. Make sure the Django_heroku.settings(locals()) line near the bottom is uncommented
- In the Settings for your Heroku deployment at https://dashboard.heroku.com/apps , go to resources and delete the one addon, then go to settings and delete the item in Config Vars if it is still there. This was the old postgres database Heroku uses by default.

- Make sure you install all of the dependencies in the Pipfile
- Run py manage.py migrate
  - This will set up all of the collections needed in your database. You can check to see if this works by seeing if your database has collections like accounts_user and accounts_user_groups
- You should now be able to run this locally with “py manage.py runserver” or on your Heroku app by pushing it to your Heroku repo