

# Deployment to Heroku

---



Node.js



Heroku Dev Center



MOVE FAST

## Unleash your inner startup

Choose Heroku for the same reasons disruptive startups do: it's the best platform for building with modern architectures, innovating quickly, and scaling precisely to meet demand.

[SIGN UP FOR FREE](#)

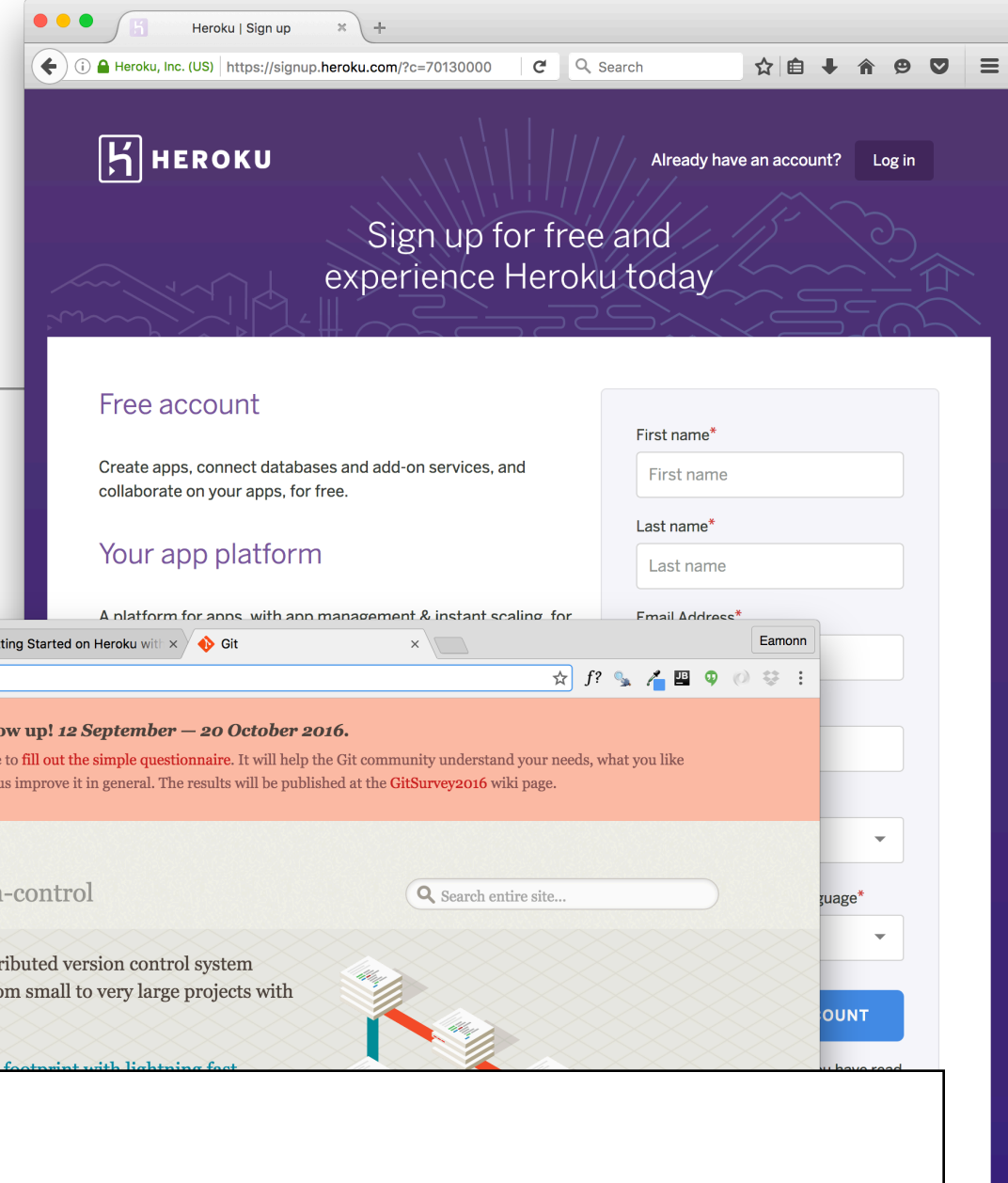
[Explore Heroku Customers](#)



