

Curso de Node.JS

Bluuweb

Vídeos:

<https://www.youtube.com/watch?v=mG4U9t5nWG8&list=PLPI81lqbj-4IEnmCXEJeEXPepr8gWtsl6>

GitHub:

<https://github.com/bluuweb/curso-node-youtube/tree/crud-mongo>

Documentação utilizada:

<https://bluuweb.github.io/node/>

Resumo feito por Roberto Pinheiro

Aula 01 - Fundamentos

O que é Node.js?

- Node.js é um ambiente em tempo de execução multiplataforma, de código aberto para a camada do servidor (mas não limitado a ele) baseado na linguagem de programação JavaScript.
- Ao contrário da maioria do código JavaScript, ele não é executado em um navegador, mas sim no servidor.
- É baseado no motor V8 do Google.
- Permite que você crie aplicativos que rodam no servidor.
- Geralmente você encontrará muitas informações sobre node.js vs php (para aplicativos da web)

Primeiros passos

- Instale o **Node** e o **Visual Studio Code**.

```
C:\bluuweb\nodejs>node -v  
v14.15.3
```

- Crie uma pasta chamada "**Fundamentos**"

```
C:\bluuweb\nodejs>md Fundamentos  
C:\bluuweb\nodejs>cd Fundamentos  
C:\bluuweb\nodejs\Fundamentos>
```

```
C:\bluuweb\nodejs\Fundamentos>code .
```

- Dentro dessa pasta adicione um arquivo chamado **app.js**:



app.js

```
const frutas = ['banana', 'laranja', 'pera', 'banana'];

frutas.forEach(fruta => {
  console.log(fruta);
})
```

node app.js

```
C:\bluuweb\nodejs\Fundamentos>node app.js
banana
laranja
pera
banana
```

app.js

```
const frutas = ['banana', 'laranja', 'pera', 'banana'];

frutas.forEach(fruta => {
  console.count(fruta);
})
```

node app.js

```
C:\bluuweb\nodejs\Fundamentos>node app.js
banana: 1
laranja: 1
pera: 1
banana: 2
```

Aula 02 - Módulos - Exports e Require

Node.js tem um sistema de módulo integrado. Um arquivo Node.js pode importar a funcionalidade exposta por outros arquivos Node.js.

Exportando

- Adicione um arquivo chamado **frutas.js**:



frutas.js

```
const frutas = ['banana', 'laranja', 'pera', 'banana'];
```

```
module.exports = frutas;
```

Importando

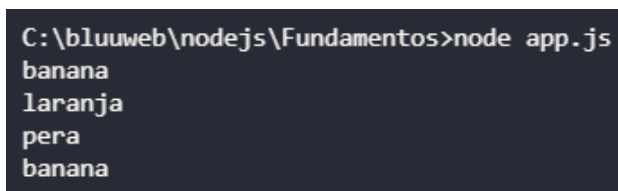
Quando você quiser importar algo, use:

app.js

```
const frutas = require('./frutas');
```

```
frutas.forEach(fruta => {  
  console.log(fruta);  
})
```

```
node app.js
```

A screenshot of a terminal window. The prompt is 'C:\bluuweb\nodejs\Fundamentos>'. The command 'node app.js' has been entered. The output of the command is displayed on the next four lines: 'banana', 'laranja', 'pera', and 'banana'.

Exportando mais de uma constante

frutas.js

```
const frutas = ['banana', 'laranja', 'pera', 'banana'];  
const valor = 1000;  
  
module.exports = {  
  frutas: frutas,  
  valor  
};
```

app.js

```
const {frutas, valor} = require('./frutas');  
  
frutas.forEach(fruta => {  
  console.log(fruta);  
})  
  
console.log(valor);
```

node app.js

```
C:\bluuweb\nodejs\Fundamentos>node app.js  
banana  
laranja  
pera  
banana  
1000
```

Aula 03 - package.json e npm

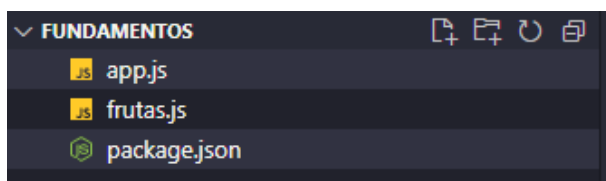
Gerando package.json

`npm init -y`

Vamos criar um arquivo que terá informações sobre o nosso projeto, o mais relevante no momento serão suas dependências e scripts

```
C:\bluuweb\nodejs\Fundamentos>npm init -y
Wrote to C:\bluuweb\nodejs\Fundamentos\package.json:

{
  "name": "Fundamentos",
  "version": "1.0.0",
  "description": "",
  "main": "app.js",
  "scripts": {
    "test": "echo \"Error: no test specified\" && exit 1"
  },
  "keywords": [],
  "author": "",
  "license": "ISC"
}
```



package.json

```
{
  "name": "Fundamentos",
  "version": "1.0.0",
  "description": "",
  "main": "app.js",
  "scripts": {
    "test": "echo \"Error: no test specified\" && exit 1"
  },
  "keywords": [],
  "author": "",
  "license": "ISC"
}
```

Exemplo de scripts

```
"scripts": {
  "dev": "webpack-dev-server --inline --progress --config build/webpack.dev.conf.js",
  "start": "npm run dev",
  "unit": "jest --config test/unit/jest.conf.js --coverage",
  "test": "npm run unit",
  "lint": "eslint --ext .js,.vue src test/unit",
  "build": "node build/build.js"
}
```

Esses scripts são aplicativos de linha de comando. Você pode executá-los chamando **npm run XXXX**, onde "XXXX" é o nome do comando. Exemplo: **npm run dev**

Você pode usar o nome que quiser para um comando e os scripts podem fazer literalmente tudo o que você quiser.

Dependências e devDependencies

```
"dependencies": {
  "cowsay": "^1.4.0"
},
"devDependencies": {
  "autoprefixer": "^7.1.2",
  "babel-core": "^6.22.1"
}
```

- Aqui está a lista de pacotes instalados por meio do npm.
- **devDependencies** refere-se a pacotes que não são necessários para a produção, eles devem ser instalados apenas em uma máquina de desenvolvimento, eles não são necessários para executar o código na produção.

NPM (Node Package Manager)

npm é o gerenciador de pacotes padrão para Node.js.

Atualizando pacotes

npm update

Atualizando pacotes: o npm pesquisará todos os pacotes por uma versão mais recente que satisfaça suas restrições de versão.

Onde o npm instala os pacotes?

Ao instalar um pacote com npm, você pode realizar 2 tipos de instalação:

- **Local:** npm install <package-name> (a pasta **node_modules** é criada)
- **Global:** npm install -g <package-name>

Geralmente utilizamos instalação global para os CLI, por exemplo o cli de **vue.js**, outra opção são para os pacotes frequentes como **nodemon**.

Exemplo de instalação de um pacote (cowsay)

<https://www.npmjs.com/package/cowsay>

cowsay



Instalando o pacote cowsay de forma local

Install

```
> npm i cowsay
```

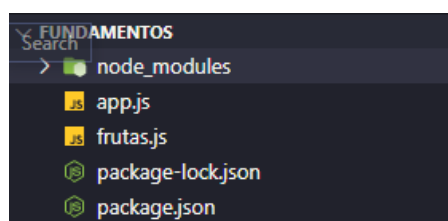
npm i cowsay

```
C:\bluuweb\nodejs\Fundamentos>npm i cowsay
npm notice created a lockfile as package-lock.json. You should commit this file.
npm WARN Fundamentos@1.0.0 No description
npm WARN Fundamentos@1.0.0 No repository field.

+ cowsay@1.5.0
added 41 packages from 10 contributors and audited 41 packages in 21.077s

3 packages are looking for funding
  run `npm fund` for details

found 0 vulnerabilities
```



package.json

```
{
  "name": "Fundamentos",
  "version": "1.0.0",
  "description": "",
  "main": "app.js",
  "scripts": {
    "test": "echo \"Error: no test specified\" && exit 1"
  },
  "keywords": [],
  "author": "",
  "license": "ISC",
  "dependencies": {
    "cowsay": "^1.5.0"
  }
}
```

- Uso:

Usage as a module

cowsay can be used as any other npm dependency

```
var cowsay = require("cowsay");

console.log(cowsay.say({
  text : "I'm a moooodule",
  e : "oO",
  T : "U "
}));

// or cowsay.think()
```

app.js

```
const cowsay = require("cowsay");
```

```
console.log(cowsay.say({
  text : "I'm a moooodule",
  e : "oO",
  T : "U "
}));
```

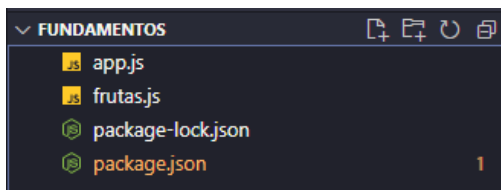
node app.js

```
C:\bluuweb\nodejs\Fundamentos>node app.js

< I'm a moooodule >
-----
      \   ^__^
       \  (oo)\_______
          (__)\       )\/\
             U   ||---w   ||
                 ||     ||
```


A pasta **node_modules** nunca deve ser enviada ao servidor.

- Apague a pasta **node_modules**.



- Para recuperá-la (com todas as dependências listadas no arquivo **package.json**), basta rodar o comando:

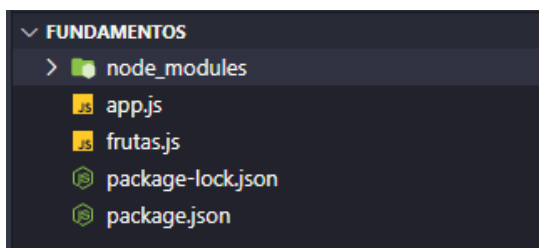
npm install

```
C:\bluuweb\nodejs\Fundamentos>npm install
npm WARN Fundamentos@1.0.0 No description
npm WARN Fundamentos@1.0.0 No repository field.

added 41 packages from 10 contributors and audited 41 packages in 5.693s

3 packages are looking for funding
  run `npm fund` for details

found 0 vulnerabilities
```



Instalando o pacote cowsay de forma global

- É instalado no computador e não especificamente no projeto.

Install

```
npm install -g cowsay
```

Em qualquer pasta, entre com o comando:

npm install -g cowsay

```
C:\Users\bet01>npm install -g cowsay
C:\Users\bet01\AppData\Roaming\npm\cowsay -> C:\Users\bet01\AppData\Roaming\npm\node_modules\cowsay\cli.js
C:\Users\bet01\AppData\Roaming\npm\cowthink -> C:\Users\bet01\AppData\Roaming\npm\node_modules\cowsay\cli.js
+ cowsay@1.5.0
added 41 packages from 10 contributors in 7.583s
```

cowsay JavaScript FTW!

```
C:\Users\beto1>cowsay JavaScript FTW!  
  
< JavaScript FTW! >  
-----  
      \      ^__^  
       \      (oo)\_____  
          (__)\\       )\\/\  
              ||----w |  
              ||     ||
```

npx

- Muito usado em React e Vue.JS
- Executa um código sem a necessidade de instalar o pacote de maneira global.
- Por exemplo:

`npx cowsay "Olá mundo!"`

```
C:\Users\beto1>npx cowsay "Olá mundo!"  
  
< Olá mundo! >  
-----  
      \      ^__^  
       \      (oo)\_____  
          (__)\\       )\\/\  
              ||----w |  
              ||     ||
```

Aula 04 - Fundamentos de servidor HTTP

Servidor HTTP?

Hypertext Transfer Protocol: O Hypertext Transfer Protocol é o protocolo de comunicação que permite a transferência de informações na World Wide Web.

- Troca de informações entre cliente e servidor.
- O servidor espera por alguma solicitação HTTP do cliente e fornece uma resposta.

Quando visitamos um site, fazemos uma solicitação HTTP GET e o servidor nos retorna, por exemplo, um index.html com o site.

Vamos configurar essas respostas em nosso "servidor web ou servidor http" com node.js

Como começo com o Node.js depois de instalá-lo?

O Node.js vem com alguns módulos que não precisam de instalação, um deles é o **http**.

Depois de instalar o Node.js, vamos construir nosso primeiro servidor web. Crie um arquivo chamado **app.js** contendo o seguinte conteúdo:

```
const http = require("http");

const port = process.env.PORT;

const server = http.createServer((req, res) => {
  res.statusCode = 200;
  res.setHeader("Content-Type", "text/html");
  res.end("<h1>Hello, World!</h1>");
});

server.listen(port, () => {
  console.log(`Server running at port ${port}`);
});
```

Agora, execute o seu servidor web usando node app.js. Visite <http://localhost:3000> e você verá uma mensagem dizendo "**Hello World**".

Ctrl + C para encerrar o servidor

Verbos ou métodos HTTP

O HTTP define um conjunto de métodos de solicitação para indicar a ação a ser executada para um determinado recurso.

Os mais populares são: GET, POST, PUT, DELETE.

app.js

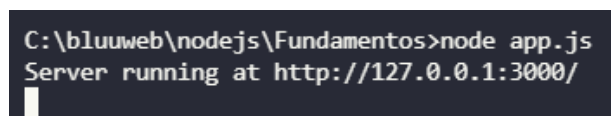
```
const http = require('http');

const hostname = '127.0.0.1';
const port = 3000;

const server = http.createServer((req, res) => {
  res.statusCode = 200;
  res.setHeader('Content-Type', 'text/plain');
  res.end('Hello World');
});

server.listen(port, hostname, () => {
  console.log(`Server running at http://${hostname}:${port}/`);
});
```

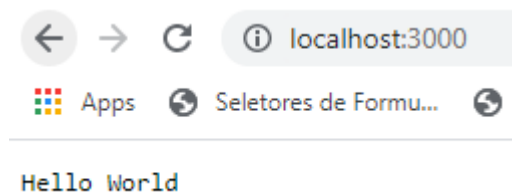
node app.js



```
C:\bluuweb\nodejs\Fundamentos>node app.js
Server running at http://127.0.0.1:3000/
```

- No browser:

localhost:3000



Faça a seguinte alteração na mensagem, e em seguida a salve:

app.js

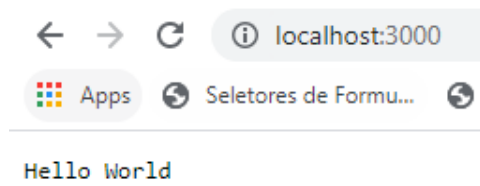
```
const http = require('http');

const hostname = '127.0.0.1';
const port = 3000;

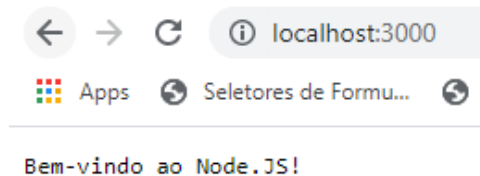
const server = http.createServer((req, res) => {
  res.statusCode = 200;
  res.setHeader('Content-Type', 'text/plain');
  res.end('Bem-vindo ao Node.JS!');
});

server.listen(port, hostname, () => {
  console.log(`Server running at http://${hostname}:${port}/`);
});
```

- Recarregue o browser e perceba que a alteração não foi refletida:



- Para que isso ocorra é necessário, interromper o servidor e em seguida reiniciá-lo:



- Para evitar que tenhamos que fazer isso cada vez que for feita uma alteração no projeto devemos instalar e utilizar um pacote chamado **nodemon**

Essa ferramenta é instalada por meio do npm e nos ajuda a ouvir as alterações em nossa configuração node.js e reiniciar o servidor automaticamente.

