

## Was ist Heroku?

**Heroku ist eine containerbasierte Cloud Plattform<sup>1</sup>** mit Unterstützung für mehrere Programmiersprachen. Derzeit unterstützt Heroku JavaScript (Node.js), PHP, Java, Python, Go, Scala und Clojure. Die Container von Heroku sind virtuelle Computersysteme: **Dynos** genannt. Eine Anwendung läuft also isoliert von allen anderen Anwendungen in einem Dyno. Ein Dyno kann jederzeit an geänderten Anforderungen angepasst werden, indem mehr Features und Ressourcen hinzugefügt werden. Heroku und alle darauf aufbauenden Anwendungen werden auf AWS gehostet.

Heroku berechnet seinen Kunden monatliche Gebühren, die sich nach der Anzahl der verwendeten virtuellen Computer richten. Es gibt einen Free-Tarif, der für Entwickler geeignet ist, die persönliche Projekte und nicht-kommerzielle Anwendungen bereitstellen möchten.

Als Anwendungsbeispiel verwenden wir eine Node.js Applikation, die das **json-server** Package verwendet.

## JSON Server

Das npm Package **json-server** kann sowohl stand-alone als auch als Library verwendet werden. Hier verwenden wir **json-server** als Library.

```
const jsonServer = require('json-server');
const server = jsonServer.create();
const router = jsonServer.router('employees.json');
const middlewares = jsonServer.defaults();
const port = process.env.PORT || 3000;

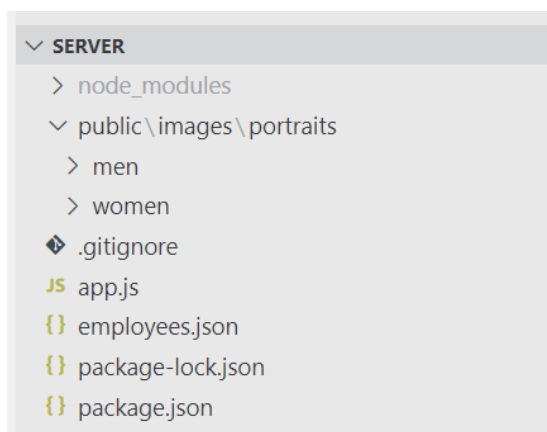
server.use(middlewares);
server.use(router);

server.listen(port, () => console.log('running...'));
```

Wenn wir den Server auf Heroku hosten, vergibt die Plattform ihre eigene Portnummer. Diese steht in `process.env.PORT`. Daher fragen wir die Variable in unserem Code ab.

Der Datenfile ist in unserem Beispiel **employees.json**.

Im **/public** Verzeichnis befinden sich die Bilder zu dem Datenfile.



<sup>1</sup> Fachbegriff: Platform as a Service (PaaS)

Robert Baumgartner

Beim **json-server** verweist die Defaultroute **/** in das Verzeichnis **/public**. Daher können wir im Datenfile die Bilder relativ adressieren und sind dadurch unabhängig von der Basis URL.

```
{
  "id": "1",
  "gender": "male",
  "name": { "title": "Mr", "first": "Lucas", "last": "Garcia" },
  "location": {
    "street": "1469 route de genas",
    "city": "saint-denis",
    "state": "saône-et-loire",
    "postcode": 43181
  },
  "email": "lucas.garcia@example.com",
  "dob": { "date": "1962-11-04T05:47:30Z", "age": 56 },
  "registered": { "date": "2004-10-16T09:30:03Z", "age": 14 },
  "phone": "05-65-50-52-66",
  "cell": "06-33-21-84-33",
  "picture": {
    "large": "/images/portraits/men/41.jpg"
  },
  "nat": "FR"
},
```



Mit **npm start** kannst du den Server starten. Der **json-server** stellt alle CRUD Operationen via REST API zur Verfügung.

```
Send Request
GET http://localhost:3000/employees
#####
Send Request
DELETE http://localhost:3000/employees/1
#####
Send Request
PATCH http://localhost:3000/employees/99
Content-Type: application/json
{
  "name": {
    "title": "Mrs",
    "first": "Sarah",
    "last": "Garcia-Ludovici"
  }
}
#####
Send Request
POST http://localhost:3000/employees
Content-Type: application/json
{
  "gender": "female",
  "name": {
    "title": "Miss",
    "first": "Sarah",
    "last": "Ludovici"
  },
  "location": {
    "street": "Thaliastrasse 125",
    "city": "Vienna",
    "state": "Austria",
    "postcode": 1160
  }
}
```

Soweit unsere Beispielapplikation.

## Hosten auf Heroku

1. Besorge dir einen Account auf Heroku

2. Lege eine neue App an:

Create New App | Heroku

dashboard.heroku.com/new-app

Salesforce Platform

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

Create New App

App name

baumgartner-employees

baumgartner-employees is available

Choose a region

Europe

Add to pipeline...

Create app

3. Folge den Anweisungen:

baumgartner-employees - Heroku

dashboard.heroku.com/apps/baumgartner-employees/deploy/heroku-git

Salesforce Platform

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

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/
$ git init
$ heroku git:remote -a baumgartner-employees
```

Deploy your application

Commit your code to the repository and deploy it to Heroku using Git.

```
$ git add .
$ git commit -am "make it better"
$ git push heroku master
```

You can now change your main deploy branch from "master" to "main" for both manual and automatic deploys, please follow the instructions [here](#).

Robert Baumgartner

#### 4. Schau, ob du beim Push Errors bekommst:

```

Compressing objects: 100% (36/36), done.
Writing objects: 100% (39/39), 121.45 KiB | 12.14 MiB/s, done.
Total 39 (delta 6), reused 0 (delta 0), pack-reused 0
remote: Compressing source files... done.
remote:
remote: -----> Building on the Heroku-20 stack
remote: -----> Determining which buildpack to use for this app
remote: -----> Node.js app detected
remote:
remote: -----> Creating runtime environment
remote:
remote:       NPM_CONFIG_LOGLEVEL=error
remote:       NODE_VERBOSE=false
remote:       NODE_ENV=production
remote:       NODE_MODULES_CACHE=true
remote:
remote: -----> Installing binaries
remote:       engines.node (package.json): unspecified
remote:       engines.npm (package.json):  unspecified (use default)
remote:
remote:       Resolving node version 14.x...
remote:       Downloading and installing node 14.18.1...
remote:       Using default npm version: 6.14.15
remote:
remote: -----> Installing dependencies
remote:       Installing node modules
remote:       added 173 packages in 3.216s
remote:
remote: -----> Build
remote:
remote: -----> Caching build
remote:       - node_modules
remote:
remote: -----> Pruning devDependencies
remote:       audited 173 packages in 1.682s
remote:
remote:       13 packages are looking for funding
remote:       run `npm fund` for details
remote:
remote:       found 0 vulnerabilities
remote:
remote: -----> Build succeeded!
remote: -----> Discovering process types
remote:       Procfile declares types     -> (none)
remote:       Default types for buildpack -> web
remote:
remote: -----> Compressing...
remote:       Done: 34.3M
remote: -----> Launching...
remote:       Released v3
remote:       https://baumgartner-employees.herokuapp.com/ deployed to Heroku
remote:
remote: Verifying deploy... done.
To https://git.heroku.com/baumgartner-employees.git
* [new branch]      master -> master
    
```



#### 5. Teste mit einem Browser:

