

Deployment

Noah Raby

So you've made a web app...
Now what?

Deployment

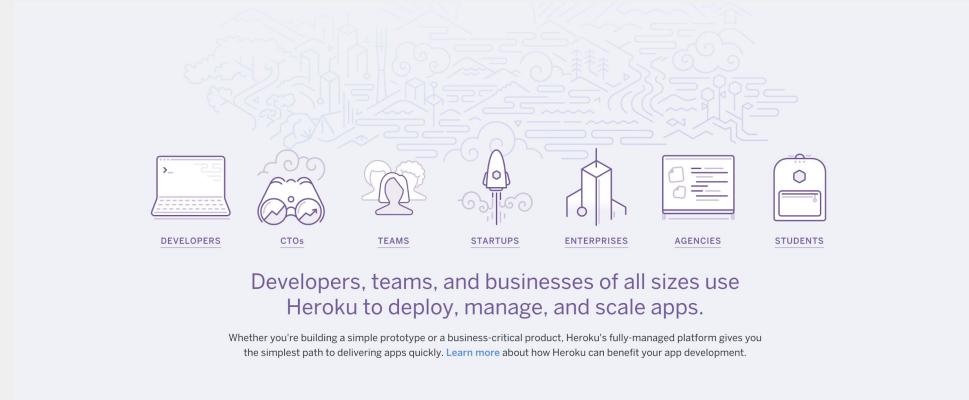
- We need to deploy! What is deployment?
 - Making your web app accessible to the world!
 - Really just getting somebody else to run your server.js
 - npm run start
 - Now any time someone wants to access your site, they can visit a url, which will open up a client
 - localhost:5000 -> yourwebsite.com

Agenda: Try Deploying Until It Works



Deployment

- We're going to use Heroku to deploy.
- yourappname.herokuapp.com
- <http://catbook2021.herokuapp.com/>



Deployment: visual walkthrough

The screenshot shows the Heroku dashboard interface. At the top left is the Heroku logo. In the center is a search bar with the placeholder text "Jump to Favorites, Apps, Pipelines, Spaces...". On the far right are three icons: a grid, a person, and a gear. Below the search bar is a navigation menu with a user icon and the text "Personal". To the right of this is a "New" button, which is highlighted with a large red oval. A search bar below the menu contains the placeholder "Filter apps and pipelines". The main content area displays a list of five applications:

App Name	Type	Region	Star Rating
catbook-workshop2	Node.js	heroku-18 · United States	★
fritter-a3-mfarejow	Node.js	heroku-18 · United States	★
mfarejow-fritter	Node.js	heroku-18 · United States	★
sound-space	Node.js	heroku-18 · United States	★
spuddies	Node.js	heroku-18 · United States	★

Deployment: visual walkthrough

The screenshot shows the Heroku dashboard interface. At the top left is the Heroku logo. A search bar in the center contains the placeholder text "Jump to Favorites, Apps, Pipelines, Spaces...". On the far right are two small icons: a grid and a user profile.

In the upper right corner, there is a "New" button with a dropdown menu. The menu items are "Create new app" (highlighted with a red oval) and "Create new pipeline".

The main content area displays a list of five applications:

- catbook-workshop2**: Node.js · heroku-18 · United States
- fritter-a3-mfarejow**: Node.js · heroku-18 · United States
- mfarejow-fritter**: Node.js · heroku-18 · United States
- sound-space**: Node.js · heroku-18 · United States
- spuddies**: Node.js · heroku-18 · United States

A search bar labeled "Filter apps and pipelines" is located above the application list.

Deployment: visual walkthrough

The screenshot shows the Heroku web interface for creating a new app. At the top left is the Heroku logo. A search bar in the center contains the placeholder text "Jump to Favorites, Apps, Pipelines, Spaces...". On the far right are account and settings icons. Below the header, the title "Create New App" is centered. The main form starts with a red number "1" above the "App name" field, which contains "test-deploy-xd". A red circle highlights this field, and a tooltip below it says "test-deploy-xd is available". The next section is "Choose a region" with a dropdown menu set to "United States". Below that is a "Add to pipeline..." button. A red number "2" is placed above the "Create app" button, which is highlighted with a red circle.

Note: the app name you provide will end up in your deployed url!

Deployment: visual walkthrough

The screenshot shows the Heroku web interface for the app "test-deploy-xd". The "Deploy" tab is highlighted with a red box. Below it, there are sections for adding the app to a pipeline and connecting to GitHub. A red circle highlights the "GitHub" button under the deployment methods.

HEROKU

Personal > test-deploy-xd

Overview Resources Deploy Metrics Activity Access Settings

Add this app to a pipeline

Create a new pipeline or choose an existing one and add this app to a stage in it.

Add this app to a stage in a pipeline to enable additional features

Pipelines let you connect multiple apps together and promote code between them. [Learn more.](#)

Pipelines connected to GitHub can enable review apps, and create apps for new pull requests. [Learn more.](#)

Choose a pipeline

Deployment method

Heroku Git Use Heroku CLI

GitHub Connect to GitHub

Container Registry Use Heroku CLI

Deploy using Heroku Git

Use git in the command line or a GUI tool to deploy this app.

Install the Heroku CLI

Download and install the [Heroku CLI](#).

If you haven't already, log in to your Heroku account and follow the prompts to create a new SSH public key.

```
$ heroku login
```

Create a new Git repository

Initialize a git repository in a new or existing directory

```
$ cd my-project/
```

Deployment: visual walkthrough

The screenshot shows the Heroku Dashboard interface. At the top, there's a navigation bar with the Heroku logo, a search bar labeled "Jump to Favorites, Apps, Pipelines, Spaces...", and user profile icons.