Instalando o pacote nodemon

npm install --save nodemon -D

```
C:\bluuweb\nodejs\Fundamentos>npm install --save nodemon -D

> nodemon@2.0.7 postinstall C:\bluuweb\nodejs\Fundamentos\node_modules\nodemon
> node bin/postinstall || exit 0

npm WARN optional SKIPPING OPTIONAL DEPENDENCY: fsevents@~2.3.1 (node_modules\chokidar\node_modules\fsevents):
npm WARN notsup SKIPPING OPTIONAL DEPENDENCY: Unsupported platform for fsevents@2.3.2: wanted {"os":"darwin","arch":"any"} (current: {"os":"win32","arch":"x64"})
npm WARN Fundamentos@1.0.0 No description
npm WARN Fundamentos@1.0.0 No repository field.

+ nodemon@2.0.7
added 117 packages from 51 contributors and audited 159 packages in 38.532s

13 packages are looking for funding
  run `npm fund` for details

found 0 vulnerabilities
```

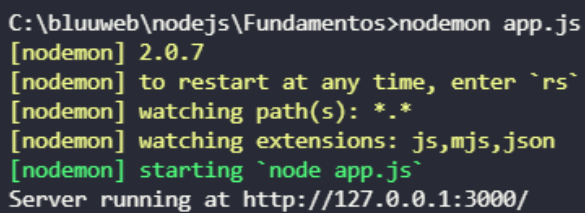
Obs: **-D** informa que o nodemon será utilizado apenas em ambiente de desenvolvimento.

package.json

```
{
  "name": "Fundamentos",
  "version": "1.0.0",
  "description": "",
  "main": "app.js",
  "scripts": {
    "test": "echo \"Error: no test specified\" && exit 1"
  },
  "keywords": [],
  "author": "",
  "license": "ISC",
  "dependencies": {
    "cowsay": "^1.5.0"
  },
  "devDependencies": {
    "nodemon": "^2.0.7"
  }
}
```

- Agora, ao invés de utilizarmos `node app.js`, usamos `nodemon app.js`

nodemon app.js



```
C:\bluweb\nodejs\Fundamentos>nodemon app.js
[nodemon] 2.0.7
[nodemon] to restart at any time, enter `rs`
[nodemon] watching path(s): *.*
[nodemon] watching extensions: js,mjs,json
[nodemon] starting `node app.js`
Server running at http://127.0.0.1:3000/
```

- Faça uma alteração na mensagem:

app.js

```
const http = require('http');

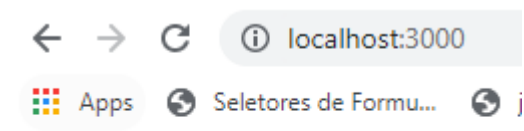
const hostname = '127.0.0.1';
const port = 3000;

const server = http.createServer((req, res) => {
  res.statusCode = 200;
  res.setHeader('Content-Type', 'text/plain');
  res.end('Seja bem-vindo ao Node.JS!');
});

server.listen(port, hostname, () => {
  console.log(`Server running at http://${hostname}:${port}/`);
});
```

- Depois de gravar o arquivo, a alteração será refletida no browser ao recarregarmos:

```
C:\bluuweb\nodejs\Fundamentos>nodemon app.js
[nodemon] 2.0.7
[nodemon] to restart at any time, enter `rs`
[nodemon] watching path(s): *.*
[nodemon] watching extensions: js,mjs,json
[nodemon] starting `node app.js`
Server running at http://127.0.0.1:3000/
[nodemon] restarting due to changes...
[nodemon] restarting due to changes...
[nodemon] starting `node app.js`
Server running at http://127.0.0.1:3000/
█
```



Seja bem-vindo ao Node.JS!

Aula 05 - Introdução ao Express

Express é uma estrutura de aplicativo da web Node.js mínima e flexível que fornece um conjunto robusto de recursos para aplicativos da web e móveis (Framework).

Com milhares de métodos de middleware e utilitários HTTP à sua disposição, construir uma API robusta é rápido e fácil.

- Em `C:\bluuweb\nodejs` crie uma pasta chamada "`rotas_basico`"

```
C:\bluuweb\nodejs>md rotas_basico

C:\bluuweb\nodejs>dir
O volume na unidade C é W10-195
O Número de Série do Volume é 6047-7D0C

Pasta de C:\bluuweb\nodejs

02/06/2021  04:00    <DIR>          .
02/06/2021  04:00    <DIR>          ..
02/06/2021  03:17    <DIR>          Fundamentos
02/06/2021  04:00    <DIR>          rotas_basico
               0 arquivo(s)                0 bytes
               4 pasta(s) 111.559.278.592 bytes disponíveis
```

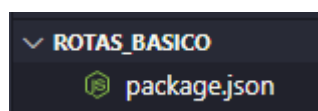
Gerando package.json

`npm init -y`

```
C:\bluuweb\nodejs\rotas_basico>npm init -y
Wrote to C:\bluuweb\nodejs\rotas_basico\package.json:

{
  "name": "rotas_basico",
  "version": "1.0.0",
  "description": "",
  "main": "index.js",
  "scripts": {
    "test": "echo \"Error: no test specified\" && exit 1"
  },
  "keywords": [],
  "author": "",
  "license": "ISC"
}
```

`code .`



Instalando o nodemon

`npm install --save nodemon -D`

```
C:\bluuweb\nodejs\rotas_basico>npm install --save nodemon -D

> nodemon@2.0.7 postinstall C:\bluuweb\nodejs\rotas_basico\node_modules\nodemon
> node bin/postinstall || exit 0

npm notice created a lockfile as package-lock.json. You should commit this file.
npm WARN optional SKIPPING OPTIONAL DEPENDENCY: fsevents@~2.3.1 (node_modules\chokidar\node_modules\fsevents):
npm WARN notsup SKIPPING OPTIONAL DEPENDENCY: Unsupported platform for fsevents@2.3.2: wanted {"os":"darwin","arch":"any"} (current: {"os":"win32","arch":"x64"})
npm WARN rotas_basico@1.0.0 No description
npm WARN rotas_basico@1.0.0 No repository field.

+ nodemon@2.0.7
added 119 packages from 53 contributors and audited 120 packages in 22.884s

11 packages are looking for funding
  run `npm fund` for details

found 0 vulnerabilities
```

Instalando o Express

`npm install --save express`

```
C:\bluuweb\nodejs\rotas_basico>npm install --save express

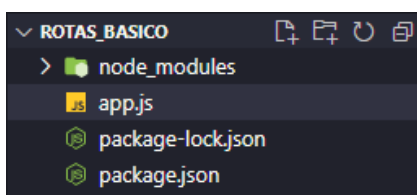
npm WARN rotas_basico@1.0.0 No description
npm WARN rotas_basico@1.0.0 No repository field.
npm WARN optional SKIPPING OPTIONAL DEPENDENCY: fsevents@2.3.2 (node_modules\fsevents):
npm WARN notsup SKIPPING OPTIONAL DEPENDENCY: Unsupported platform for fsevents@2.3.2: wanted {"os":"darwin","arch":"any"} (current: {"os":"win32","arch":"x64"})

+ express@4.17.1
added 56 packages from 37 contributors and audited 176 packages in 14.246s

11 packages are looking for funding
  run `npm fund` for details

found 0 vulnerabilities
```

- Na pasta do projeto adicione um arquivo chamado **app.js**:



app.js

```
const express = require('express');
const app = express();

const port = 3000;

app.get('/', (req, res) => {
  res.send('Hello World!');
});

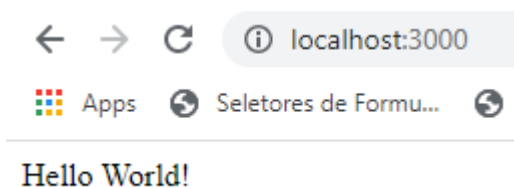
app.listen(port, () =>{
  console.log('Servidor rodando na porta ' + port);
});
```

nodemon app.js

```
C:\bluuweb\nodejs\rotas_basico>nodemon app.js
[nodemon] 2.0.7
[nodemon] to restart at any time, enter `rs`
[nodemon] watching path(s): *.*
[nodemon] watching extensions: js,mjs,json
[nodemon] starting `node app.js`
Servidor rodando na porta 3000
```

- No browser:

localhost:3000



app.js

```
const express = require('express');
const app = express();
```

```
const port = 3000;
```

```
// Rotas
```

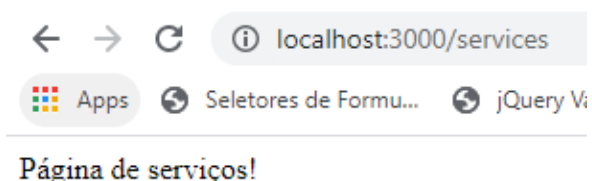
```
app.get('/', (req, res) => {
  res.send('Hello World!');
});
```

```
app.get('/services', (req, res) => {
  res.send('Página de serviços!');
});
```

```
app.listen(port, () =>{
  console.log('Servidor rodando na porta ' + port);
});
```

- No browser:

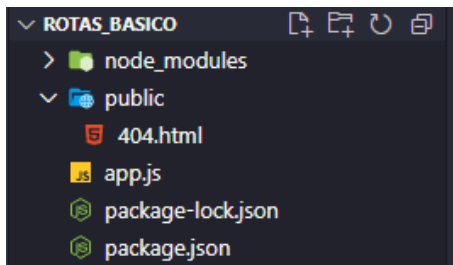
localhost:3000/services



Arquivos estáticos

Para servir arquivos estáticos, como imagens, arquivos CSS e arquivos JavaScript, use a função de middleware integrada `express.static` do Express.

- Na pasta inicial do projeto adicione uma subpasta chamada "**public**" e dentro dela adicione um arquivo chamado "**404.html**":



Configurando um middleware

Em palavras simples, middleware é uma ação executada antes de nossa função de rota.

app.js

```
const express = require('express');
const app = express();
const port = 3000;

// Rotas

app.get('/', (req, res) => {
  res.send('Hello World!');
});

app.get('/services', (req, res) => {
  res.send('Página de serviços!');
});

app.use(express.static(__dirname + "/public"));

app.listen(port, () =>{
  console.log('Servidor rodando na porta ' + port);
});
```

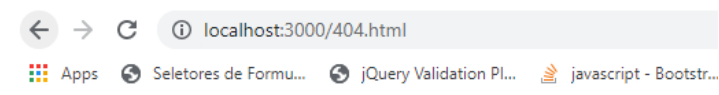
`__dirname` é o caminho de acordo com a máquina onde o código é executado:

public\404.html

```
<!DOCTYPE html>
<html lang="pt-br">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Erro 404</title>
</head>
<body>
  <h1>Erro 404 - Página não encontrada!</h1>
</body>
</html>
```

- No browser:

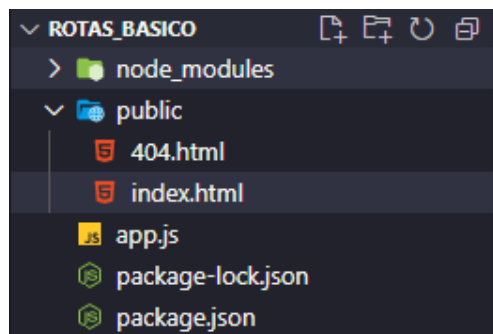
<http://localhost:3000/404.html>



Erro 404 - Página não encontrada!

Configurando página inicial como um arquivo estático

- Dentro da pasta **public** adicione um arquivo chamado **index.html**:



app.js

```
const express = require('express');
const app = express();

const port = 3000;

app.use(express.static(__dirname + "/public"));

// Rotas

app.get('/', (req, res) => {
  res.send('Hello World!');
});

app.get('/services', (req, res) => {
  res.send('Página de serviços!');
});

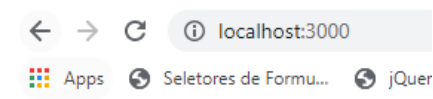
app.listen(port, () =>{
  console.log('Servidor rodando na porta ' + port);
});
```

public\index.html

```
<!DOCTYPE html>
<html lang="pt-br">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Home</title>
</head>
<body>
  <h1>Página inicial</h1>
</body>
</html>
```

- No browser:

localhost:3000



Página inicial

Tratando rota não encontrada

app.js

```
const express = require('express');
const app = express();

const port = 3000;

app.use(express.static(__dirname + "/public"));

// Rotas

app.get('/', (req, res) => {
  res.send('Hello World!');
});

app.get('/services', (req, res) => {
  res.send('Página de serviços!');
});

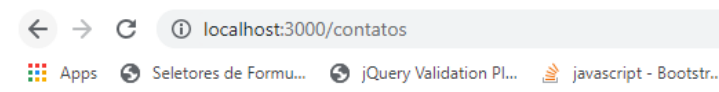
// Rota não encontrada

app.use((req, res, next) => {
  res.status(404).sendFile(__dirname + "/public/404.html");
});

app.listen(port, () =>{
  console.log('Servidor rodando na porta ' + port);
});
```

- No browser:

<http://localhost:3000/contatos>



Erro 404 - Página não encontrada!

Aula 06 - Template Engines (EJS) com Express

- Em `C:\bluuweb\nodejs` crie uma pasta chamada "`views_deploy_router`"

```
C:\bluuweb\nodejs>md views_deploy_router
```

```
C:\bluuweb\nodejs>dir
O volume na unidade C é W10-195
O Número de Série do Volume é 6047-7D0C

Pasta de C:\bluuweb\nodejs

02/06/2021  09:17    <DIR>          .
02/06/2021  09:17    <DIR>          ..
02/06/2021  03:17    <DIR>          Fundamentos
02/06/2021  04:12    <DIR>          rotas_basico
02/06/2021  09:32    <DIR>          views_deploy_router
               0 arquivo(s)                0 bytes
               5 pasta(s)  111.303.753.728 bytes disponíveis
```

```
C:\bluuweb\nodejs>cd views_deploy_router
```

```
C:\bluuweb\nodejs\views_deploy_router>
```

Gerando package.json

`npm init -y`

```
C:\bluuweb\nodejs\views_deploy_router>npm init -y
Wrote to C:\bluuweb\nodejs\views_deploy_router\package.json:

{
  "name": "views_deploy_router",
  "version": "1.0.0",
  "description": "",
  "main": "index.js",
  "scripts": {
    "test": "echo \"Error: no test specified\" && exit 1"
  },
  "keywords": [],
  "author": "",
  "license": "ISC"
}
```

code .

```
✓ VIEWS_DEPLOY_ROUTER
  package.json
```

Instalando o nodemon

npm install --save nodemon -D

```
C:\bluuweb\nodejs\views_deploy_router>npm install --save nodemon -D

> nodemon@2.0.7 postinstall C:\bluuweb\nodejs\views_deploy_router\node_modules\nodemon
> node bin/postinstall || exit 0

npm notice created a lockfile as package-lock.json. You should commit this file.
npm WARN optional SKIPPING OPTIONAL DEPENDENCY: fsevents@~2.3.1 (node_modules\chokidar\node_modules\fsevents):
npm WARN notsup SKIPPING OPTIONAL DEPENDENCY: Unsupported platform for fsevents@2.3.2: wanted {"os":"darwin","arch":"any"} (current: {"os":"win32","arch":"x64"})
npm WARN views_deploy_router@1.0.0 No description
npm WARN views_deploy_router@1.0.0 No repository field.

+ nodemon@2.0.7
added 119 packages from 53 contributors and audited 120 packages in 40.144s

11 packages are looking for funding
  run `npm fund` for details

found 0 vulnerabilities
```

Instalando o Express

npm install --save express

```
C:\bluuweb\nodejs\views_deploy_router>npm install --save express

npm WARN views_deploy_router@1.0.0 No description
npm WARN views_deploy_router@1.0.0 No repository field.
npm WARN optional SKIPPING OPTIONAL DEPENDENCY: fsevents@2.3.2 (node_modules\fsevents):
npm WARN notsup SKIPPING OPTIONAL DEPENDENCY: Unsupported platform for fsevents@2.3.2: wanted {"os":"darwin","arch":"any"} (current: {"os":"win32","arch":"x64"})

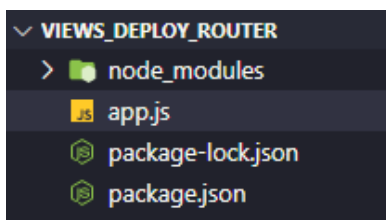
+ express@4.17.1
added 56 packages from 37 contributors and audited 176 packages in 18.938s

11 packages are looking for funding
  run `npm fund` for details

found 0 vulnerabilities

C:\bluuweb\nodejs\views_deploy_router>
```

- Na pasta do projeto adicione um arquivo chamado **app.js**:



app.js

```
const express = require('express');
const app = express();

const port = 3000;

app.get('/', (req, res) => {
  res.send('Hello World!');
});

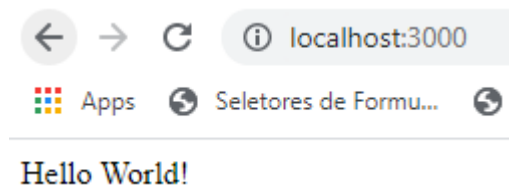
app.listen(port, () => {
  console.log('Servidor rodando na porta ' + port);
});
```

nodemon app.js

```
C:\bluuweb\nodejs\views_deploy_router>nodemon app.js
[nodemon] 2.0.7
[nodemon] to restart at any time, enter `rs`
[nodemon] watching path(s): *.*
[nodemon] watching extensions: js,mjs,json
[nodemon] starting `node app.js`
Servidor rodando na porta 3000
█
```