- Platform as a Service (PaaS).
- Offers simplified node.js application deployment & management

# Requirements

- Sign up for a free account on heroku
- Install Git
- Install Heroku Command Line



## Set up

In this step you will install the Heroku Command Line Interface (CLI), formerly known as the Heroku Toolbelt. You will use the CLI to manage and scale your applications, to provision add-ons, to view the logs of your application as it runs on Heroku, as well as to help run your application locally.

 Download the Heroku CLI for...

Once installed, you can use the `heroku` command from your command shell.

Log in using the email address and password you used when creating your Heroku

# Requirements

- Sign up for free mLab account

## Trusted. Loved. Most widely deployed.

mLab's Database-as-a-Service proudly powers over **350,000** MongoDB deployments on AWS, Azure, and Google



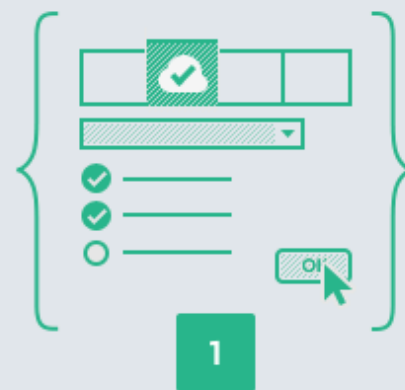
Thousands of companies trust mLab with their data



CONDÉ NAST



### MongoDB in your choice of cloud. It's this easy.



Create a database

Provision MongoDB on-demand on AWS, Azure, or Google.



Paste its connection URI

Copy and paste the connection string into your code.



Build the future

Focus on your product instead of operations.

# Deployment: 7 Steps

---

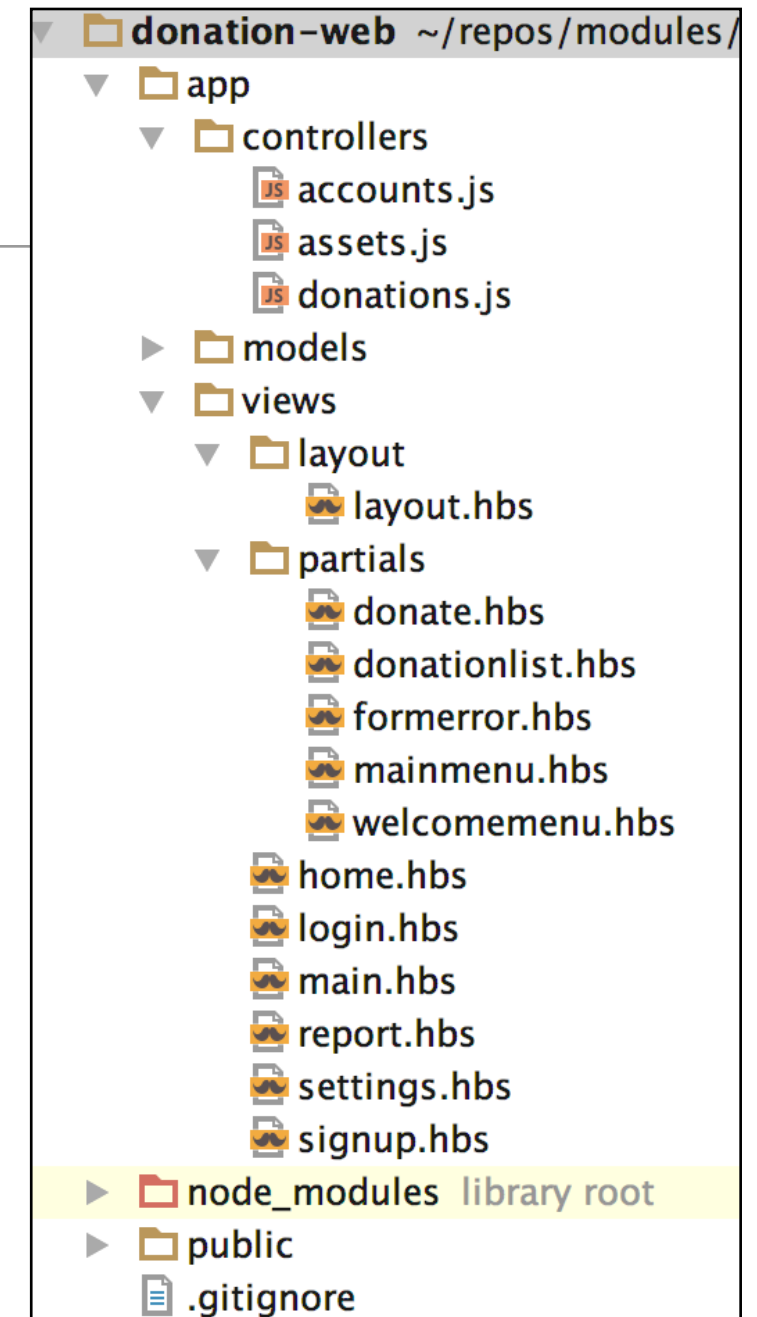
1. Commit project to git
2. Create an application using the heroku command line
3. Provision a MongoDB database
4. Prepare **package.json** for deployment
5. Push the application to heroku remote
6. Enable 'Production' mode, db connection, & restart
7. Monitor the Heroku Logs