Add this app to a pipeline
Create a new pipeline or choose an existing one and add this app to a stage in it.

Add this app to a stage in a pipeline to enable additional features
Pipelines let you connect multiple apps together and **promote code** between them.
[Learn more.](#)

Pipelines connected to GitHub can enable **review apps**, and create apps for new pull requests.
[Learn more.](#)

Choose a pipeline

Deployment method

Connect to GitHub
Connect this app to GitHub to enable code diffs and deploys.

GitHub Connect to GitHub

Container Registry Use Heroku CLI

A dropdown menu is open, listing GitHub organizations: dansun123, CSOP, PokerLoft, scholar-chat, weblab-class, weblab-staff, and **weblab-workshops**. The "weblab-workshops" option is highlighted with a red box.

A search bar contains the text "catbook-react". A "Search" button is to the right.

Below the search bar, a message says "Missing a GitHub organization? [Ensure Heroku Dashboard has team access.](#)"

Two connection buttons are shown: "weblab-workshops/catbook-react" and "weblab-workshops/catbook". Each has a "Connect" button to its right.

Deployment: visual walkthrough

one and add this app to a stage in it.

together and **promote code** between them.
[Learn more.](#)

apps, and create apps for new pull requests.
[Learn more.](#)

Choose a pipeline

Deployment method

Heroku Git
Use Heroku CLI

Github
Connect to GitHub

Container Registry
Use Heroku CLI

Connect to GitHub

Connect this app to GitHub to enable code
diffs and deploys.

Search for a repository to connect to

weblab-workshops catbook-react

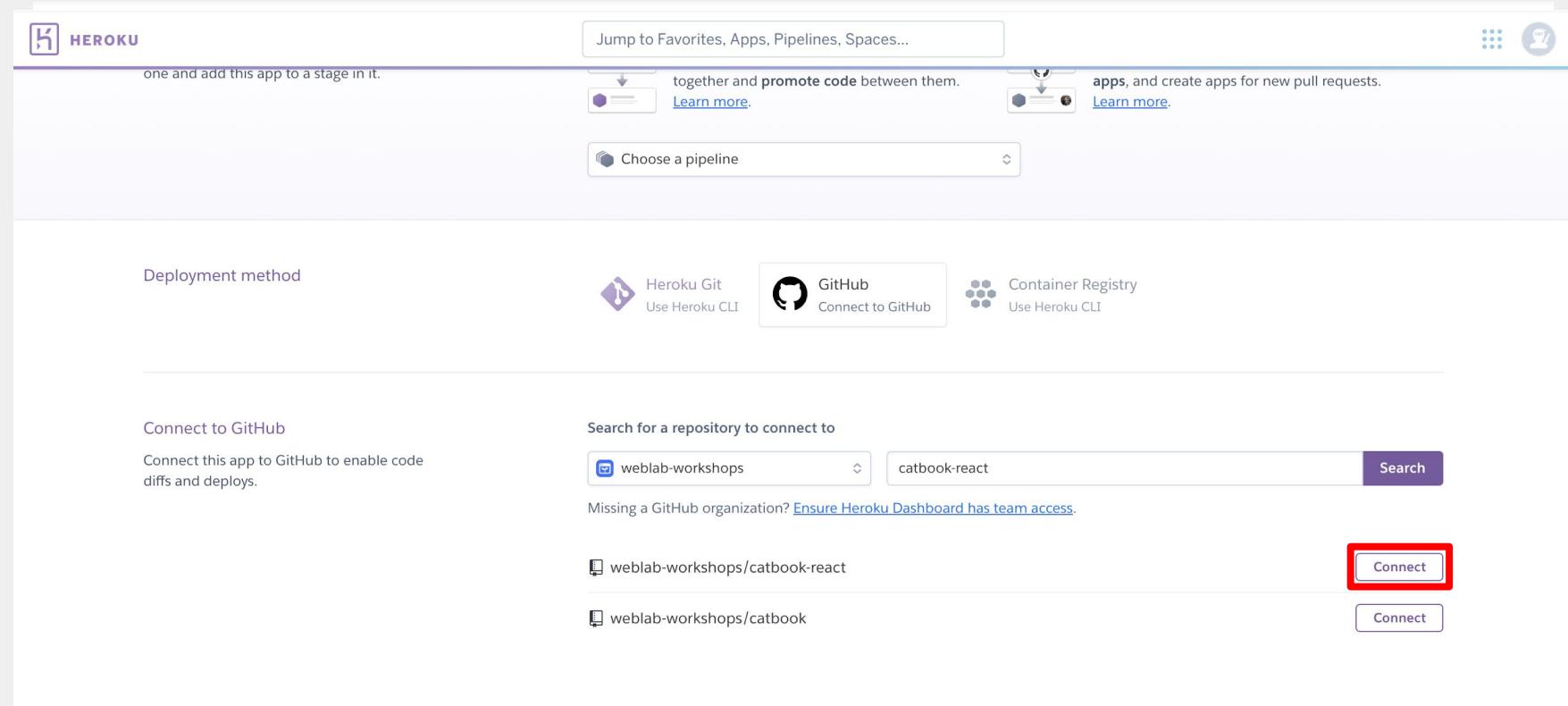
Missing a GitHub organization? [Ensure Heroku Dashboard has team access.](#)

weblab-workshops/catbook-react

weblab-workshops/catbook

Connect

Connect



Deployment: visual walkthrough

The screenshot shows the Heroku Deployment settings page. At the top, there's a navigation bar with the Heroku logo, a search bar labeled "Jump to Favorites, Apps, Pipelines, Spaces...", and a user icon.

Automatic deploys

Enables a chosen branch to be automatically deployed to this app.

Enable automatic deploys from GitHub

Every push to the branch you specify here will deploy a new version of this app. **Deploys happen automatically:** be sure that this branch is always in a deployable state and any tests have passed before you push. [Learn more](#).

Choose a branch to deploy

master

Wait for CI to pass before deploy

Only enable this option if you have a Continuous Integration service configured on your repo.

Enable Automatic Deploys

Manual deploy

Deploy the current state of a branch to this app.

Deploy a GitHub branch

This will deploy the current state of the branch you specify below. [Learn more](#).

Choose a branch to deploy

master

Deploy Branch

Deployment: visual walkthrough

Manual deploy

Deploy the current state of a branch to this app.

Deploy a GitHub branch

This will deploy the current state of the branch you specify below. [Learn more](#).

Enter the name of the branch to deploy

Deploy Branch

Receive code from GitHub ✓

Build master ea4461a5 ✓

Release phase ✓

Deploy to Heroku ✓

Your app was successfully deployed.

 View

Boom! We've deployed!

Or did we?

Deployment: follow through

Manual deploy

Deploy the current state of a branch to this app.

Deploy a GitHub branch

This will deploy the current state of the branch you specify below. [Learn more](#).

Enter the name of the branch to deploy

Deploy Branch

Receive code from GitHub ✓

Build master ea4461a5 ✓

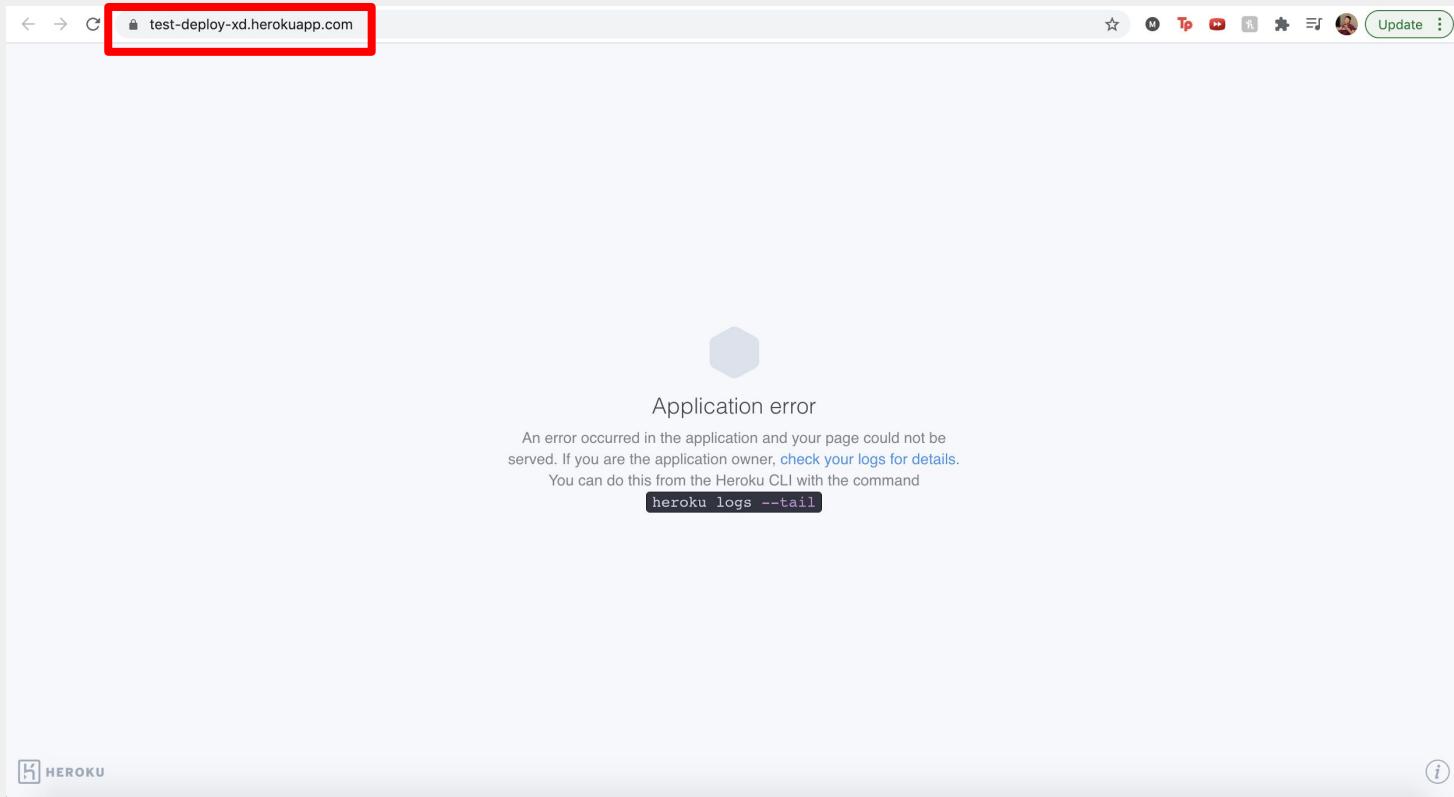
Release phase ✓

Deploy to Heroku ✓

Your app was successfully deployed.

 View

Deployment: follow through



Deployment: follow through

- Let's check the logs

The screenshot shows the Heroku dashboard for the app 'test-deploy-xd'. The 'More' menu is open, and the 'View logs' option is highlighted with a red circle.