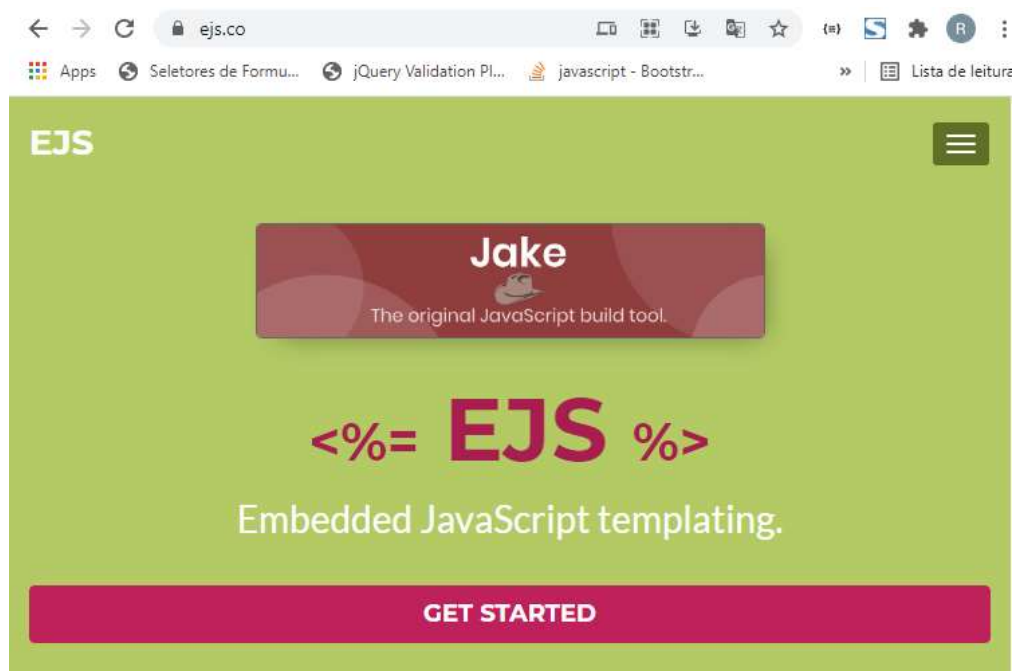
- No browser:

localhost:3000

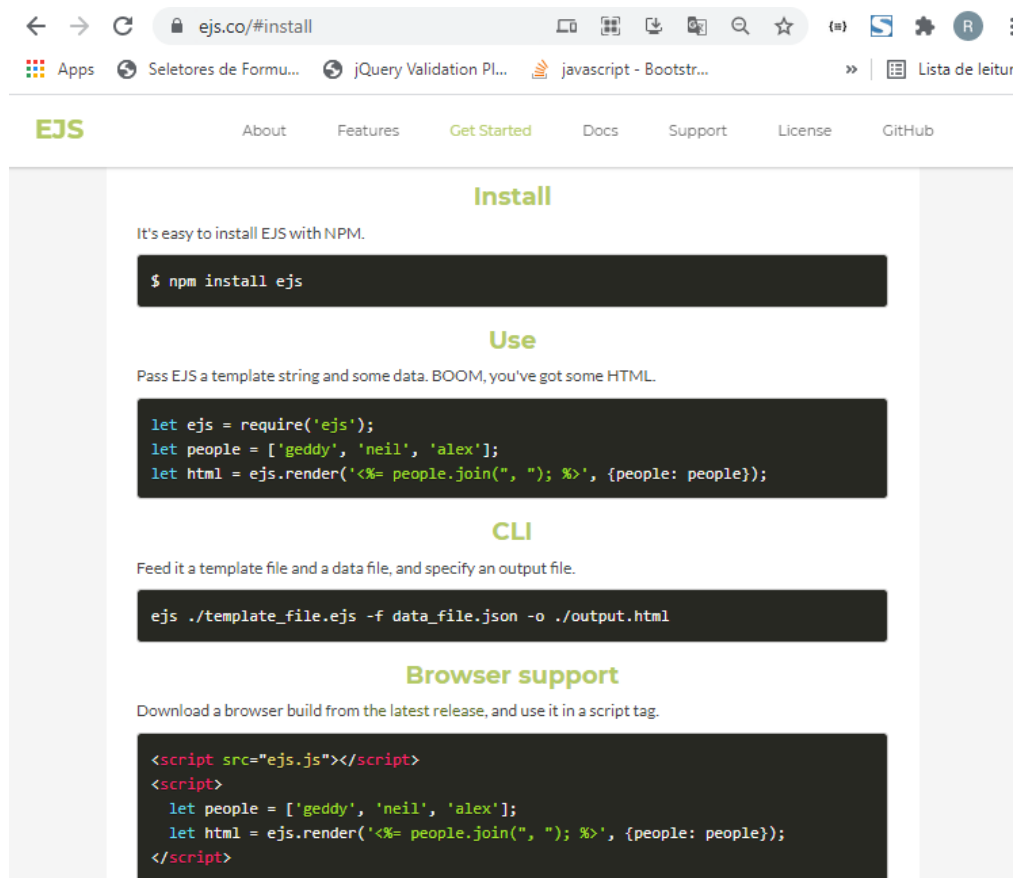


ejs

<https://ejs.co/>



<https://ejs.co/#install>



Instalando EJS

npm install --save ejs

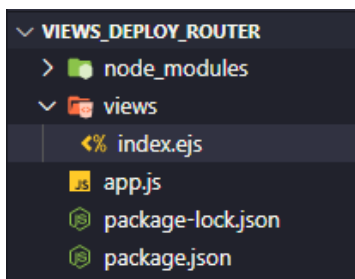
```
C:\blunweb\nodejs\views_deploy_router>npm install --save ejs
npm WARN views_deploy_router@1.0.0 No description
npm WARN views_deploy_router@1.0.0 No repository field.
npm WARN optional SKIPPING OPTIONAL DEPENDENCY: fsevents@2.3.2 (node_modules\fsevents):
npm WARN notsup SKIPPING OPTIONAL DEPENDENCY: Unsupported platform for fsevents@2.3.2: wanted {"os":"darwin","arch":"any"} (current: {"os":"win32","arch":"x64"})

+ ejs@3.1.6
added 9 packages from 5 contributors and audited 185 packages in 6.795s

11 packages are looking for funding
  run `npm fund` for details

found 0 vulnerabilities
```

- Na pasta do projeto, crie uma subpasta chamada "views". Dentro dela adicione um arquivo chamado "index.ejs"



views\index.ejs

```
<!DOCTYPE html>
<html lang="pt-br">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title><%= titulo %></title>
</head>
<body>
  <h1><%= conteudo %></h1>
</body>
</html>
```

app.js

```
const express = require('express');
const app = express();

const port = 3000;

// motor de templates
app.set('view engine', 'ejs');
app.set('views', __dirname + '/views');

app.use(express.static(__dirname + "/public"));

// Rotas

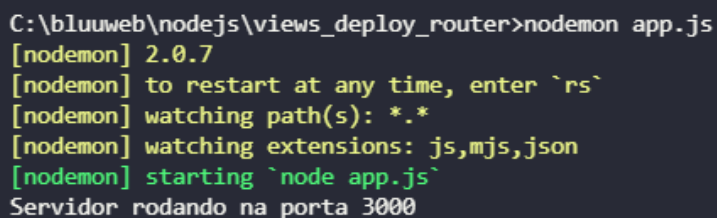
app.get('/', (req, res) => {
  res.render("index", {titulo: "Template EJS", conteudo: "Título dinâmico"});
});

app.get('/services', (req, res) => {
  res.send('Página de serviços!');
});

app.get('/contact', (req, res) => {
  res.send('Página de contato!');
});

app.listen(port, () =>{
  console.log('Servidor rodando na porta ' + port);
});
```

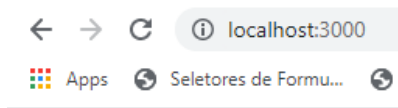
nodemon app.js

A terminal window with a dark background and light-colored text. The prompt is 'C:\bluuweb\nodejs\views_deploy_router>'. The command 'nodemon app.js' has been executed. The output shows 'nodemon 2.0.7' in green, followed by instructions to restart with 'rs', the watched path '.*', the watched extensions 'js,mjs,json', and the command 'starting node app.js' in green. Finally, it displays 'Servidor rodando na porta 3000' in white, with a white cursor line below it.

```
C:\bluuweb\nodejs\views_deploy_router>nodemon app.js
[nodemon] 2.0.7
[nodemon] to restart at any time, enter `rs`
[nodemon] watching path(s): *.*
[nodemon] watching extensions: js,mjs,json
[nodemon] starting `node app.js`
Servidor rodando na porta 3000
█
```

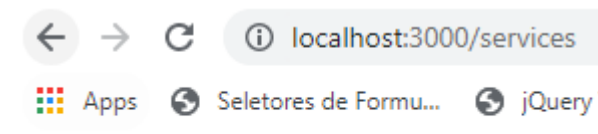
- No browser:

<http://localhost:3000/>



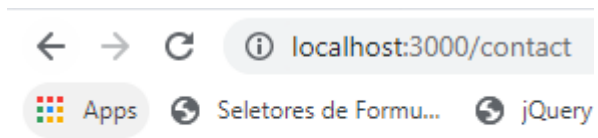
Título dinâmico

<http://localhost:3000/services>



Página de serviços!

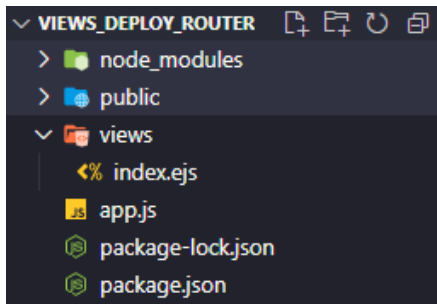
<http://localhost:3000/contact>



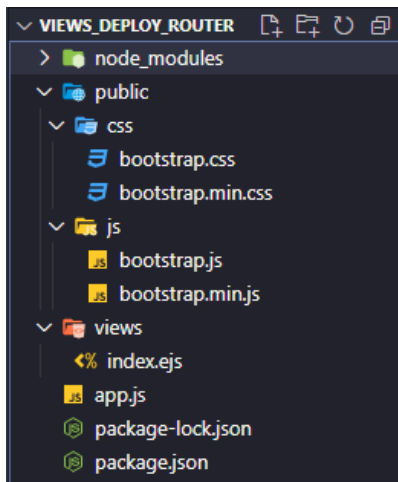
Página de contato!

Aula 07 - Navbar Responsive com Bootstrap 4

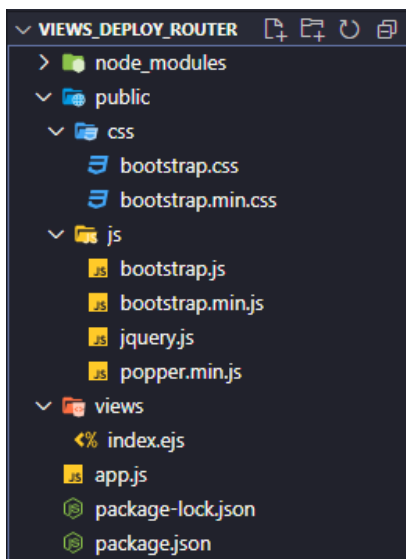
- Dentro da pasta do projeto, adicione uma semipasta chamada "public":



- Baixe o **bootstrap** e instale as pastas **css** e **js** dentro da pasta **public**. Mantenha apenas os arquivos exibidos na figura abaixo, os demais apague.

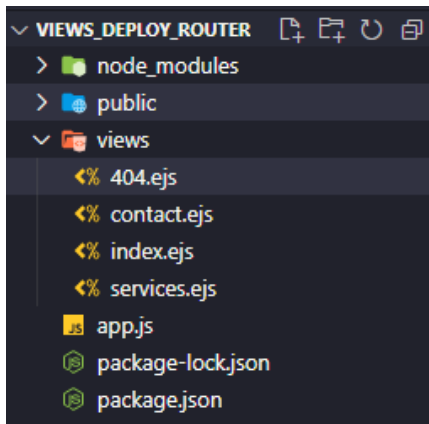


- Baixe o **jquery 3.2.1.js** e o adicione na pasta **js**.
- Baixe o arquivo **popper.min.js** e também adicione na pasta **js**.



- Dentro da pasta **views**, adicione os seguintes arquivos:

services.ejs
contact.ejs
404.ejs



app.js

```
const express = require('express');
const app = express();

const port = 3000;

// motor de templates
app.set('view engine', 'ejs');
app.set('views', __dirname + '/views');

app.use(express.static(__dirname + "/public"));

// Rotas

app.get('/', (req, res) => {
  // res.send('Hello World!');
  res.render("index", {bodyTitle: "Página inicial", bodyContent: "Seja bem-vindo ao nosso site!"});
});

app.get('/services', (req, res) => {
  // res.send('Página de serviços!');
  res.render("services", {bodyTitle: "Página de Serviços", bodyContent: "Oferecemos os melhores serviços aos nossos clientes!"});
});

app.get('/contact', (req, res) => {
  // res.send('Página de contato!');
  res.render("contact", {bodyTitle: "Página de Contato", bodyContent: "Entre em contato conosco!"});
});

// Rota não encontrada

app.use((req, res, next) => {
  // res.status(404).sendFile(__dirname + "/public/404.html");
  res.status(404).render("404", {bodyTitle: "Página não encontrada!", bodyContent: "Essa página não existe!"});
});
```

```
app.listen(port, () =>{
  console.log('Servidor rodando na porta ' + port);
});
```

views\index.ejs

```
<!DOCTYPE html>
<html lang="pt-br">

<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>
    <%= bodyTitle %>
  </title>
</head>

<body>
  <h1>
    <%= bodyContent %>
  </h1>
</body>

</html>
```

views\services.ejs

```
<!DOCTYPE html>
<html lang="pt-br">

<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>
    <%= bodyTitle %>
  </title>
</head>

<body>
  <h1>
    <%= bodyContent %>
  </h1>
</body>

</html>
```


views\contact.ejs

```
<!DOCTYPE html>
<html lang="pt-br">

<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>
    <%= bodyTitle %>
  </title>
</head>

<body>
  <h1>
    <%= bodyContent %>
  </h1>
</body>

</html>
```

views\404.ejs

```
<!DOCTYPE html>
<html lang="pt-br">

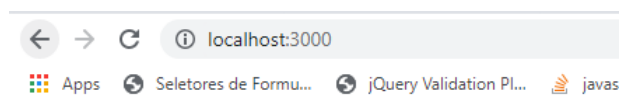
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>
    <%= bodyTitle %>
  </title>
</head>

<body>
  <h1>
    <%= bodyContent %>
  </h1>
</body>

</html>
```

- No browser:

localhost:3000

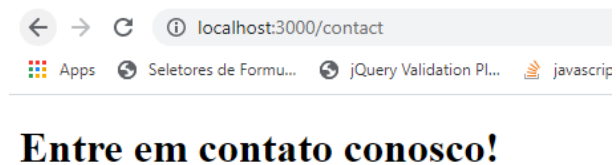


Seja bem-vindo ao nosso site!

localhost:3000/services



localhost:3000/contact



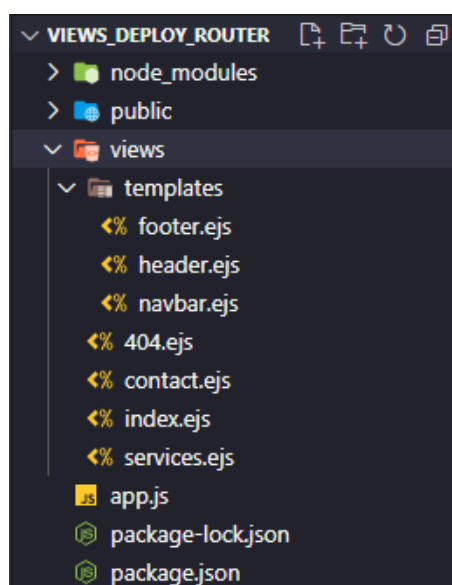
localhost:3000/contato



Usando templates

- Dentro da pasta **views** adicione uma pasta chamada "**templates**" e dentro dela adicione os arquivos:

- header.ejs
- footer.ejs
- navbar.ejs



Templates

views\templates\header.ejs

```
<!DOCTYPE html>
<html lang="pt-br">

<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>
    <%= bodyTitle %>
  </title>
  <link rel="stylesheet" href="/css/bootstrap.css">
</head>

<body>
  <%- include("../navbar.ejs") %>
```

views\templates\footer.ejs

```
<div class="text-white bg-dark text-center p-3 mb-2 mt-1" role="alert">
  <a style="color: yellow" href="">Orion3</a> - By: Roberto Pinheiro -
  Copyright
  &copy;
  <?php echo date('Y'); ?> - Todos os direitos reservados.
</div>

<script src="/js/jquery.js"></script>
<script src="/js/popper.min.js"></script>
<script src="/js/bootstrap.min.js"></script>

</body>

</html>
```

views\templates\navbar.ejs

```
<nav class="navbar navbar-expand-md navbar-dark bg-dark">
  <a class="navbar-brand" href="/">Orion3</a>
  <button class="navbar-toggler" type="button" data-toggle="collapse" data-target="#navbarSupportedContent"
    aria-controls="navbarSupportedContent" aria-expanded="false" aria-label="Toggle navigation">
    <span class="navbar-toggler-icon"></span>
  </button>

  <div class="collapse navbar-collapse" id="navbarSupportedContent">
    <ul class="navbar-nav mr-auto">
      <li class="nav-item active">
        <a class="nav-link" href="/">Home <span class="sr-only">(current)</span></a>
      </li>
      <li class="nav-item">
        <a class="nav-link" href="/services">Serviços</a>
      </li>
      <li class="nav-item">
```

```

    <a class="nav-link" href="/contact">Contato</a>
  </li>
  <li class="nav-item dropdown">
    <a class="nav-link dropdown-toggle" href="#" id="navbarDropdown" role="button" data-toggle="dropdown"
      aria-haspopup="true" aria-expanded="false">
      Downloads
    </a>
    <div class="dropdown-menu" aria-labelledby="navbarDropdown">
      <a class="dropdown-item" href="#">Download 1</a>
      <a class="dropdown-item" href="#">Download 2</a>
      <div class="dropdown-divider"></div>
      <a class="dropdown-item" href="#">Outros downloads</a>
    </div>
  </li>
</ul>
<form class="form-inline my-2 my-lg-0">
  <input class="form-control mr-sm-2" type="search" placeholder="Search" aria-label="Search">
  <button class="btn btn-outline-success my-2 my-sm-0" type="submit">Search</button>
</form>
</div>
</nav>

