

Curso: Spring Boot com Angular 7

<https://www.udemy.com/user/hugo-silva>

Prof. Hugo Silva

Capítulo: Implantação do Heroku e deploy do Front end

Objetivo geral:

- Implementar o Front end no Heroku

Implantando o Heroku

Checklist:

1) Cadastrando-se no [Heroku](#)

- Acesse **heroku.com** e faça o seguinte procedimento
 - Sign up
 - Preencha os dados e envie o formulário
 - Abra seu email e clique no link de confirmação
 - Defina uma senha
- Criando um novo **APP**
 - Create New App
 - Nome (opcional)
 - Escolha a localidade

2) Instalação do [Heroku CLI](#)

- <https://devcenter.heroku.com/articles/heroku-cli#download-and-install>
- Escolha a versão para seu sistema e faça o procedimento de instalação
- Abra o terminal (no Windows o cmd) e faça login:

- `heroku login`

3) Retorne para o Front end e execute o seguinte comando na pasta raiz do projeto

Nota: Este comando instala o **express** para servir os arquivos e o **path** para configurarmos o nosso servidor

- `npm install --save express path`

4) Se quisermos que o Heroku construa nosso projeto em seus servidores, precisamos lhe dizer para fazer isso e quais versões do node / npm o nosso código foi desenvolvido.

Então, no **package.json** que fica na pasta Raiz do Projeto Informe a versão do **node** / **npm**

```
"scripts": {
  (...)
  "start": "node server.js", //Ou criando o arquivo Procfile
  "postinstall": "ng build --prod --aot"
},
"engines": {
  "node": "11.1.0",
  "npm": "6.7.0"
},
```

- Lembre-se de substituir as versões do **node** e **npm** pelas versões que você possui. Você pode descobrir abrindo o terminal (no Windows o cmd) e executar os seguintes comandos:

```
o node --version
o npm --version
```

5) Como o Heroku está construindo nosso código, precisamos dar a ele a capacidade de executar todos os módulos que foram usados para desenvolver o projeto.

- Mova as dependências de **"devDependencies"** para **"dependencies"**

6) Vamos criar o nosso arquivo do servidor. No diretório principal do seu aplicativo (aquele onde fica o **package.json**), crie um arquivo chamado **server.js**. Adicione o seguinte

```
const path = require('path');
const express = require('express');
const nameApp = 'sba7-app';
const app = express();
// Servir arquivos estáticos
app.use(express.static(`${__dirname}/dist/${nameApp}`));
// Envie todos os pedidos para index.html
app.get('/*', function(req, res) {
  res.sendFile(path.join(`${__dirname}/dist/${nameApp}/index.html`));
});
// Porta padrão do Heroku
app.listen(process.env.PORT || 5000);
```

NOTA: Lembre-se que **nameApp** (nos dois casos) é o nome do seu aplicativo.

7) Agora, vamos criar um **Procfile** para dizer ao Heroku como desejamos que nosso aplicativo seja executado. No diretório do projeto (o mesmo com **package.json**) crie um arquivo chamado **Procfile** e coloque o seguinte

- `web: node server.js`
- Em seguida, execute o comando `ng build --prod --aot` para construir a build da aplicação

8) Associando o repositório local com o [Heroku](#):

- Acesse o terminal (no Windows o CMD) na pasta do projeto:

- o `git init`
 - o `heroku git:remote -a sba7-app`
 - o `git add .`
 - o `git commit -am "Meu Front end"`

- Envie o projeto para o **Heroku**

- o `git push heroku master`