# 1: Commit project to git

```
git add .
```

```
git commit -m "first commit"
```

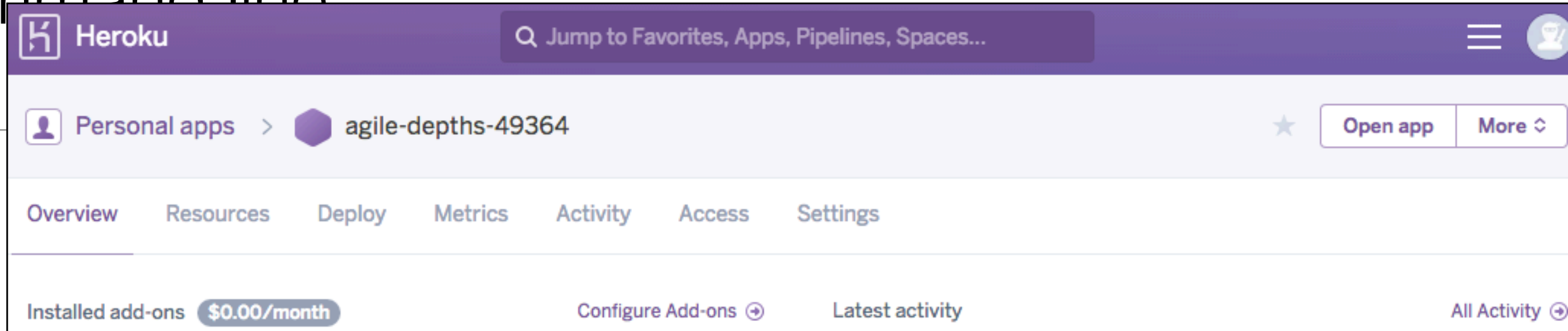
- Create gitignore file
- add all files
- commit all files



**.gitignore**

```
.idea  
node_modules
```

# 2: Create an application using the heroku command line



Now log in to your heroku account:

```
heroku login
Email:
Password (typing will be hidden):
```

Once logged in, create a new application on heroku

```
heroku create
```

This will respond with a new name in a few seconds:

```
Creating app... calm-brushlands-29225
https://calm-brushlands-29225.herokuapp.com/ | https://git.heroku.com/calm-brushlands-29225.git
```