```

- Altere as views:

Views

views\index.ejs

```

<%- include("templates/header", {title: "Home" }) %>

<div class="container mb-4 mt-2">
  <h1 class="text-primary"><%= bodyTitle %></h1>
  <h2><%= bodyContent %></h2>
</div>

<%- include("templates/footer") %>

```

views\services.ejs

```

<%- include("templates/header", {title: "Serviços"}) %>

<div class="container mb-4 mt-2">
  <h1 class="text-success"><%= bodyTitle %></h1>
  <h2><%= bodyContent %></h2>
</div>

<%- include("templates/footer") %>

```

views\contact.ejs

```
<%- include("templates/header", {title: "Contato" }) %>
```

```
<div class="container mb-4 mt-2">  
  <h1 class="text-info"><%= bodyTitle %></h1>  
  <h2><%= bodyContent %></h2>  
</div>
```

```
<%- include("templates/footer") %>
```

views\404.ejs

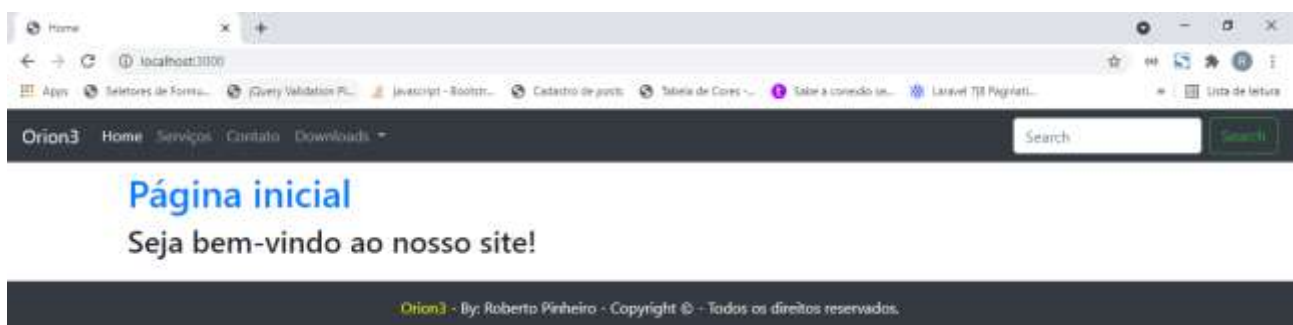
```
<%- include("templates/header", {title: "Erro 404" }) %>
```

```
<div class="container mb-4 mt-2">  
  <h1 class="text-danger"><%= bodyTitle %></h1>  
  <h2><%= bodyContent %></h2>  
</div>
```

```
<%- include("templates/footer") %>
```

- No browser:

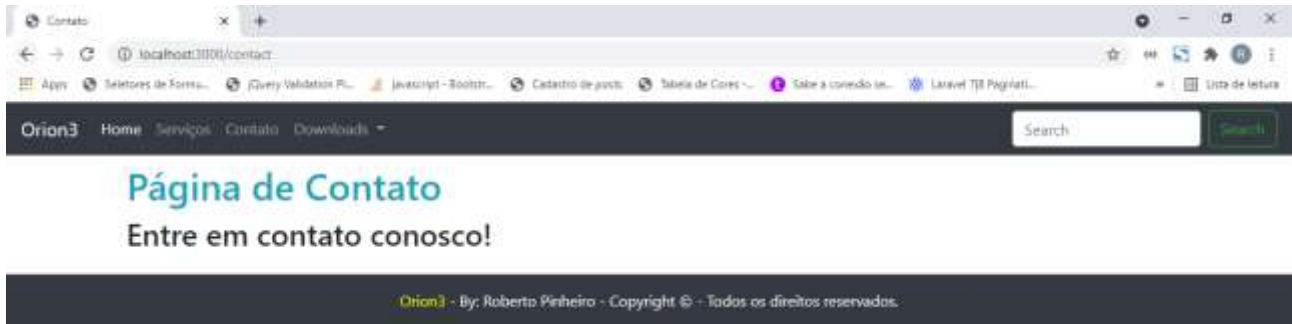
localhost:3000



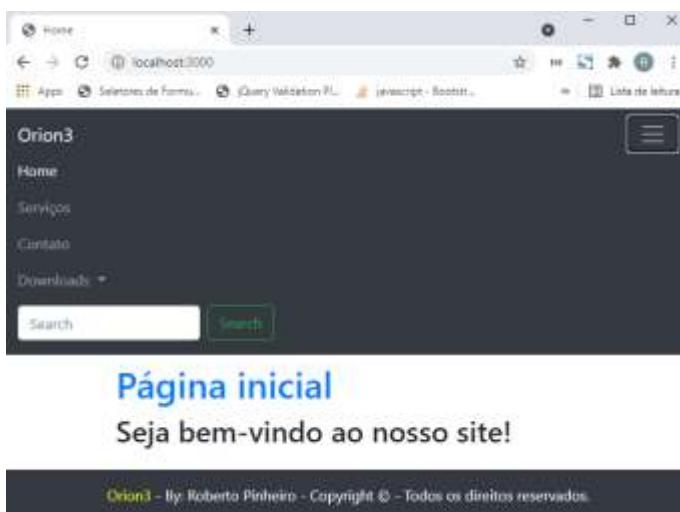
localhost:3000/services



localhost:3000/contact



localhost:3000/service





Aula 08 - Deploy com Heroku CLI

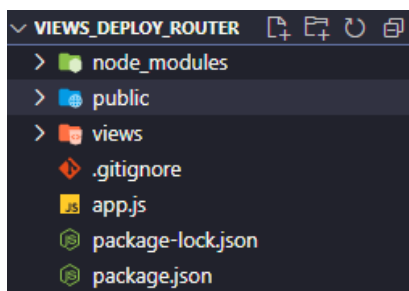
- Abra uma conta no Heroku.
- Instale Heroku CLI.

Configurações

package.json

```
{
  "name": "views_deploy_router",
  "version": "1.0.0",
  "description": "",
  "main": "index.js",
  "scripts": {
    "test": "echo \\\"Error: no test specified\\\" && exit 1",
    "start": "node app"
  },
  "keywords": [],
  "author": "",
  "license": "ISC",
  "devDependencies": {
    "nodemon": "^2.0.7"
  },
  "dependencies": {
    "ejs": "^3.1.6",
    "express": "^4.17.1"
  }
}
```

- Na pasta raiz do projeto, adicione o arquivo **.gitignore**:



.gitignore

node_modules

- Altere o arquivo app.js:

app.js

```
const express = require('express');
const app = express();
```

```
const port = process.env.PORT || 3000;
```

```
// motor de templates
app.set('view engine', 'ejs');
app.set('views', __dirname + '/views');
```

```
app.use(express.static(__dirname + "/public"));
```

```
// Rotas
```

```
app.get('/', (req, res) => {
  // res.send('Hello World!');
  res.render("index", {bodyTitle: "Página inicial", bodyContent: "Seja bem-vindo ao nosso site!"});
});
```

```
app.get('/services', (req, res) => {
  // res.send('Página de serviços!');
  res.render("services", {bodyTitle: "Página de Serviços", bodyContent: "Oferecemos os melhores serviços aos nossos clientes!"});
});
```

```
app.get('/contact', (req, res) => {
  // res.send('Página de contato!');
  res.render("contact", {bodyTitle: "Página de Contato", bodyContent: "Entre em contato conosco!"});
});
```

```
// Rota não encontrada
```

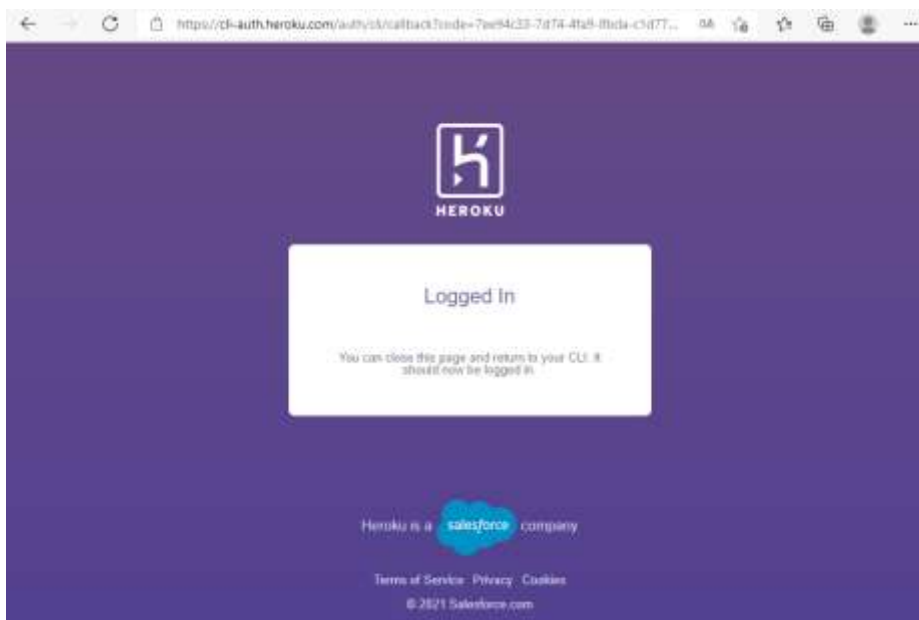
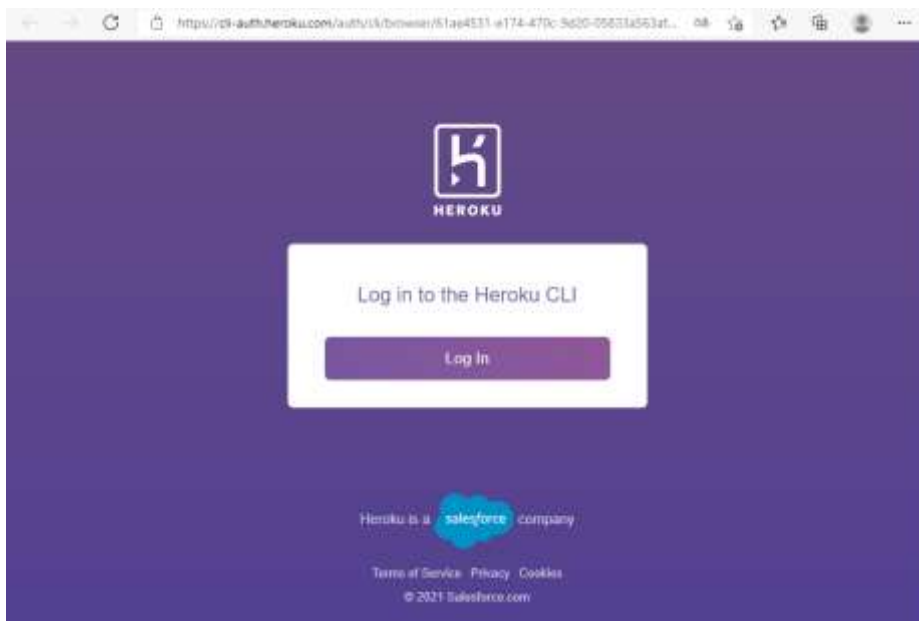
```
app.use((req, res, next) => {
  // res.status(404).sendFile(__dirname + "/public/404.html");
  res.status(404).render("404", {bodyTitle: "Página não encontrada!", bodyContent: "Essa página não existe!"});
});
```

```
app.listen(port, () =>{
  console.log('Servidor rodando na porta ' + port);
});
```

Deploy

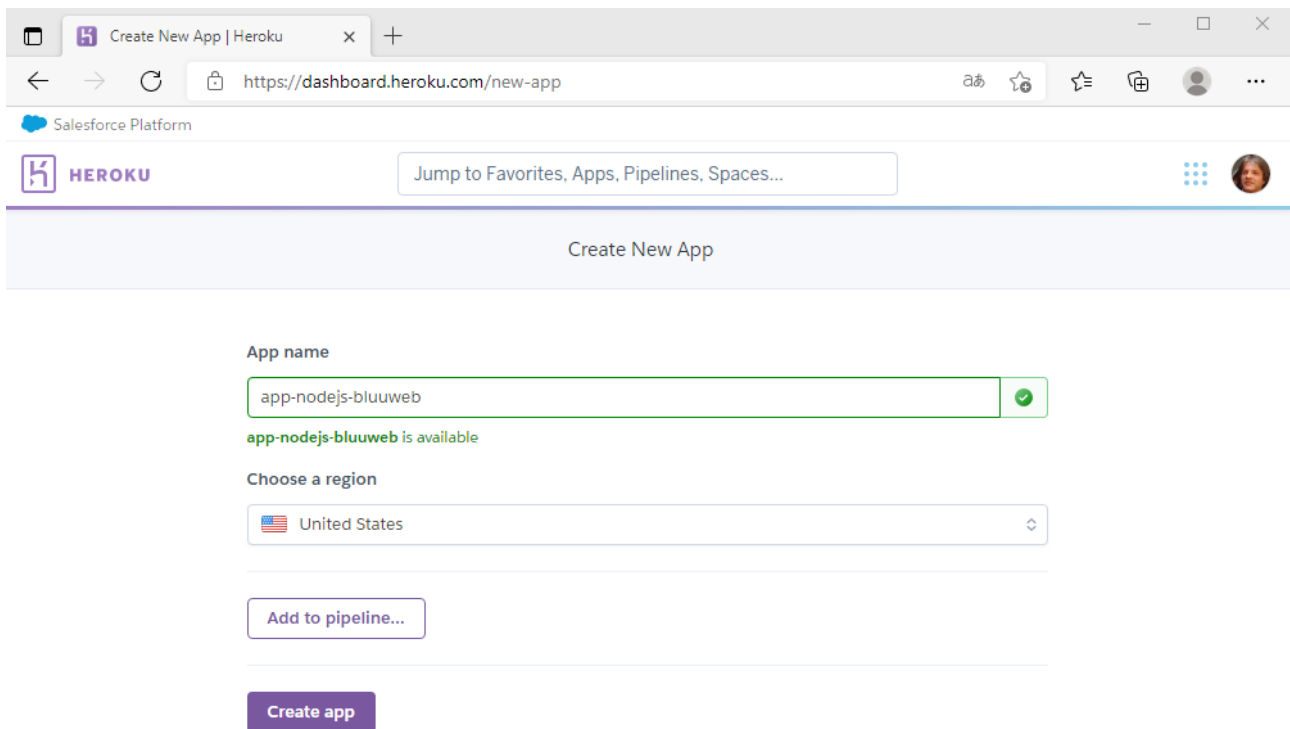
heroku login

```
C:\bluuweb\nodejs\views_deploy_router>heroku login
heroku: Press any key to open up the browser to login or q to exit:
Opening browser to https://cli-auth.heroku.com/auth/cli/browser/734181e2-62f8-4e31-b6d6-b2f514a414f8?requestor=SFMyNTY.g2gbQAA
AA8x00kuMTIwLjE0Ny4yMDBuBgAU4B_NeQFiAAFRgA.1f5joP-BP--i_uk8czKugP--W_G3th7xdlkfVFmc8d4
Logging in... done
Logged in as betopinheiro1005@yahoo.com.br
```



- Crie uma nova aplicação no Heroku:

app-nodejs-bluuweb



Create New App

App name

app-nodejs-bluuweb

app-nodejs-bluuweb is available

Choose a region

United States

Add to pipeline...

Create 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 app-nodejs-bluuweb
```

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](#).

git init

```
C:\bluuweb\nodejs\views_deploy_router>git init
Initialized empty Git repository in C:/bluuweb/nodejs/views_deploy_router/.git/
```

heroku git:remote -a app-nodejs-bluuweb

```
C:\bluuweb\nodejs\views_deploy_router>heroku git:remote -a app-nodejs-bluuweb
set git remote heroku to https://git.heroku.com/app-nodejs-bluuweb.git
```

git add .