App Details:
Personal > test-deploy-xd
GitHub weblab-class/MFarejowicz

Navigation:
Overview Resources Deploy Metrics Activity Access Settings

Pipelines:
Add this app to a pipeline
Create a new pipeline or choose an existing one and add this app to a stage in it.
Pipelines let you connect multiple apps together and **promote code** between them.
[Learn more.](#)

Deployment method:
Heroku Git (Use Heroku CLI) GitHub Connected (Connected) Container Registry (Use Heroku CLI)

GitHub Integration:
App connected to GitHub
Code diffs, manual and auto deploys are available for this app.
Connected to weblab-class/MFarejowicz by MFarejowicz
Disconnect...
Releases in the [activity feed](#) link to GitHub to view commit diffs

Deployment: follow through

The screenshot shows the Heroku application logs for the 'test-deploy-xd' app. The logs detail the deployment process, including the build step, the creation of release v4, and the starting of the web process. A red box highlights the startup logs for the web process, which shows it connecting to MongoDB and failing to bind to the port due to a boot timeout.

```
2020-01-16T02:57:31.000000+00:00 app[api]: Build started by user mfarejow@mit.edu
2020-01-16T02:58:18.789172+00:00 app[api]: Release v4 created by user mfarejow@mit.edu
2020-01-16T02:58:19.891764+00:00 heroku[web.1]: State changed from crashed to starting
2020-01-16T02:58:18.789172+00:00 app[api]: Deploy 7370b0d8 by user mfarejow@mit.edu
2020-01-16T02:58:23.422497+00:00 heroku[web.1]: Starting process with command `npm start`
2020-01-16T02:58:22.000000+00:00 app[api]: Build succeeded
2020-01-16T02:58:25.484591+00:00 app[web.1]:
2020-01-16T02:58:25.484609+00:00 app[web.1]: > weblab-skeleton@1.0.0 start /app
2020-01-16T02:58:25.484612+00:00 app[web.1]: > nodemon
2020-01-16T02:58:25.484614+00:00 app[web.1]:
2020-01-16T02:58:25.849671+00:00 app[web.1]: [33m[nodemon] 1.19.4 [39m
2020-01-16T02:58:25.849684+00:00 app[web.1]: [33m[nodemon] to restart at any time, enter `rs` [39m
2020-01-16T02:58:25.850255+00:00 app[web.1]: [33m[nodemon] watching dir(s): **/* [39m
2020-01-16T02:58:25.850344+00:00 app[web.1]: [33m[nodemon] watching extensions: js,mjs,json [39m
2020-01-16T02:58:25.850793+00:00 app[web.1]: [32m[nodemon] starting `node server/server.js` [39m
2020-01-16T02:58:26.526566+00:00 app[web.1]: Warning: connect.session() MemoryStore is not
2020-01-16T02:58:26.526616+00:00 app[web.1]: designed for a production environment, as it will leak
2020-01-16T02:58:26.605991+00:00 app[web.1]: Server running on port: 3000
2020-01-16T02:58:26.739714+00:00 app[web.1]: Connected to MongoDB
2020-01-16T02:59:23.915070+00:00 heroku[web.1]: State changed from starting to crashed
2020-01-16T02:59:23.919993+00:00 heroku[web.1]: State changed from crashed to starting
2020-01-16T02:59:23.788758+00:00 heroku[web.1]: Error R10 (Boot timeout) --> Web process failed to bind to $PORT within 60 seconds of launch
2020-01-16T02:59:23.788758+00:00 heroku[web.1]: Stopping process with SIGKILL
2020-01-16T02:59:23.897193+00:00 heroku[web.1]: Process exited with status 137
2020-01-16T02:59:27.734012+00:00 heroku[web.1]: Starting process with command `npm start`

2020-01-16T02:59:29.978665+00:00 app[web.1]: > weblab-skeleton@1.0.0 start /app
2020-01-16T02:59:29.978667+00:00 app[web.1]: > nodemon
2020-01-16T02:59:29.978669+00:00 app[web.1]:
2020-01-16T02:59:30.318584+00:00 app[web.1]: [33m[nodemon] 1.19.4 [39m
```

Autoscroll with output Save

Deployment: follow through

```
2020-01-16T02:58:26.605991+00:00 app[web.1]: Server running on port: 3000
2020-01-16T02:58:26.739714+00:00 app[web.1]: Connected to MongoDB
2020-01-16T02:59:23.915070+00:00 heroku[web.1]: State changed from starting to crashed
2020-01-16T02:59:23.919993+00:00 heroku[web.1]: State changed from crashed to starting
2020-01-16T02:59:23.788758+00:00 heroku[web.1]: Error R10 (Boot timeout) -> Web process failed to bind to $PORT within 60 seconds of launch
2020-01-16T02:59:23.788758+00:00 heroku[web.1]: Stopping process with SIGKILL
2020-01-16T02:59:23.897193+00:00 heroku[web.1]: Process exited with status 137
2020-01-16T02:59:27.734012+00:00 heroku[web.1]: Starting process with command `npm start`
```