# 3: Provision a MongoDB database

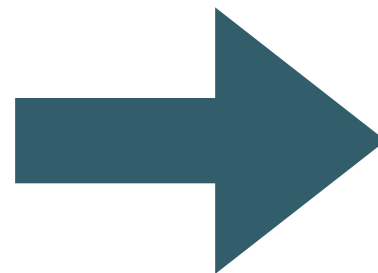
Because we are using a mongo database, we need to provision this feature:

```
heroku addons:create mongolab
```

This should respond as follows:

```
Creating mongo
Adding mongolab-silhouetted-58540 to a
Setting MONGODB_URI and restarting agi
Welcome to mLab. Your new subscription
Use `heroku addons:docs mongolab` to v
```

Require Credit Card Registration



Heroku

Personal apps > agile-depths-49364

Overview Resources Deploy Metrics Activity Acc

Installed add-ons \$0.00/month Configure Add-on

mLab MongoDB Sandbox

Dyno formation \$0.00/month Configure Dyno

This app has no process types yet  
Add a Procfile to your app in order to define its process types. [Learn more](#)



# 3: Create Mongo Database on MongoLab

- Create Database + Create Special User for database
- Extract 'Connection String'

Home

Database: donation Delete database

To connect using the mongo shell:

```
% mongo ds055626.mlab.com:55626/donation -u <dbuser> -p <dbpassword>
```

To connect using a driver via the standard MongoDB URI ([what's this?](#)):

```
mongodb://<dbuser>:<dbpassword>@ds055626.mlab.com:55626/donation
```

mongod version: 3.2.10 (MMAPv1)

⚠ Sandbox databases do not have redundancy and therefore are not suitable for production. Visit our [guide to running in production](#) for more info.

**Collections** Users Stats Backups Tools Add collection

*[None at this time]*

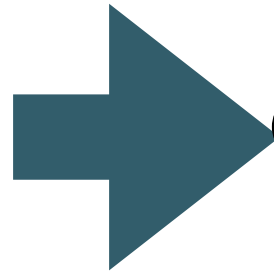
**System Collections**

NAME	DOCUMENTS	SIZE
system.indexes	0	0.00 KB

## 4: Prepare package.json for deployment

---

- define 'start' script
- then commit this change:



```
{
  "name": "donation-web",
  "version": "1.0.0",
  "description": "an application to host donations for",
  "main": "index.js",
  "scripts": {
    "start": "node index",
    "test": "echo \"Error: no test specified\" && exit 1"
  },
  "author": "",
  "license": "ISC",
  "dependencies": {
    "handlebars": "^4.0.5",
    "hapi": "^14.1.0",
    "hapi-auth-cookie": "^6.1.1",
    "inert": "^4.0.1",
    "joi": "^9.0.4",
    "mongoose": "^4.5.8",
    "vision": "^4.1.0"
  }
}
```

```
git commit -m "node launch script added"
```

# 5: Push the application to heroku remote

---

```
git push heroku master
```

```
heroku open
```

```
Counting objects: 43, done.
Delta compression using up to 8 threads.
Compressing objects: 100% (39/39), done.
Writing objects: 100% (43/43), 506.04 KiB | 0 bytes/s, done.
Total 43 (delta 6), reused 0 (delta 0)
remote: Compressing source files... done.
remote: Building source:
remote:
remote: -----> Node.js app detected
remote:
remote: -----> Creating runtime environment
remote:
remote:           NPM_CONFIG_LOGLEVEL=error
remote:           NPM_CONFIG_PRODUCTION=true
remote:           NODE_ENV=production
remote:           NODE_MODULES_CACHE=true
```

```
remote:           Procfile declares types      -> (none)
remote:           Default types for buildpack -> web
remote:
remote: -----> Compressing...
remote:           Done: 15.3M
remote: -----> Launching...
remote:           Released v5
remote:           https://agile-depths-49364.herokuapp.com/ deployed to Heroku
remote:
remote: Verifying deploy.... done.
To https://git.heroku.com/agile-depths-49364.git
 * [new branch]      master -> master
```