```
C:\bluuweb\nodejs\views_deploy_router>git add .
warning: LF will be replaced by CRLF in package-lock.json.
The file will have its original line endings in your working directory
warning: LF will be replaced by CRLF in package.json.
The file will have its original line endings in your working directory
warning: LF will be replaced by CRLF in public/css/bootstrap.css.
The file will have its original line endings in your working directory
warning: LF will be replaced by CRLF in public/css/bootstrap.min.css.
The file will have its original line endings in your working directory
warning: LF will be replaced by CRLF in public/js/bootstrap.js.
The file will have its original line endings in your working directory
warning: LF will be replaced by CRLF in public/js/bootstrap.min.js.
The file will have its original line endings in your working directory
```

git commit -am "First commit"

```
C:\bluuweb\nodejs\views_deploy_router>git commit -am "First commit"
[master (root-commit) 13c974e] First commit
17 files changed, 24892 insertions(+)
create mode 100644 .gitignore
create mode 100644 app.js
create mode 100644 package-lock.json
create mode 100644 package.json
create mode 100644 public/css/bootstrap.css
create mode 100644 public/css/bootstrap.min.css
create mode 100644 public/js/bootstrap.js
create mode 100644 public/js/bootstrap.min.js
create mode 100644 public/js/jquery.js
create mode 100644 public/js/popper.min.js
create mode 100644 views/404.ejs
create mode 100644 views/contact.ejs
create mode 100644 views/index.ejs
create mode 100644 views/services.ejs
create mode 100644 views/templates/footer.ejs
create mode 100644 views/templates/header.ejs
create mode 100644 views/templates/navbar.ejs
```

git push heroku master

```
C:\bluuweb\nodejs\views_deploy_router>git push heroku master
Enumerating objects: 24, done.
Counting objects: 100% (24/24), done.
Delta compression using up to 2 threads
Compressing objects: 100% (23/23), done.
Writing objects: 100% (24/24), 180.60 KiB | 635.00 KiB/s, done.
Total 24 (delta 3), reused 0 (delta 0), pack-reused 0
remote: Compressing source files... done.
remote: Building source:
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: -----> Build
remote:
remote: -----> Caching build
remote:       - node_modules
remote:
remote: -----> Pruning devDependencies
remote:       removed 113 packages and audited 71 packages in 1.856s
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: 33.3M
remote: -----> Launching...
remote:       Released v3
remote:       https://app-nodejs-bluuweb.herokuapp.com/ deployed to Heroku
remote:
remote: Verifying deploy... done.
To https://git.heroku.com/app-nodejs-bluuweb.git
* [new branch]      master -> master
```

- No browser:

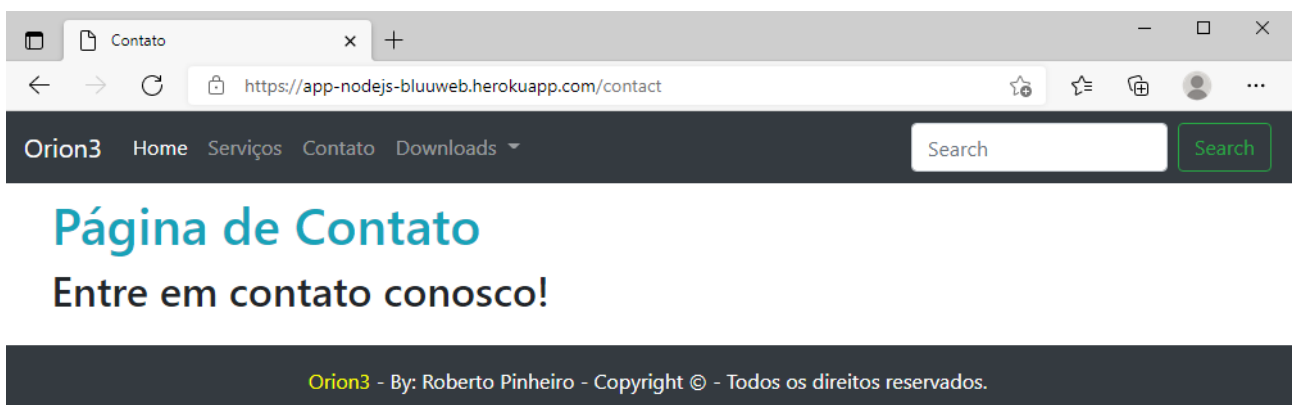
<https://app-nodejs-bluuweb.herokuapp.com/>



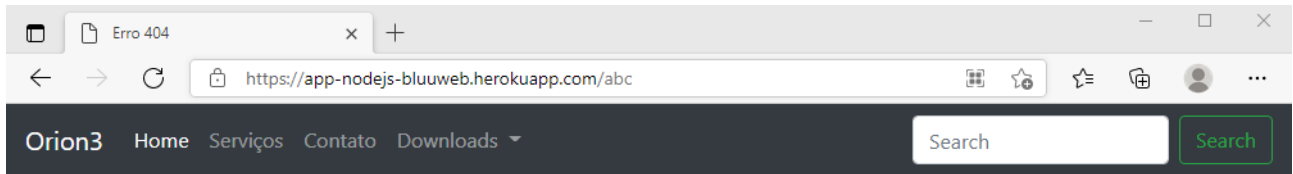
<https://app-nodejs-bluuweb.herokuapp.com/services>



<https://app-nodejs-bluuweb.herokuapp.com/contact>



https://app-nodejs-bluuweb.herokuapp.com/abc



Página não encontrada!

Essa página não existe!

Orion3 - By: Roberto Pinheiro - Copyright © - Todos os direitos reservados.

Aula 09 - Router Express

package.json

```
{
  "name": "views_deploy_router",
  "version": "1.0.0",
  "description": "",
  "main": "index.js",
  "scripts": {
    "test": "echo \"Error: no test specified\" && exit 1",
    "start": "node app",
    "dev": "nodemon app"
  },
  "keywords": [],
  "author": "",
  "license": "ISC",
  "devDependencies": {
    "nodemon": "^2.0.7"
  },
  "dependencies": {
    "ejs": "^3.1.6",
    "express": "^4.17.1"
  }
}
```

npm run dev

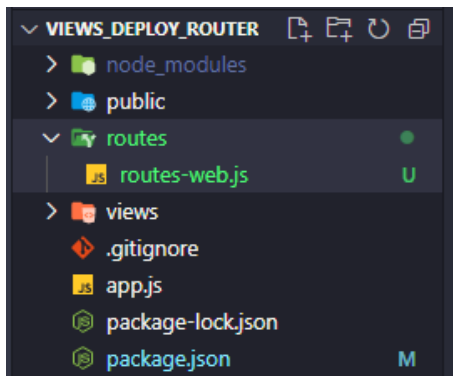
```
C:\bluuweb\nodejs\views_deploy_router>npm run dev

> views_deploy_router@1.0.0 dev C:\bluuweb\nodejs\views_deploy_router
> nodemon app

[nodemon] 2.0.7
[nodemon] to restart at any time, enter `rs`
[nodemon] watching path(s): *.*
[nodemon] watching extensions: js,mjs,json
[nodemon] starting `node app.js`
Servidor rodando na porta 3000
█
```


Rotas

- Na pasta do projeto adicione uma subpasta chamada "routes" e dentro dela adicione um arquivo chamado "routes-web.js":



routes\routes-web.js

```
const express = require('express');
const router = express.Router();

// Rotas

router.get('/', (req, res) => {
  res.render("index", {bodyTitle: "Página inicial", bodyContent: "Seja bem-vindo ao nosso site!"});
});

router.get('/services', (req, res) => {
  res.render("services", {bodyTitle: "Página de Serviços", bodyContent: "Oferecemos os melhores serviços aos nossos clientes!"});
});

router.get('/contact', (req, res) => {
  res.render("contact", {bodyTitle: "Página de Contato", bodyContent: "Entre em contato conosco!"});
});

module.exports = router;
```

- Altere o arquivo "app.js":

app.js

```
const express = require('express');
const app = express();

const port = process.env.PORT || 3000;

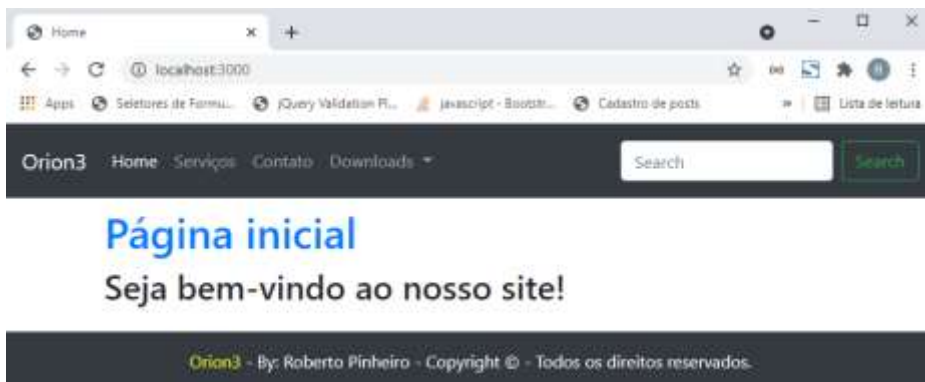
// motor de templates
app.set('view engine', 'ejs');
app.set('views', __dirname + '/views');

app.use(express.static(__dirname + "/public"));

// Rotas Web
app.use('/', require('./routes/routes-web'));

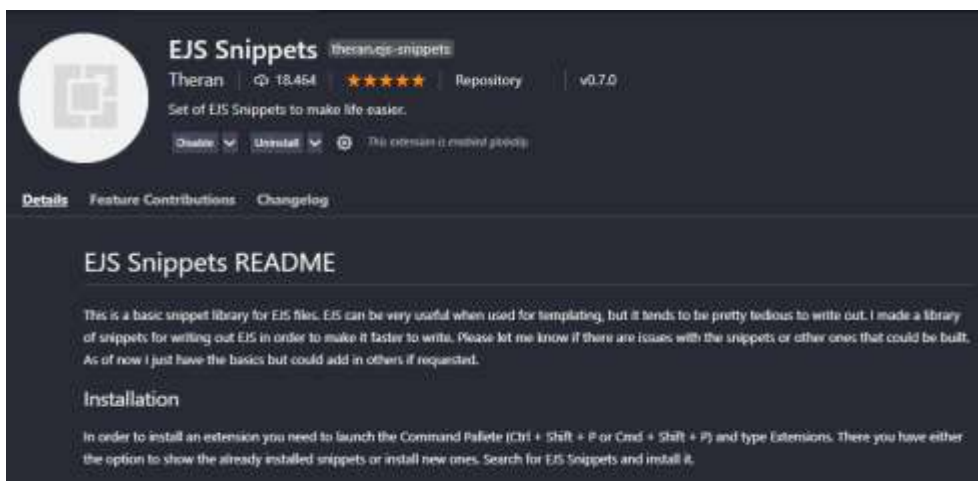
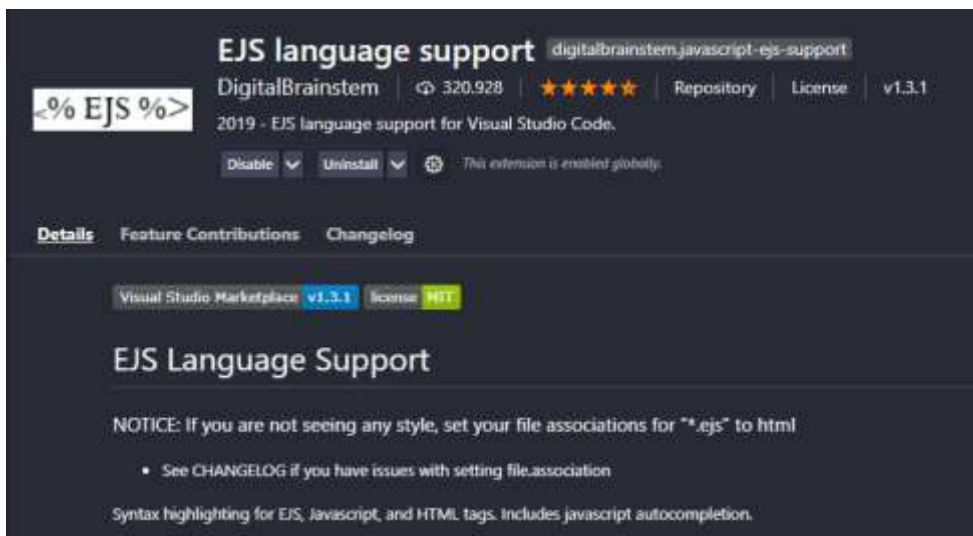
// Rota não encontrada
app.use((req, res, next) => {
  // res.status(404).sendFile(__dirname + "/public/404.html");
  res.status(404).render("404", {bodyTitle: "Página não encontrada!", bodyContent: "Essa página não existe!"});
});

app.listen(port, () =>{
  console.log('Servidor rodando na porta ' + port);
});
```

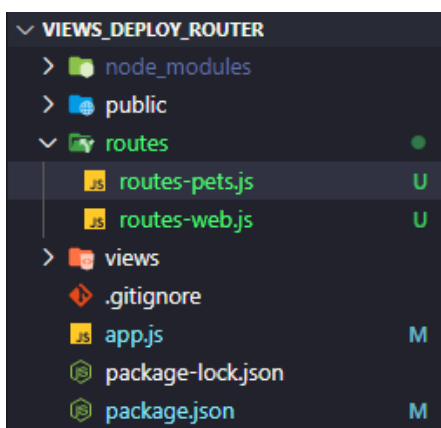


- Instale duas extensões no Visual Studio Code:

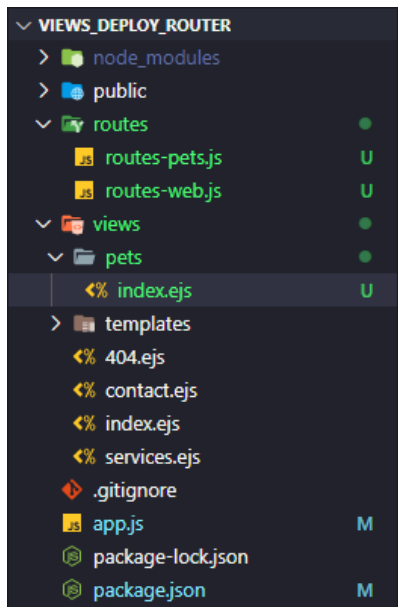
- EJS language support
- EJS Snippets



- Dentro da pasta "**routes**" e adicione um arquivo chamado "**routes-pets.js**":



- Dentro da pasta **views** adicione a subpasta **pets** e dentro dela o arquivo **index.ejs**:



routes\routes-pets.js

```
const express = require('express');
const router = express.Router();

// Rotas

router.get('/', (req, res) => {
  res.render("pets/index", {
    arrayPets: [
      {id: 1, nome: "Brisa", especie: "gato", sexo: 'F'},
      {id: 2, nome: "Fred", especie: "gato", sexo: 'M'},
      {id: 3, nome: "Teo", especie: "gato", sexo: 'M'},
      {id: 4, nome: "Benjamin", especie: "gato", sexo: 'M'},
      {id: 5, nome: "Tiquinho", especie: "gato", sexo: 'M'},
      {id: 6, nome: "Siloeh", especie: "cachorro", sexo: 'F'},
      {id: 7, nome: "Valentim", especie: "cachorro", sexo: 'M'},
      {id: 8, nome: "Roque", especie: "cachorro", sexo: 'M'}
    ]
  });
});

module.exports = router;
```

app.js

```
const express = require('express');
const app = express();

const port = process.env.PORT || 3000;

// motor de templates
app.set('view engine', 'ejs');
app.set('views', __dirname + '/views');

app.use(express.static(__dirname + "/public"));

// Rotas Web
app.use('/', require('./routes/rotasWeb'));
app.use('/pets', require('./routes/routes-pets'));

// Rota não encontrada
app.use((req, res, next) => {
  // res.status(404).sendFile(__dirname + "/public/404.html");
  res.status(404).render("404", {bodyTitle: "Página não encontrada!", bodyContent: "Essa página não existe!"});
});

app.listen(port, () =>{
  console.log('Servidor rodando na porta ' + port);
});
```

views\pets\index.ejs

```
<%- include("../templates/header", {bodyTitle: "Home" }) %>

<div class="container mb-4 mt-2">
  <table class="table">
    <thead>
      <tr>
        <th scope="col">Id</th>
        <th scope="col">Nome</th>
        <th scope="col">Espécie</th>
        <th scope="col">Sexo</th>
        <th scope="col">Ações</th>
      </tr>
    </thead>
    <tbody>
      <% arrayPets.forEach(pet => { %>
        <tr>
          <th scope="row"><%= pet.id %> </th>
          <td><%= pet.nome %> </td>
          <td><%= pet.especie %> </td>
          <td><%= pet.sexo %> </td>
          <td></td>
        </tr>
      <% }} %>
    </tbody>
  </table>
</div>