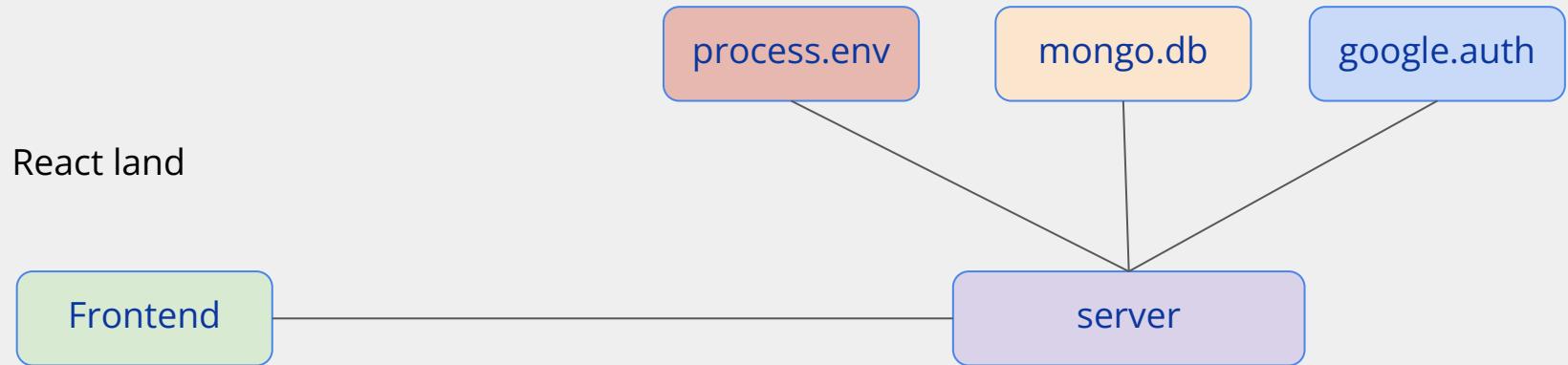
Deployment: follow through

Heroku dynamically assigns your app a port, so you can't set the port to a fixed number. Heroku adds the port to the `env`, so you can pull it from there. Switch your listen to this:

```
.listen(process.env.PORT || 3000)
```

That way it'll still listen to port 3000 when you test locally, but it will also work on Heroku.

Understanding process.env



Understanding process.env

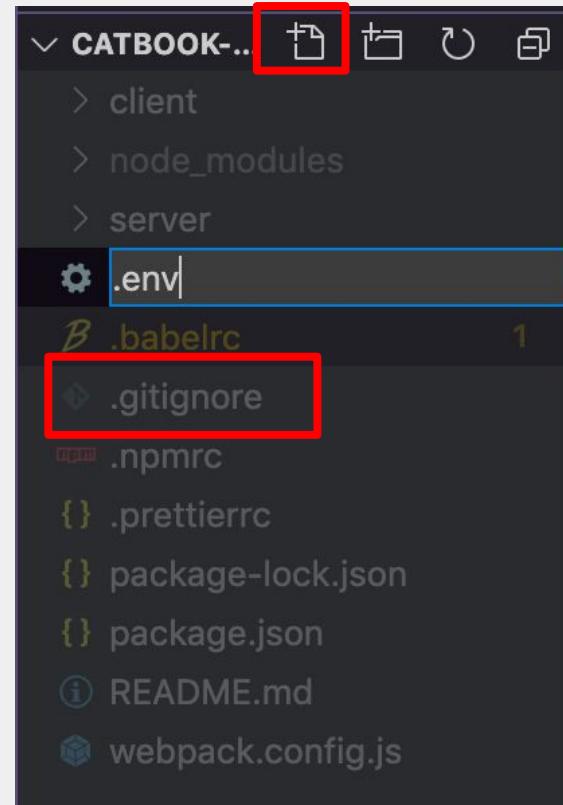
- process.env serves as a sort of **global variable** for our **backend** files to access
- We'll use it to determine which PORT to run on
- We can also use it to hide sensitive info

Understanding process.env

```
34 // Server configuration below
35 const mongoConnectionURL =
36 "mongodb+srv://weblab:jAT4p5IAgYWQgR@catbook-yLndp.mongodb.net/test?retryWrites=true&w=majority";
37 const databaseName = "catbook";
```

How to use process.env

- Step 1: Add a file named `.env` to your project's outermost folder
- Step 2: Add `.env` to your project's `.gitignore` file



How to use process.env

- Step 1: Add a file named `.env` to your project's outermost folder
- Step 2: Add `.env` to your project's `.gitignore` file

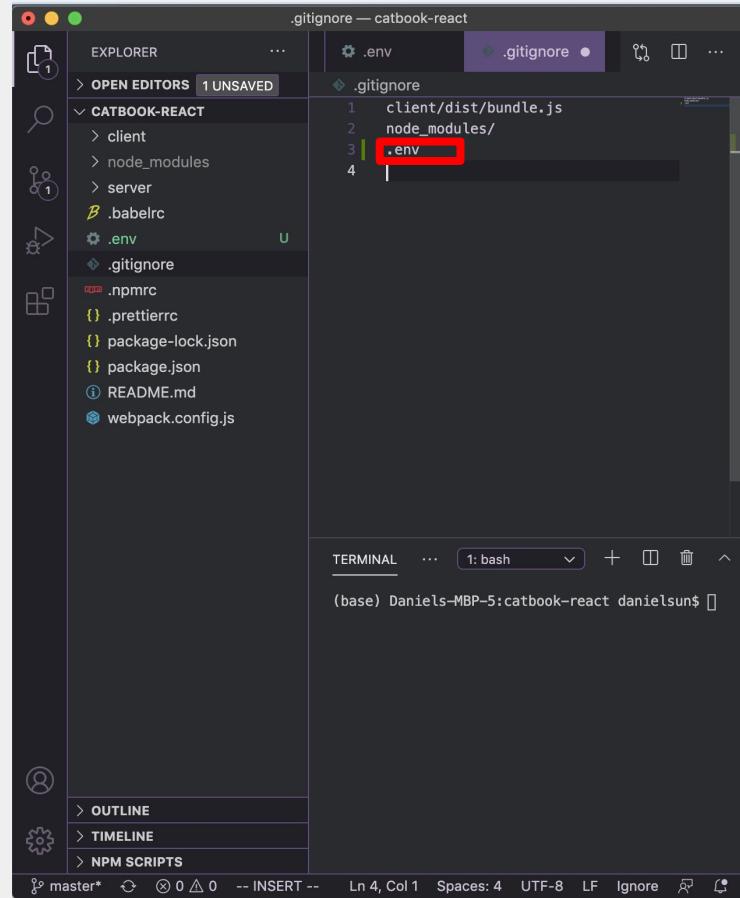
The screenshot shows the VS Code interface with the following details:

- EXPLORER:** Shows the project structure under `CATBOOK-REACT`, including `client`, `node_modules`, `server`, `.babelrc`, `.env`, `.gitignore`, `.npmrc`, `.prettierrc`, `package-lock.json`, `package.json`, `README.md`, and `webpack.config.js`.
- gitignore — catbook-react**: The current file being edited.
- Content of .gitignore:**

```
client/dist/bundle.js
node_modules/
[REDACTED]
```
- TERMINAL:** Shows the command `(base) Daniels-MBP-5:catbook-react danielsun$`.
- Status Bar:** Shows the current branch is `master*`, mode is `INSERT`, and other terminal settings like spaces and encoding.

How to use process.env

- Step 1: Add a file named `.env` to your project's outermost folder
- Step 2: Add `.env` to your project's `.gitignore` file



A screenshot of the Visual Studio Code interface. The Explorer sidebar shows a project structure with folders like 'CATBOOK-REACT', 'client', 'node_modules', 'server', '.babelrc', '.env' (which is highlighted in green), '.gitignore', '.npmrc', '.prettierrc', 'package-lock.json', 'package.json', 'README.md', and 'webpack.config.js'. In the center, the 'gitignore' file is open in the editor, showing the following content:

```
client/dist/bundle.js
node_modules/
.env
```

The line containing `.env` is highlighted with a red box. At the bottom, the terminal window shows the command `(base) Daniels-MBP-5:catbook-react danielsun$`.

How to use process.env

- Step 3: Add relevant *sEcREts* to the .env file

```
⚙ .env
1   ATLAS_SRV = "mongodb+srv://weblab:jAT4po55IAgYWQgR@catbook-ylndp.mongodb.net/test?retryWrites=true&w=majority"
2   SESSION_SECRET = "session-secret"
3
```

- Note! Everyone has to make their own .env file!

How to use process.env

- Step 4: Install the dotenv package!
 - Use `npm install dotenv`
- Step 5: Add the following line near the top of `server.js`
 - `require('dotenv').config();`
- Step 6: Use `process.env.VAR_NAME` where needed!

How to use process.env

```
// Server configuration below
const mongoConnectionURL =
  "mongodb+srv://weblab:jAT4po55IAgYWQgR@catbook-ylndp.mongodb.net/test?retryWrites=true&w=majority"
```

.env

```
1  ATLAS_SRV = "mongodb+srv://weblab:jAT4po55IAgYWQgR@catbook-ylndp.mongodb.net/test?retryWrites=true&w=majority"
2  SESSION_SECRET = "session-secret"
3
```

```
app.use(
  session({
    secret: "session-secret",
    resave: false,
    saveUninitialized: false,
  })
);
```

How to use process.env

```
34 // Server configuration below
35 const mongoConnectionURL = process.env.ATLAS_SRV;
36 const databaseName = "catbook";
```

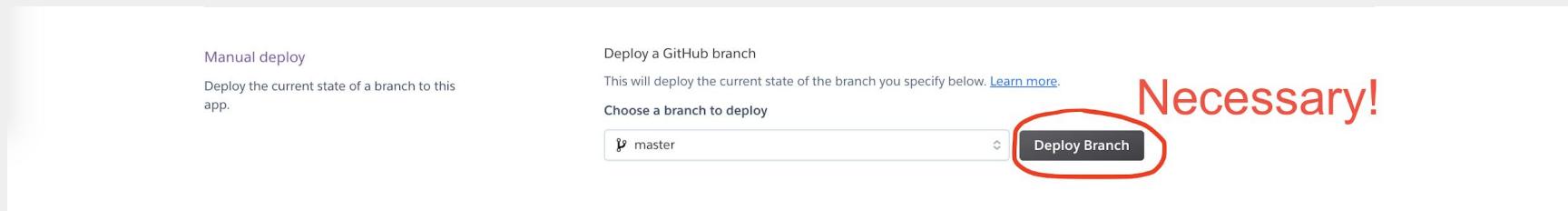
```
55 // set up a session, which will persist login data across requests
56 app.use(
57   session({
58     secret: process.env.SESSION_SECRET,
59     resave: false,
60     saveUninitialized: false,
61   })
62 );
```

How to use process.env

```
95  const port = process.env.PORT || 3000;  
96  const server = http.Server(app);  
97  socket.init(server);
```

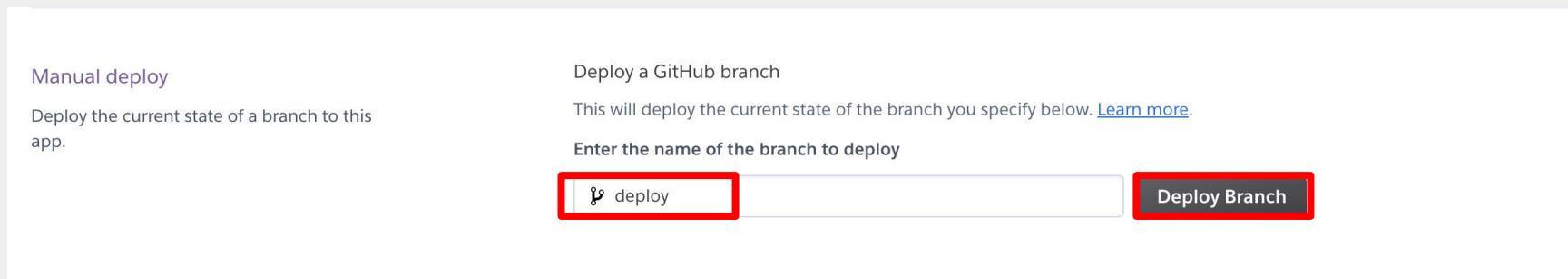
Deployment: attempt 2

- After making these changes, always remember to:
 - git add <file>
 - git commit -m "message"
 - git push
- After that, we can redeploy!
 - Head to heroku, then the deploy tab