- Use git to transmit app sources to heroku
- Browse to the deployed url

## 6: Enable 'Production' mode

---

- Although the app will launch, we will not be able to get beyond the sign up screen. Trying to register a new user will generate an internal error on the app.

```
9.449373+00:00 heroku[web.1]: Starting process with command `npm start`
1.420126+00:00 app[web.1]:
1.420148+00:00 app[web.1]: > donation-web@1.0.0 start /app
app[web.1]: > node index
app[web.1]:
app[web.1]: Server listening at: http://1bee6dd6-e0a2-4b2a-81fe-9c6e9485bea3:48
2.420000+00:00 app[web.1]: Mongoose disconnected
2.427543+00:00 app[web.1]: Mongoose connection error: MongoError: getaddrinfo ENOTFOUND undefi
```

```
heroku logs --tail
```

```
var dbURI = 'mongodb://localhost/donation';
if (process.env.NODE_ENV === 'production') {
  dbURI = process.env.MONGODB_URI;
}
```

- Need to reset the app to production mode, and restart

```
heroku config:set NODE_ENV="production"
```

## 6: Set db connection string & restart

---

- On Command Line, set MONGOLAB\_URI to connection string harvested from mLab

```
heroku config:set MONGOLAB_URI=mongodb://donationuser:donationuser@dsXXX.mlab.com:XXXX/donation
```

```
var dbURI = 'mongodb://localhost/donation';  
if (process.env.NODE_ENV === 'production') {  
  dbURI = process.env.MONGOLAB_URI;  
}
```

- Restart the app

```
heroku restart
```

# 7: Monitor the Heroku Logs

---

```
heroku logs --tail
```

```
2016-08-07T11:45:49.449373+00:00 heroku[web.1]: Starting process with command `npm start`  
2016-08-07T11:45:51.420126+00:00 app[web.1]:  
2016-08-07T11:45:51.420148+00:00 app[web.1]: > donation-web@1.0.0 start /app  
2016-08-07T11:45:51.420149+00:00 app[web.1]: > node index  
2016-08-07T11:45:51.420150+00:00 app[web.1]:  
2016-08-07T11:45:52.419684+00:00 app[web.1]: Server listening at: http://1bee6dd6-e0a2-4b2a-81fe-9c6e9485
```

# Connect Robomongo to mLab

---

- The database we are now using is in the cloud - and it might be useful to be able to browse directly to it.

- For this url

```
mongodb://heroku_4pt2zvkj:omev5e4sctvbiala0i1t5cbstdj@ds011902.mlab.com:11902/heroku_4pt2
```