<%- include("../templates/footer") %>
```

- No browser:

http://localhost:3000/pets

←

→

↻

localhost:3000/pets

☆

(=)

S

⚙

R

⋮

Apps

Seletores de Formu...

jQuery Validation PL...

javascript - Bootstr...

Cadastro de posts

»

Lista de leitura

Orion3

Home

Serviços

Contato

Downloads ▾

Search

Search

Id	Nome	Espécie	Sexo	Ações
1	Brisa	gato	F	
2	Fred	gato	M	
3	Teo	gato	M	
4	Benjamin	gato	M	
5	Tiquinho	gato	M	
6	Siloeh	cachorro	F	
7	Valentim	cachorro	M	
8	Roque	cachorro	M	

Orion3 - By: Roberto Pinheiro - Copyright © - Todos os direitos reservados.

Aula 10 - Conexão a MongoDB

- Em `C:\bluuweb\nodejs` crie uma pasta chamada "`mongodb`"

```
C:\bluuweb\nodejs>md mongodb

C:\bluuweb\nodejs>dir
O volume na unidade C é W10-195
O Número de Série do Volume é 6047-7D0C

Pasta de C:\bluuweb\nodejs

02/06/2021  13:36    <DIR>        .
02/06/2021  13:36    <DIR>        ..
02/06/2021  03:17    <DIR>        Fundamentos
02/06/2021  13:36    <DIR>        mongodb
02/06/2021  04:12    <DIR>        rotas_basico
02/06/2021  11:47    <DIR>        views_deploy_router
                0 arquivo(s)                0 bytes
                6 pasta(s)   111.288.029.184 bytes disponíveis
```

`cd mongodb`

```
C:\bluuweb\nodejs>cd mongodb
C:\bluuweb\nodejs\mongodb>
```

- Copie todo o conteúdo da pasta `C:\bluuweb\nodejs\views_deploy_router` para dentro da pasta `C:\bluuweb\nodejs\mongodb`

code .

Instalando Mongoose

`npm install --save mongoose`

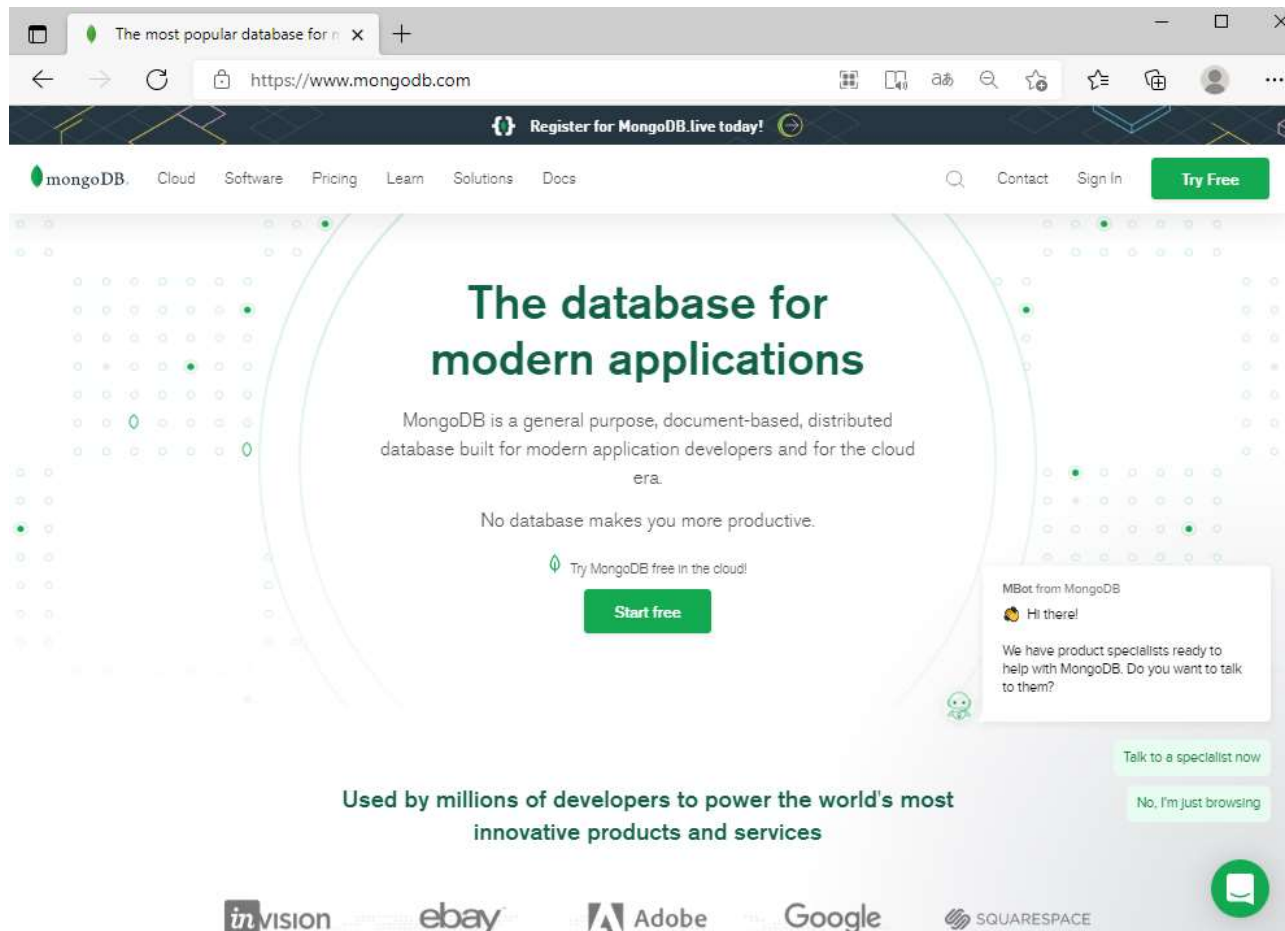
```
C:\bluuweb\nodejs\mongodb>npm install --save mongoose
npm WARN views_deploy_router@1.0.0 No description
npm WARN views_deploy_router@1.0.0 No repository field.
npm WARN optional SKIPPING OPTIONAL DEPENDENCY: fsevents@2.3.2 (node_modules\fsevents):
npm WARN notsup SKIPPING OPTIONAL DEPENDENCY: Unsupported platform for fsevents@2.3.2: wanted {"os":"darwin","arch":"any"} (current: {"os":"win32","arch":"x64"})

+ mongoose@5.12.12
added 30 packages from 92 contributors and audited 215 packages in 33.97s

13 packages are looking for funding
  run `npm fund` for details

found 0 vulnerabilities
```


<https://www.mongodb.com/>



The MongoDB homepage features a dark header with navigation links: Cloud, Software, Pricing, Learn, Solutions, Docs, Contact, Sign In, and a prominent green 'Try Free' button. The main content area has a light background with a grid of green dots. The headline reads 'The database for modern applications', followed by a description of MongoDB as a general purpose, document-based, distributed database. A sub-headline states 'No database makes you more productive.' and a green button says 'Start free'. A chatbot window on the right says 'Hi there!' and offers to help with MongoDB. At the bottom, logos for InVision, eBay, Adobe, Google, and Squarespace are displayed.

The database for modern applications

MongoDB is a general purpose, document-based, distributed database built for modern application developers and for the cloud era.

No database makes you more productive.

Try MongoDB free in the cloud!

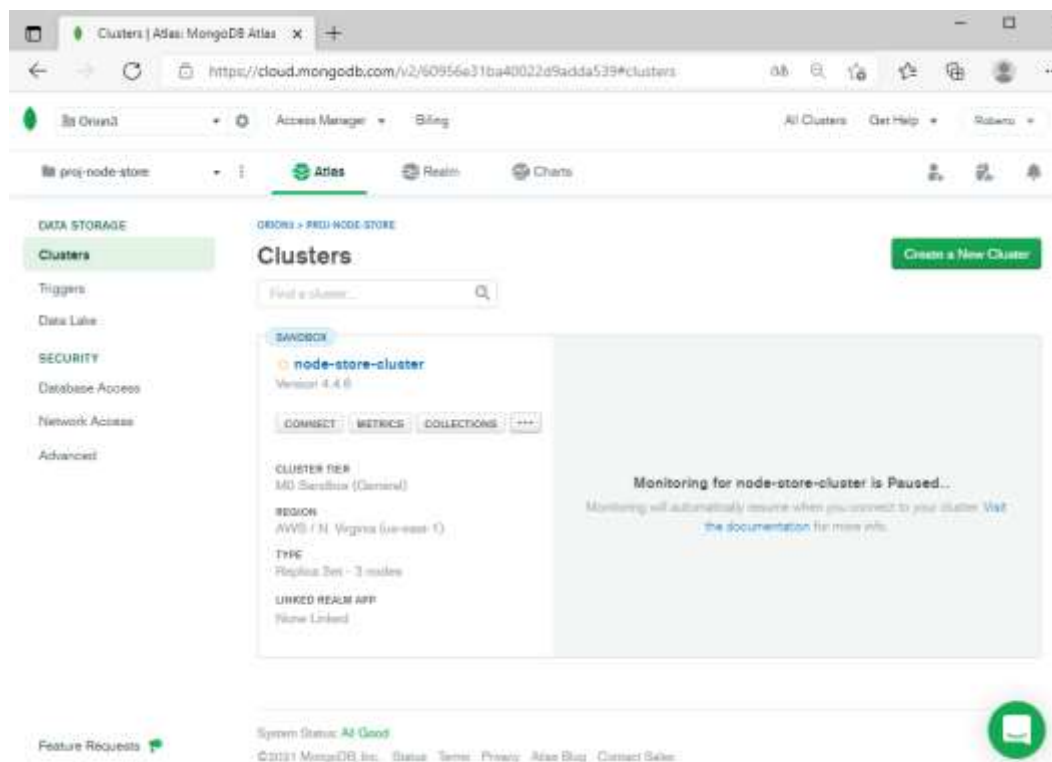
[Start free](#)

MBot from MongoDB
Hi there!
We have product specialists ready to help with MongoDB. Do you want to talk to them?

[Talk to a specialist now](#)
[No, I'm just browsing](#)

Used by millions of developers to power the world's most innovative products and services

invision ebay Adobe Google SQUARESPACE



The MongoDB Atlas Clusters page shows a list of clusters. The first cluster is 'node-store-cluster' (Version 4.4.6) in the 'AWS (N. Virginia (us-east-1))' region. It is a 'Replica Set - 3 nodes' type. The page includes a sidebar with navigation links like Clusters, Triggers, Data Lake, SECURITY, Database Access, Network Access, and Advanced. A green button 'Create a New Cluster' is visible. The bottom of the page shows system status as 'All Good' and copyright information for MongoDB, Inc. 2021.

Clusters

[Create a New Cluster](#)

Find a cluster...

node-store-cluster
Version 4.4.6

[CONNECT](#) [METRICS](#) [COLLECTIONS](#) [...](#)

CLUSTER Tier
M0 Sandbox (Default)

REGION
AWS (N. Virginia (us-east-1))

TYPE
Replica Set - 3 nodes

LINKED REALM APP
None Linked

Monitoring for node-store-cluster is Paused...
Monitoring will automatically resume when you connect to your cluster. [Visit the documentation for more info.](#)

System Status: **All Good**

©2021 MongoDB, Inc. [Status](#) [Terms](#) [Privacy](#) [Atlas Blog](#) [Contact Sales](#)

- Crie um projeto chamado "nodejs-bluuweb":

blogapp

proj-node-store (current)

View All Projects

+ New Project

ORIONS > PROJECTS

Create a Project

Name Your Project

Add Members

Next

Name Your Project

Project names have to be unique within the organization (and other restrictions).

nodejs-bluuweb

Cancel

Next

CLUSTERS > CREATE A SHARED CLUSTER

Create a Shared Cluster

Welcome to MongoDB Atlas! We've recommended some of our most popular options, but feel free to customize your cluster to your needs. For more information, check our [documentation](#).

Cloud Provider & Region

AWS, N. Virginia (us-east-1) ▾



Google Cloud

Azure

★ Recommended region ⓘ

ASIA	NORTH AMERICA	EUROPE
Mumbai (ap-south-1)	N. Virginia (us-east-1) ★	Frankfurt (eu-central-1) ★
Singapore (ap-southeast-1) ★	Oregon (us-west-2) ★	Ireland (eu-west-1) ★
AUSTRALIA		
Sydney (ap-southeast-2) ★		

Cluster Tier

M0 Sandbox (Shared RAM, 512 MB Storage) >
Encrypted

Additional Settings

MongoDB 4.4, No Backup >

Cluster Name

Cluster0 >

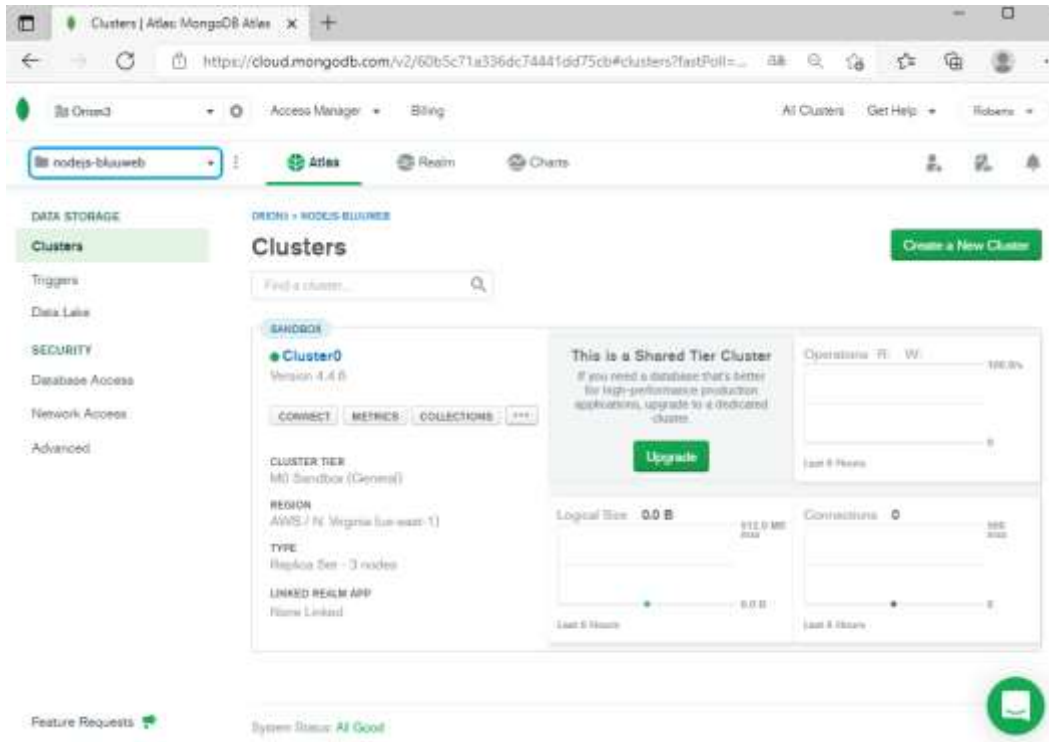
FREE

Free forever! Our M0 cluster is ideal for experimenting in a limited sandbox. You can upgrade to a production cluster anytime.

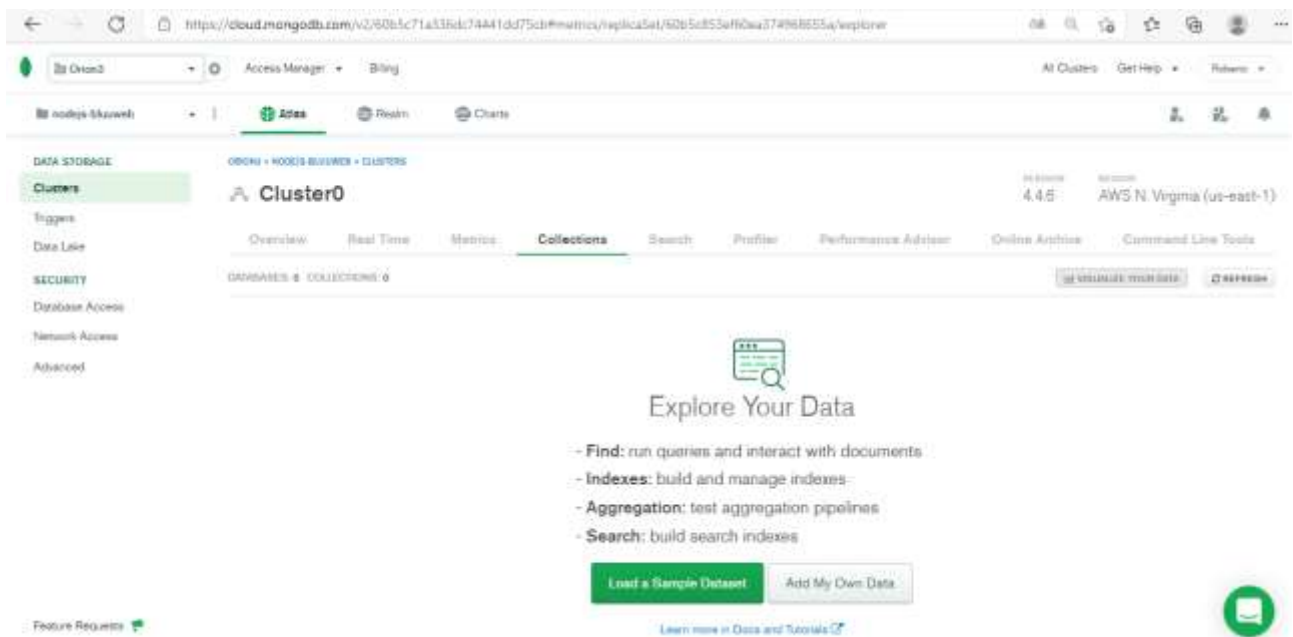
[Back](#)

Create Cluster

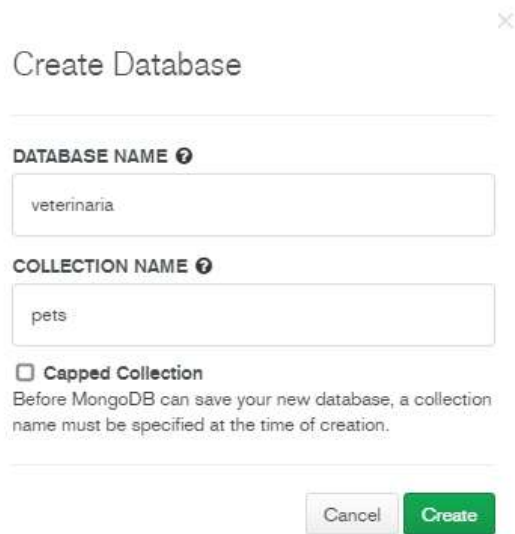
- Clique no botão "Create Cluster" (FREE) e aguarde a criação (o processo leva alguns minutos):



- Clique no botão "COLLECTIONS":



- Clique no botão "Add My Own Data":



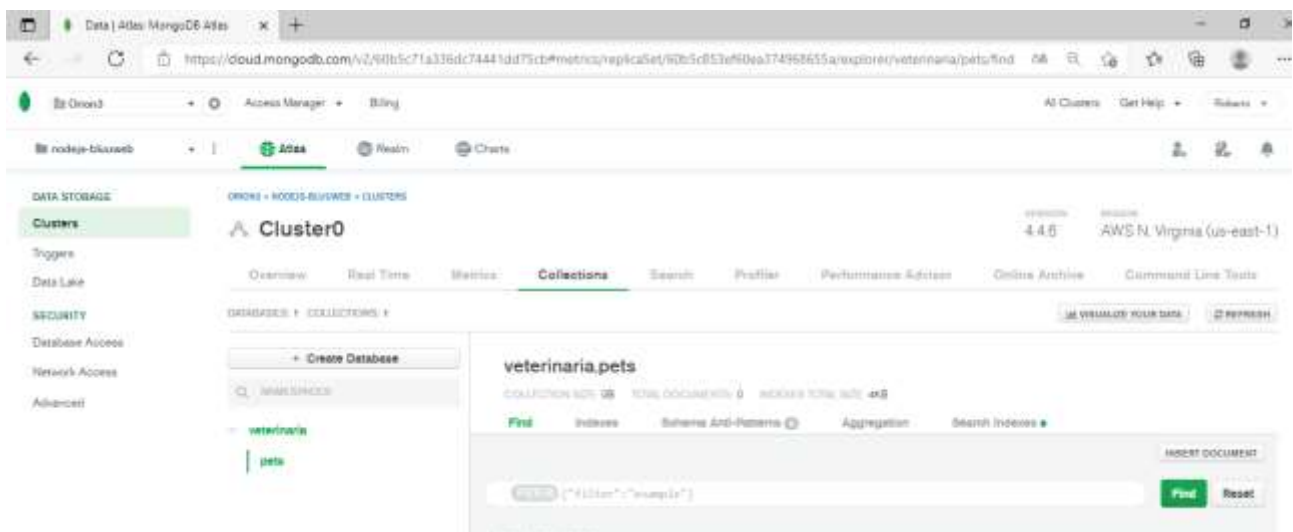
A dialog box titled "Create Database" with a close button (X) in the top right corner. It contains two input fields: "DATABASE NAME" with the value "veterinaria" and "COLLECTION NAME" with the value "pets". Below these fields is a checkbox labeled "Capped Collection" which is unchecked. A note below the checkbox states: "Before MongoDB can save your new database, a collection name must be specified at the time of creation." At the bottom are two buttons: "Cancel" and "Create".

- Entre com:

DATABASE NAME: veterinaria

COLLECTION NAME: pets

- Clique no botão "Create"



- Clique no botão "**INSERT DOCUMENT**" e insira o primeiro pet (campos: nome, especie, sexo com os respectivos valores).

Insert to Collection

VIEW ☐ ☒ ☐