Deployment: attempt 2

- For following along in catbook-react:
 - git reset --hard
 - git checkout deploy
- After that, we can redeploy!
 - Head to heroku, then the deploy tab



The screenshot shows the 'Manual deploy' section of the Heroku Deploy Branch interface. It includes fields for specifying a GitHub branch and a branch name, along with a 'Deploy Branch' button.

Manual deploy

Deploy a GitHub branch

Deploy the current state of a branch to this app.

This will deploy the current state of the branch you specify below. [Learn more](#).

Enter the name of the branch to deploy

Deploy Branch

Will that work?

Understanding config vars

- Heroku doesn't actually have access to our .env files!
- To fix this, we can set Heroku config variables
 - Heroku version of a .env file

How to use config vars

The screenshot shows the Heroku Settings page for the app "test-deploy-xd". The "Settings" tab is circled in red. The page includes sections for adding the app to a pipeline, choosing a pipeline, deployment methods (Heroku Git, GitHub Connected, Container Registry), and automatic deploys.

Add this app to a pipeline
Create a new pipeline or choose an existing one and add this app to a stage in it.

Add this app to a stage in a pipeline to enable additional features
Pipelines let you connect multiple apps together and **promote code** between them.
[Learn more.](#)

Pipelines connected to GitHub can enable **review apps**, and create apps for new pull requests.
[Learn more.](#)

Choose a pipeline

Deployment method

Heroku Git
Use Heroku CLI

GitHub Connected

Container Registry
Use Heroku CLI

App connected to GitHub
Code diffs, manual and auto deploys are available for this app.

Connected to [weblab-class/MFarejowicz](#) by [MFarejowicz](#) [Disconnect...](#)

Releases in the [activity feed](#) link to GitHub to view commit diffs

Automatic deploys
Enables a chosen branch to be automatically deployed to this app.

Enable automatic deploys from GitHub
Every push to the branch you specify here will deploy a new version of this app. **Deploys happen automatically:** be sure that this branch is always in a deployable state and any tests have passed before you push. [Learn more.](#)

How to use config vars

The screenshot shows the Heroku application settings page for 'test-deploy-xd'. At the top, there's a navigation bar with the Heroku logo, a search bar, and user account information. Below the navigation, the app name 'test-deploy-xd' is displayed along with its GitHub repository link. A navigation menu includes 'Overview', 'Resources', 'Deploy', 'Metrics', 'Activity', 'Access', and 'Settings' (which is underlined). In the main content area, there's a section titled 'Config Vars' with a descriptive paragraph. To the right of this paragraph is a button labeled 'Reveal Config Vars', which is circled in red. Below this section is a table with 'Info' and 'Region' columns. The 'Info' column lists 'Stack', 'Framework', 'Slug Size', 'GitHub Repo', and 'Heroku Git URL'. The 'Region' column lists 'United States', 'heroku-18', 'Node.js', '39.8 MiB of 500 MiB', and the GitHub repository URL again. At the bottom, there's a 'Buildpacks' section with an 'Add buildpack' button.

Info	Region
Stack	United States
Framework	heroku-18
Slug Size	Node.js
GitHub Repo	39.8 MiB of 500 MiB
Heroku Git URL	https://git.heroku.com/test-deploy-xd.git

Buildpacks

Add buildpack

How to use config vars

The screenshot shows the Heroku dashboard for the app 'test-deploy-xd'. The 'Settings' tab is selected. The 'Name' field is set to 'test-deploy-xd'. The 'Config Vars' section is visible, showing a message: 'There are no config vars for this app yet' and a link to 'Learn about config vars'. Below this, there are input fields for 'KEY' and 'VALUE', and a red oval highlights the 'Add' button. The 'Info' section at the bottom provides details like Region (United States), Stack (heroku-18), Framework (Node.js), Slug Size (39.8 MiB of 500 MiB), and GitHub Repo (weblab-class/MFarejowicz).

HEROKU

Jump to Favorites, Apps, Pipelines, Spaces...

Personal > test-deploy-xd

GitHub weblab-class/MFarejowicz

Overview Resources Deploy Metrics Activity Access Settings

Name test-deploy-xd Edit

Config Vars

Config Vars Hide Config Vars

There are no config vars for this app yet

[Learn about config vars](#) in the Dev Center.

KEY VALUE Add

Info

Region	United States
Stack	heroku-18
Framework	Node.js
Slug Size	39.8 MiB of 500 MiB
GitHub Repo	weblab-class/MFarejowicz

How to use config vars

HEROKU

Jump to Favorites, Apps, Pipelines, Spaces...

Personal > test-deploy-xd

GitHub weblab-class/MFarejowicz

Open app More

Overview Resources Deploy Metrics Activity Access Settings

Name test-deploy-xd Edit

Config Vars

Config Vars

Hide Config Vars

ATLAS_SRV

SESSION_SECRET

session-secret

KEY VALUE Add

Config vars change the way your app behaves. In addition to creating your own, some add-ons come with their own.