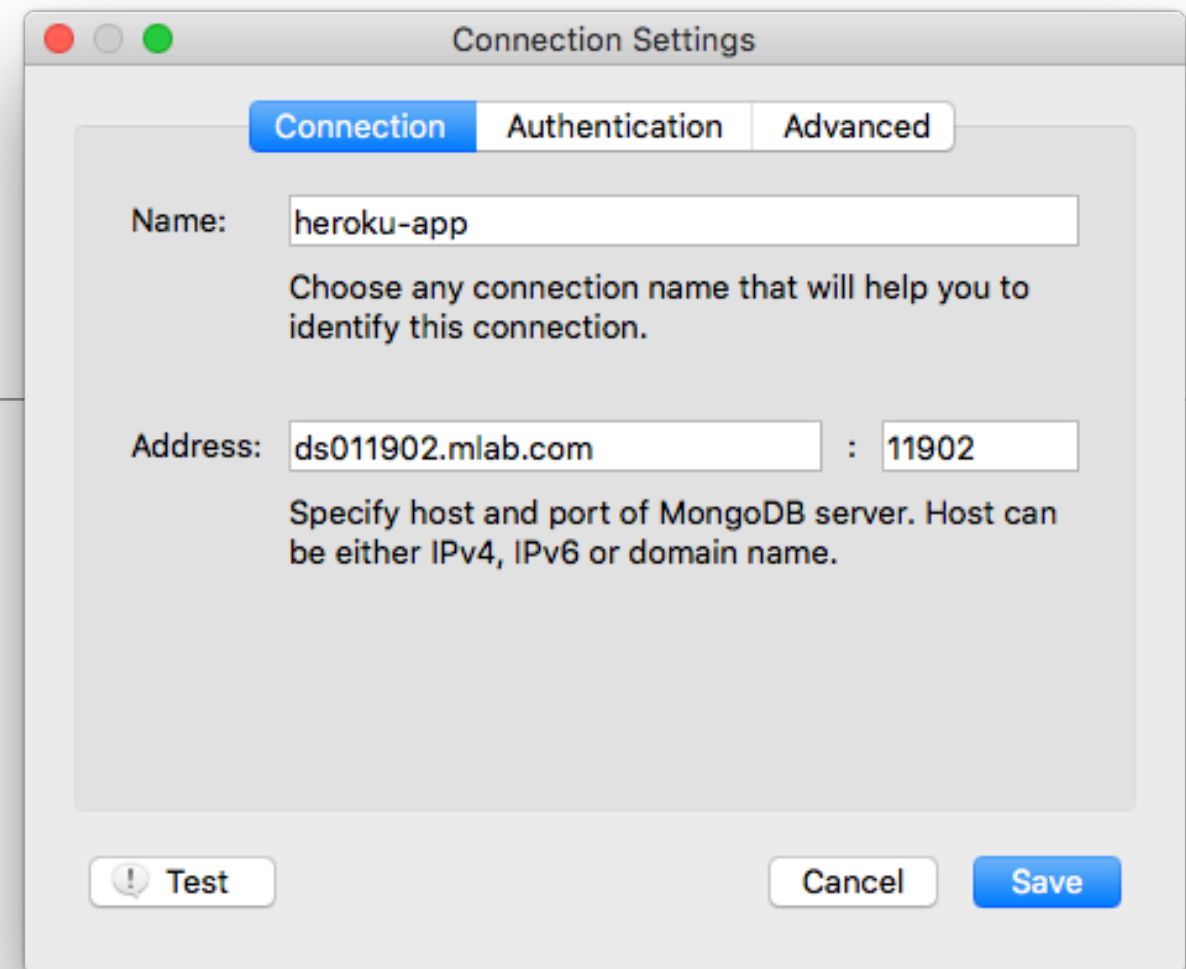
- This encodes the following connection settings:

```
address: ds011902.mlab.com
port: 11902
database: heroku_4pt2zvkj
password: omev5e4sctvbiala0i1t5cbstdj
```



# Robomongo Connection

```
address: ds011902.mlab.com
port: 11902
database: heroku_4pt2zvkJ
password: omev5e4sctvbiala0i1t5cbstdj
```