```
1  _id : ObjectId("60b5d1b5d04d8e38770af3e1")
2  nome : "Brisa"
3  especie : "gato"
4  sexo : "F"
```

ObjectId

String

String

String

- Repita a operação para todos os demais pets.

QUERY RESULTS 1-8 OF 8

```
_id: ObjectId("60b5d1b5d04d8e38770af3e1")
nome: "Brisa"
especie: "gato"
sexo: "F"
```

```
_id: ObjectId("60b5d27bd04d8e38770af3e2")
nome: "Fred"
especie: "gato"
sexo: "M"
```

```
_id: ObjectId("60b5d2cad04d8e38770af3e3")
nome: "Téo"
especie: "gato"
sexo: "M"
```

```
_id: ObjectId("60b5d301d04d8e38770af3e4")
nome: "Benjamin"
especie: "gato"
sexo: "M"
```

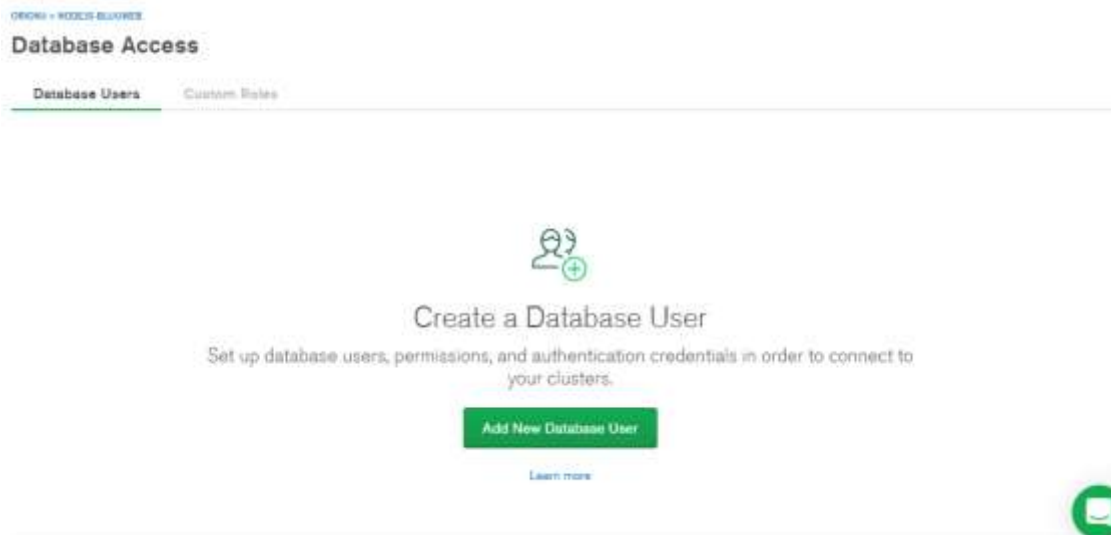
```
_id: ObjectId("60b5d349d04d8e38770af3e5")  
nome: "Tiquinho"  
especie: "gato"  
sexo: "M"
```

```
_id: ObjectId("60b5d393d04d8e38770af3e6")  
nome: "Silloeh"  
especie: "cachorro"  
sexo: "f"
```

```
_id: ObjectId("60b5d3cbd04d8e38770af3e7")  
nome: "Valentim"  
especie: "cachorro"  
sexo: "M"
```

```
_id: ObjectId("60b5d3efd04d8e38770af3e8")  
nome: "Roque"  
especie: "cachorro"  
sexo: "M"
```

- Em "SECURITY" clique em "Database access"



- Clique no botão "Add New Database User" e entre com o nome do usuário e a senha:

Add New Database User

Create a database user to grant an application or user, access to databases and collections in your clusters in this Atlas project. Granular access control can be configured with default privileges or custom roles. You can grant access to an Atlas project or organization using the corresponding [Access Manager](#) 

Authentication Method

Password

Certificate

AWS IAM
(MongoDB 4.4 and up)

MongoDB uses [SCRAM](#) as its default authentication method.

Password Authentication

betopinheiro

DpuggJSVjb4qFQyx

HIDE

 Autogenerate Secure Password

 Copy

Database User Privileges

Select a [built-in role](#) or [privileges](#) for this user.

Read and write to any database ▼

- Clique no botão "Add User"

We are deploying your changes (current action: creating a plan)

ORDERS • RODRIGS-BLUTHER

Database Access

Database Users

Custom Roles

 ADD NEW DATABASE USER

User Name &	Authentication Method <	MongoDB Roles	Resources	Actions
 betopinheiro	SCRAM	readWriteAnyDatabase@admin	All Resources	 EDIT  DELETE

- Em Cluster0, clique no botão "CONNECT":

Connect to Cluster0

Setup connection security Choose a connection method Connect

You need to secure your MongoDB Atlas cluster before you can use it. Set which users and IP addresses can access your cluster now. [Read more](#)

You're ready to connect. Choose how you want to connect in the next step.

- 1 Add a connection IP address
✓ An IP address has been added to the IP Access List. Add another address in the [IP Access List](#) tab.
- 2 Create a Database User
✓ A MongoDB user has been added to this project. Not yours? Create one in the [MongoDB Users](#) tab.
You'll need your MongoDB user's credentials in the next step.

Close Choose a connection method


- Clique no botão "Choose a connection method":

Connect to Cluster0

✓ Setup connection security Choose a connection method Connect


Choose a connection method [View documentation](#)

Get your pre-formatted connection string by selecting your tool below.




Connect with the mongo shell
Interact with your cluster using MongoDB's interactive Javascript interface

>



Connect your application
Connect your application to your cluster using MongoDB's native drivers

>



Connect using MongoDB Compass
Explore, modify, and visualize your data with MongoDB's GUI

>

Go Back Close

- Clique em "Connect your application"

Connect to Cluster0

✓ Setup connection security

✓ Choose a connection method

Connect

1

Select your driver and version

DRIVER

Node.js

VERSION


3.6 or later

2

Add your connection string into your application code

☐ Include full driver code example

mongodb+srv://betopinheiro:<password>@cluster0.cyvyx.mongodb.net/myFirstDatabase?
retryWrites=true&w=majority



Replace **<password>** with the password for the **betopinheiro** user. Replace **myFirstDatabase** with the name of the database that connections will use by default. Ensure any option params are [URL encoded](#).

Having trouble connecting? [View our troubleshooting documentation](#)

Go Back

Close

- Copie a string de conexão, substituindo **<password>** por **DpuggJSVjb4qFQyx** e **myFirstDatabase** por **veterinaria**.

mongodb+srv://betopinheiro:DpuggJSVjb4qFQyx@cluster0.cyvyx.mongodb.net/veterinaria?retryWrites=true&w=majority

- Essa é a string a ser utilizada na configuração de conexão ao BD.

app.js

```
const express = require('express');
const app = express();

const port = process.env.PORT || 3000;

const mongoose = require('mongoose');

// Conexão a base de dados
const user = "betopinheiro";
const password = "DpuggJSVjb4qFQyx";
const dbname = "veterinaria";
const uri = "mongodb+srv://$${user}:$${password}@cluster0.cyvyx.mongodb.net/$${dbname}?retryWrites=true&w=majority";

mongoose.connect(uri, {useNewUrlParser: true, useUnifiedTopology: true})
.then(() => {
  console.log('Base de dados conectada!');
}).catch((error) => {
  console.log(error);
});

// motor de templates
app.set('view engine', 'ejs');
app.set('views', __dirname + '/views');

app.use(express.static(__dirname + "/public"));

// Rotas Web
app.use('/', require('./routes/routes-web'));
app.use('/pets', require('./routes/routes-pets'));

// Rota não encontrada
app.use((req, res, next) => {
  // res.status(404).sendFile(__dirname + "/public/404.html");
  res.status(404).render("404", {bodyTitle: "Página não encontrada!", bodyContent: "Essa página não existe!"});
});

app.listen(port, () =>{
  console.log('Servidor rodando na porta ' + port);
});
```

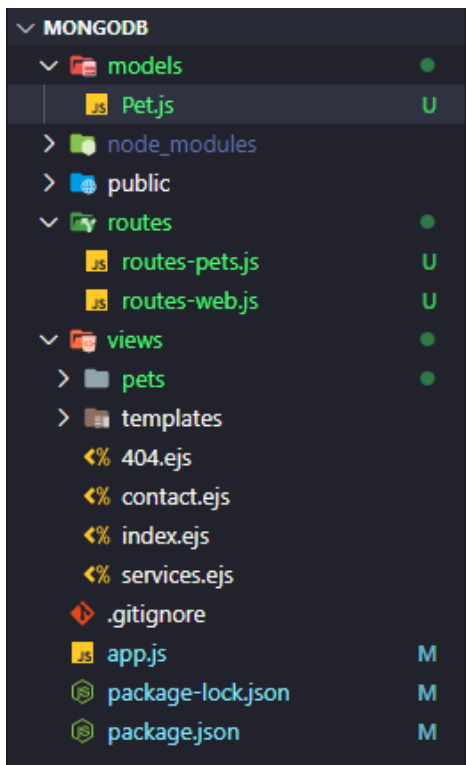
npm run dev

```
C:\bluuweb\nodejs\mongodb>npm run dev

> views_deploy_router@1.0.0 dev C:\bluuweb\nodejs\mongodb
> nodemon app

[nodemon] 2.0.7
[nodemon] to restart at any time, enter `rs`
[nodemon] watching path(s): *.*
[nodemon] watching extensions: js,mjs,json
[nodemon] starting `node app.js`
Servidor rodando na porta 3000
Base de dados conectada!
```

- Na pasta do projeto crie uma subpasta chamada "**models**" e dentro dela adicione um arquivo chamado **Pet.js**:



models\Pet.js

```
const mongoose = require('mongoose');
const Schema = mongoose.Schema;

const petSchema = new Schema({
  nome: String,
  especie: String,
  sexo: String
});

// Criar model
const Pet = mongoose.model('Pet', petSchema);

module.exports = Pet;
```

routes\routes-pets.js

```
const express = require('express');
const router = express.Router();
const Pet = require('../models/Pet');

// Rotas

router.get('/', async (req, res) => {

  try {
    const arrayPetsDB = await Pet.find();
    console.log(arrayPetsDB);
  } catch (error) {
    console.log(error);
  }

  res.render("pets/index", {
    arrayPets: [
      {id: 1, nome: "Brisa", especie: "gato", sexo: 'F'},
      {id: 2, nome: "Fred", especie: "gato", sexo: 'M'},
      {id: 3, nome: "Teo", especie: "gato", sexo: 'M'},
      {id: 4, nome: "Benjamin", especie: "gato", sexo: 'M'},
      {id: 5, nome: "Tiquinho", especie: "gato", sexo: 'M'},
      {id: 6, nome: "Siloeh", especie: "cachorro", sexo: 'F'},
      {id: 7, nome: "Valentim", especie: "cachorro", sexo: 'M'},
      {id: 8, nome: "Roque", especie: "cachorro", sexo: 'M'}
    ]
  });
});

module.exports = router;
```

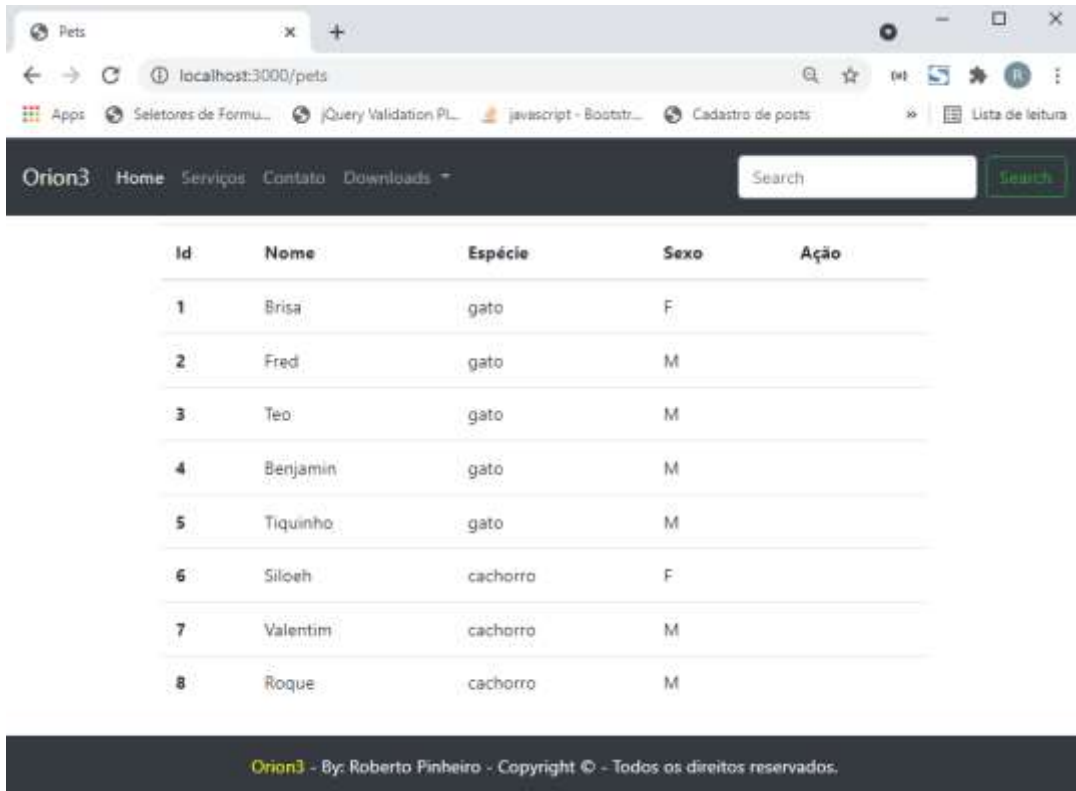
npm run dev