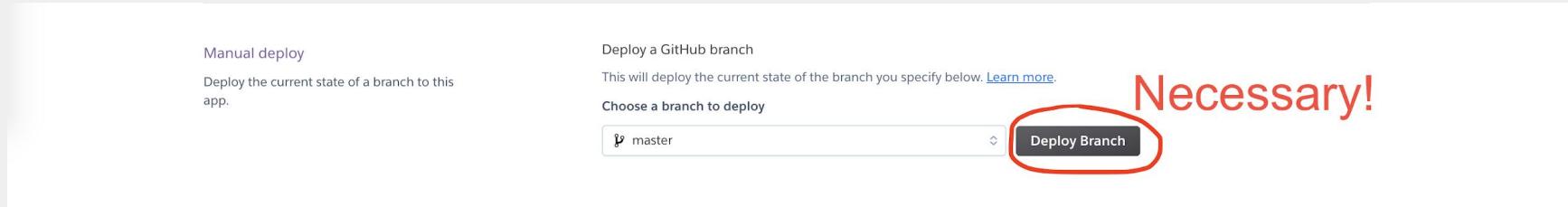
Note: we don't need the quotes around the value here!

Recap: .env vs config vars

- `.env` is a *local* file, that we use to keep sensitive information private
- `config vars` are a *heroku* feature, that we use for a similar reason.

Deployment: attempt 3

- After setting config vars, we'll need to redeploy again!
 - Head to heroku, then the deploy tab



Success?

Let's test the site!

Now what's wrong?

The screenshot shows a web application interface. At the top left is a Google logo with a "Login" button. Below it, the text "Good luck on your project :)" is displayed. Underneath this, the heading "What we provide in this skeleton" is followed by a bulleted list:

- Google Auth (Skeleton.js & auth.js)
- Socket Infrastructure (client-socket.js & server-socket.js)
- User Model (auth.js & user.js)

Below this, the heading "What you need to change" is followed by another bulleted list:

- Change the font in utilities.css
- Change the Frontend CLIENT_ID for Google Auth (Skeleton.js)
- Change the Server CLIENT_ID for Google Auth (auth.js)
- Change the Database SRV for Atlas (server.js)
- Change the Database Name for MongoDB (server.js)
- Add a favicon to your website at the path client/dist/favicon.ico
- Update website title in client/dist/index.html

Fixing login

- Recall we linked our website auth with google and had to whitelist localhost:5000
- weblab.to/clientid
- Let's head to console.developers.google.com

Fixing login

The screenshot shows the Google APIs console interface. At the top, there is a navigation bar with the text "Google APIs" and "test-deploy". On the far right of the navigation bar are icons for help, notifications, and user profile.

The main area has a header "APIs & Services" with a sub-header "APIs & Services" and a button "+ ENABLE APIs AND SERVICES". Below this is a message: "You don't have any APIs available to use yet. To get started, click 'Enable APIs and services' or go to the [API library](#)".

The left sidebar contains a list of options:

- Dashboard
- Library
- Credentials** (This option is highlighted with a red oval)
- OAuth consent screen
- Domain verification
- Page usage agreements

Fixing login

Google APIs test-deploy ▾

Search for APIs and Services

?

1

⋮

Profile

API	APIs & Services	Credentials	+ CREATE CREDENTIALS	DELETE								
	Dashboard	Create credentials to access your enabled APIs. Learn more										
	Library											
	Credentials	<table border="1"><thead><tr><th>Name</th><th>Creation date</th><th>Restrictions</th><th>Key</th></tr></thead><tbody><tr><td>No API keys to display</td><td></td><td></td><td></td></tr></tbody></table>			Name	Creation date	Restrictions	Key	No API keys to display			
Name	Creation date	Restrictions	Key									
No API keys to display												
	OAuth consent screen											
	Domain verification											
	Page usage agreements											

API Keys

Name	Creation date	Restrictions	Key
No API keys to display			

OAuth 2.0 Client IDs

Name	Creation date	Type	Client ID	⋮
Test-Deploy	Jan 4, 2021	Web application	840472150171-4421...	

Service Accounts

Email	Name
No service accounts to display	

[Manage service accounts](#)

Fixing login

Google APIs test-deploy

Client ID for Web application

API APIs & Services

Credentials

Name * OAuth client

The name of your OAuth 2.0 Client ID. This is different from your app name and will not be shown to end users.

The domains of the URIs you add below will be automatically added to your OAuth consent screen as authorized domains.

Authorized JavaScript origins

For use with requests from a browser

URIs

http://localhost:5000

https://test-deploy-xd.herokuapp.com/

+ ADD URI

Authorized redirect URIs

For use with requests from a web server

+ ADD URI

SAVE CANCEL

Client ID 545750287095-1jtj8pmk85n3lqncnjf5bhjh9rlg2m5.apps.googleusercontent.com
Client secret Cfg-NQZJPNzXIPkn57nSs7CV
Creation date January 16, 2020 at 1:22:49 PM GMT-5
Total usage (last 30 days) 0

xd.herokuapp.com/'. The second entry is circled in red with the number '2'. Below it, the '+ ADD URI' button is circled in red with the number '1'."/>

2

1

Fixing login

- Hit save on the google console, and then we try our website out again!

Success! (hopefully)

Things might not work out.

Recap

- Link your app to Heroku through Github
- Create .env file and add to your .gitignore
 - Remember .env contains stuff that github should not know about
- Change port from 3000 to process.env.PORT || 3000
 - process.env is a big object with fields (such as PORT) accessible by the backend
- Move sensitive data into the .env file
 - Including your mongo srv and session secret
 - Access them in backend code with process.env.VARIABLE_NAME
- Make sure heroku knows about your .env variables
 - Open config vars in settings and put them in. You don't need quotes for strings
- Tell your google auth to allow logins from your new website address
 - weblab.to/clientid
- Click deploy