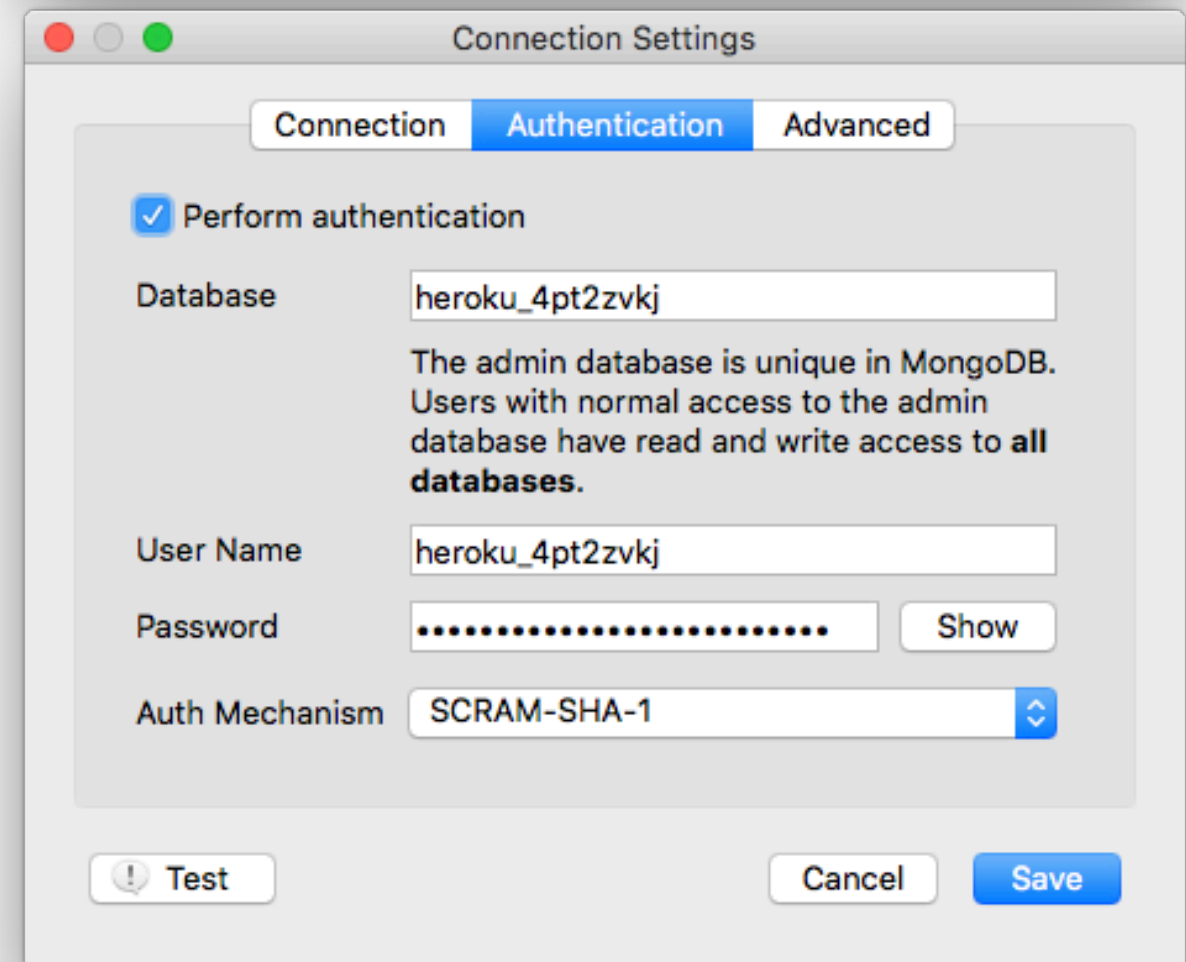
Connection Settings

Connection Authentication Advanced

Name: heroku-app  
Choose any connection name that will help you to identify this connection.

Address: ds011902.mlab.com : 11902  
Specify host and port of MongoDB server. Host can be either IPv4, IPv6 or domain name.

Test Cancel Save



Connection Settings

Connection Authentication Advanced

Perform authentication

Database heroku\_4pt2zvkJ  
The admin database is unique in MongoDB. Users with normal access to the admin database have read and write access to **all databases**.

User Name heroku\_4pt2zvkJ

Password ..... Show

Auth Mechanism SCRAM-SHA-1

Test Cancel Save

- New Connection (2)
- heroku-app (1)
  - heroku\_4pt2zvkJ
    - Collections (3)
      - System
      - donations
      - Indexes
      - users**
        - Indexes
    - Functions
    - Users

db.getColle... db.get... db.getColle... db.get... db.getColle... db.get... db.getColle... db.get...

```
heroku-app ds011902.mlab.com:11902 heroku_4pt2zvkJ  
db.getCollection('users').find({})
```

users 0.114 sec. 0 50

Key	Value	Type
(1) ObjectId("5725d5dd6...)	{ 6 fields }	Object
_id	ObjectId("5725d5dd680385110...)	ObjectId
firstName	homer	String
lastName	simpson	String
email	homer@simpson.com	String
password	secret	String
_v	0	Int32