```
C:\bluuweb\nodejs\mongodb>npm run dev

> views_deploy_router@1.0.0 dev C:\bluuweb\nodejs\mongodb
> nodemon app

[nodemon] 2.0.7
[nodemon] to restart at any time, enter `rs`
[nodemon] watching path(s): *.*
[nodemon] watching extensions: js,mjs,json
[nodemon] starting `node app.js`
Servidor rodando na porta 3000
Base de dados conectada!
█
```

- Recarregue o browser:



The screenshot shows a web browser window with the address bar at `localhost:3000/pets`. The page title is "Pets". The navigation bar includes "Orion3", "Home", "Serviços", "Contato", and "Downloads", along with a search bar. Below the navigation bar is a table with 5 columns: "Id", "Nome", "Espécie", "Sexo", and "Ação". The table contains 8 rows of pet data. At the bottom of the page, there is a footer that reads "Orion3 - By: Roberto Pinheiro - Copyright © - Todos os direitos reservados."

Id	Nome	Espécie	Sexo	Ação
1	Brisa	gato	F	
2	Fred	gato	M	
3	Teo	gato	M	
4	Benjamin	gato	M	
5	Tiquinho	gato	M	
6	Siloeh	cachorro	F	
7	Valentim	cachorro	M	
8	Roque	cachorro	M	

Orion3 - By: Roberto Pinheiro - Copyright © - Todos os direitos reservados.

- Observe no terminal:

```
C:\bluuweb\nodejs\mongodb>npm run dev
> views_deploy_router@1.0.0 dev C:\bluuweb\nodejs\mongodb
> nodemon app

[nodemon] 2.0.7
[nodemon] to restart at any time, enter `rs`
[nodemon] watching path(s): *.*
[nodemon] watching extensions: js,mjs,json
[nodemon] starting `node app.js`
Servidor rodando na porta 3000
Base de dados conectada!
[
  {
    _id: 60b5d1b5d04d8e38770af3e1,
    nome: 'Brisa',
    especie: 'gato',
    sexo: 'F'
  },
  {
    _id: 60b5d27bd04d8e38770af3e2,
    nome: 'Fred',
    especie: 'gato',
    sexo: 'M'
  },
  {
    _id: 60b5d2cad04d8e38770af3e3,
    nome: 'Téo',
    especie: 'gato',
    sexo: 'M'
  }
],
```

```
{
  _id: 60b5d301d04d8e38770af3e4,
  nome: 'Benjamin',
  especie: 'gato',
  sexo: 'M'
},
{
  _id: 60b5d349d04d8e38770af3e5,
  nome: 'Tiquinho',
  especie: 'gato',
  sexo: 'M'
},
{
  _id: 60b5d393d04d8e38770af3e6,
  nome: 'Silloeh',
  especie: 'cachorro',
  sexo: 'F'
},
{
  _id: 60b5d3cbd04d8e38770af3e7,
  nome: 'Valentim',
  especie: 'cachorro',
  sexo: 'M'
},
{
  _id: 60b5d3efd04d8e38770af3e8,
  nome: 'Roque',
  especie: 'cachorro',
  sexo: 'M'
}
```

routes\routes-pets.js

```
const express = require('express');
const router = express.Router();
const Pet = require('../models/Pet');

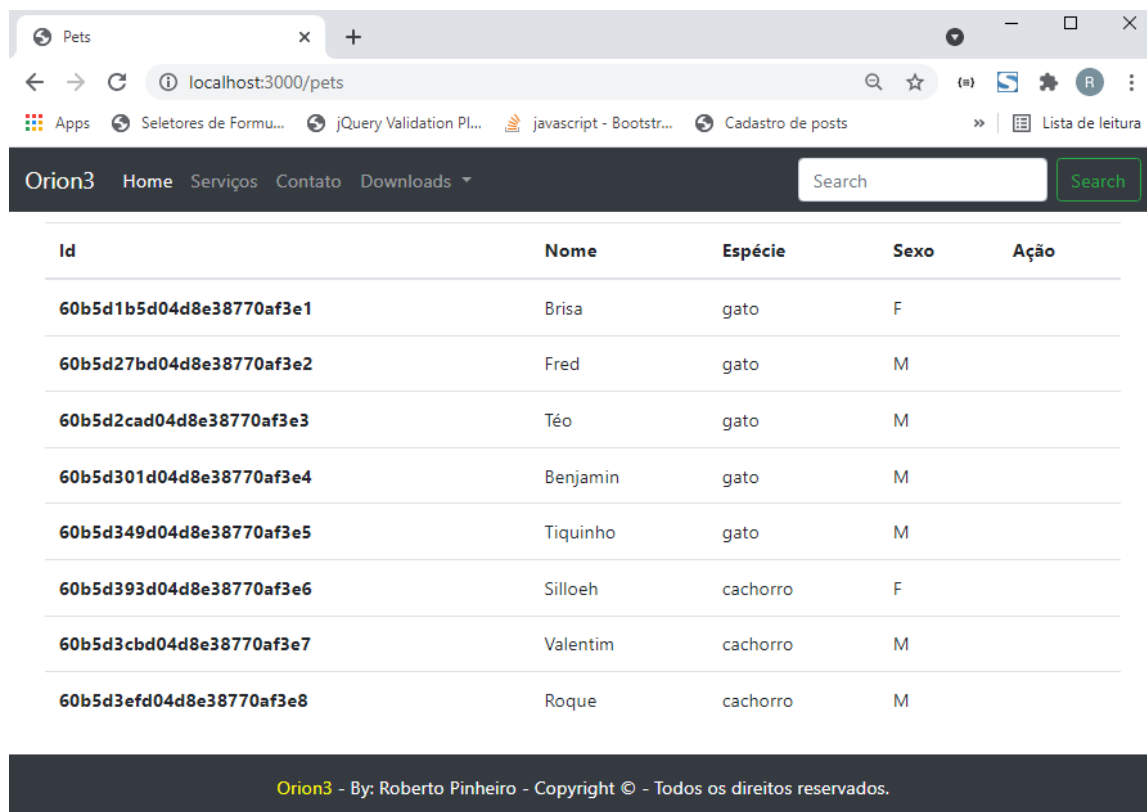
// Rotas

router.get('/', async (req, res) => {
  try {
    const arrayPetsDB = await Pet.find();
    console.log(arrayPetsDB);
    res.render("pets/index", {arrayPets: arrayPetsDB});
  } catch (error) {
    console.log(error);
  }
});

module.exports = router;
```

- No browser:

<http://localhost:3000/pets>



Orion3 Home Serviços Contato Downloads ▾ Search Search

Id	Nome	Espécie	Sexo	Ação
60b5d1b5d04d8e38770af3e1	Brisa	gato	F	
60b5d27bd04d8e38770af3e2	Fred	gato	M	
60b5d2cad04d8e38770af3e3	Téo	gato	M	
60b5d301d04d8e38770af3e4	Benjamin	gato	M	
60b5d349d04d8e38770af3e5	Tiquinho	gato	M	
60b5d393d04d8e38770af3e6	Silloeh	cachorro	F	
60b5d3cbd04d8e38770af3e7	Valentim	cachorro	M	
60b5d3efd04d8e38770af3e8	Roque	cachorro	M	

Orion3 - By: Roberto Pinheiro - Copyright © - Todos os direitos reservados.

Aula 11 - Variáveis de ambiente - dotenv (.env)

Instalando dotenv

<https://www.npmjs.com/package/dotenv>

Install

```
# with npm
npm install dotenv

# or with Yarn
yarn add dotenv
```

npm install --save dotenv

```
C:\bluuweb\nodejs\mongodb>npm install --save dotenv
npm WARN views_deploy_router@1.0.0 No description
npm WARN views_deploy_router@1.0.0 No repository field.
npm WARN optional SKIPPING OPTIONAL DEPENDENCY: fsevents@2.3.2 (node_modules\fsevents):
npm WARN notsup SKIPPING OPTIONAL DEPENDENCY: Unsupported platform for fsevents@2.3.2: wanted {"os":"darwin","arch":"any"} (current: {"os":"win32","arch":"x64"})

+ dotenv@10.0.0
added 1 package and audited 216 packages in 10.238s

13 packages are looking for funding
  run `npm fund` for details

found 0 vulnerabilities
```

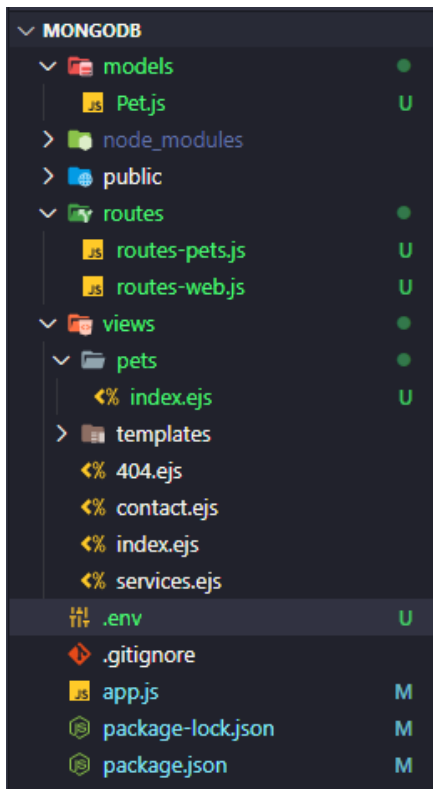
Uso

Usage

As early as possible in your application, require and configure dotenv.

```
require('dotenv').config()
```


- Na pasta do projeto, adicione um arquivo chamado **.env**:



.env

```
PORT=3002
USER=betopinheiro
PASSWORD=DpuggJSVjb4qFQyx
DBNAME=veterinaria
```

app.js

```
const express = require('express');
const app = express();
const mongoose = require('mongoose');

require('dotenv').config();

// Conexão a base de dados
const uri =
`mongodb+srv://${process.env.USER}:${process.env.PASSWORD}@cluster0.cyvyx.mongodb.net/${process.env.DBNAME}?retryWrites=true&w=majority`;

mongoose.connect(uri, {useNewUrlParser: true, useUnifiedTopology: true})
.then(() => {
  console.log('Base de dados conectada!');
}).catch((error) => {
  console.log(error);
});

// motor de templates
app.set('view engine', 'ejs');
```

```
app.set('views', __dirname + '/views');

app.use(express.static(__dirname + "/public"));

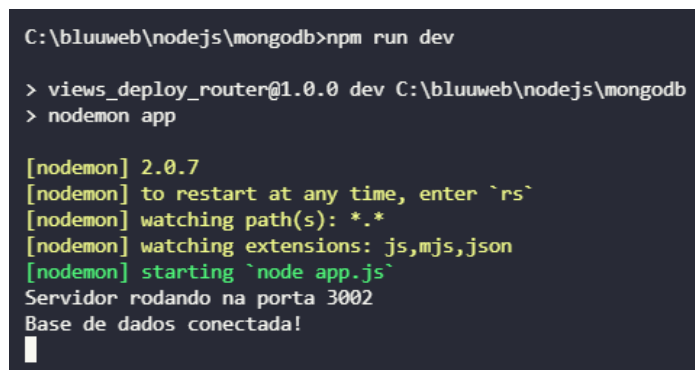
// Rotas Web
app.use('/', require('./routes/rotasWeb'));
app.use('/pets', require('./routes/routes-pets'));

// Rota não encontrada
app.use((req, res, next) => {
  // res.status(404).sendFile(__dirname + "/public/404.html");
  res.status(404).render("404", {bodyTitle: "Página não encontrada!", bodyContent: "Essa página não existe!"});
});

const port = process.env.PORT || 3000;

app.listen(port, () =>{
  console.log('Servidor rodando na porta ' + port);
});
```

npm run dev



```
C:\bluuweb\nodejs\mongodb>npm run dev

> views_deploy_router@1.0.0 dev C:\bluuweb\nodejs\mongodb
> nodemon app

[nodemon] 2.0.7
[nodemon] to restart at any time, enter `rs`
[nodemon] watching path(s): *.*
[nodemon] watching extensions: js,mjs,json
[nodemon] starting `node app.js`
Servidor rodando na porta 3002
Base de dados conectada!
```







.gitignore

node_modules

.env

Configurando as variáveis de ambiente no Heroku

<https://dashboard.heroku.com/apps/app-nodejs-bluuweb/settings>

Config Vars		Hide Config Vars
USER	betopinheiro	 
PASSWORD	DpuggJSVjb4qFQyx	 
DBNAME	veterinaria	 

Deploy - Atualizando

heroku login

git add .

```
beto1@DESKTOP-85HCHH0 MINGW64 /c/bluuweb/nodejs/mongodb (master)
$ git add .
warning: LF will be replaced by CRLF in package-lock.json.
The file will have its original line endings in your working directory
warning: LF will be replaced by CRLF in package.json.
The file will have its original line endings in your working directory
```

git commit -m "Variaveis de ambiente"

```
beto1@DESKTOP-85HCHH0 MINGW64 /c/bluuweb/nodejs/mongodb (master)
$ git commit -m "Variaveis de ambiente"
[master fd47d17] Variaveis de ambiente
8 files changed, 325 insertions(+), 20 deletions(-)
create mode 100644 models/Pet.js
create mode 100644 routes/routes-pets.js
create mode 100644 routes/routes-web.js
create mode 100644 views/pets/index.ejs
```

git push heroku master



Aula 12 - CRUD - Criar documentos em MongoDB

- Em `C:\bluuweb\nodejs` crie uma pasta chamada "`crud-mongo`"

```
C:\bluuweb\nodejs>md crud-mongo

C:\bluuweb\nodejs>dir
O volume na unidade C é W10-195
O Número de Série do Volume é 6047-7D0C

Pasta de C:\bluuweb\nodejs

02/06/2021  14:38    <DIR>          .
02/06/2021  14:38    <DIR>          ..
02/06/2021  14:38    <DIR>          crud-mongo
02/06/2021  03:17    <DIR>          Fundamentos
02/06/2021  14:21    <DIR>          mongodb
02/06/2021  04:12    <DIR>          rotas_basico
02/06/2021  11:47    <DIR>          views_deploy_router
                   0 arquivo(s)                0 bytes
                   7 pasta(s)          111.256.322.048 bytes disponíveis
```

`cd mongodb`

```
C:\bluuweb\nodejs>cd crud-mongo

C:\bluuweb\nodejs\crud-mongo>
```

- Copie todo o conteúdo da pasta `C:\bluuweb\nodejs\mongodb` para dentro da pasta `C:\bluuweb\nodejs\crud-mongo`

code .

Instalando body-parser

<https://www.npmjs.com/package/body-parser>

Installation

```
$ npm install body-parser
```

`npm install --save body-parser`

```
C:\bluuweb\nodejs\crud-mongo>npm install --save body-parser
npm WARN views_deploy_router@1.0.0 No description
npm WARN views_deploy_router@1.0.0 No repository field.
npm WARN optional SKIPPING OPTIONAL DEPENDENCY: fsevents@2.3.2 (node_modules\fsevents):
npm WARN notsup SKIPPING OPTIONAL DEPENDENCY: Unsupported platform for fsevents@2.3.2: wanted {"os":"darwin","arch":"any"} (current: {"os":"win32","arch":"x64"})

+ body-parser@1.19.0
updated 1 package and audited 217 packages in 10.942s

13 packages are looking for funding
  run `npm fund` for details

found 0 vulnerabilities
```

Uso

```
const bodyParser = require('body-parser')
```

Configurações

app.js

```
const express = require('express');
const app = express();
const mongoose = require('mongoose');
require('dotenv').config();
const bodyParser = require('body-parser')
const port = process.env.PORT || 3000;

// parse application/x-www-form-urlencoded
app.use(bodyParser.urlencoded({ extended: false }))

// parse application/json
app.use(bodyParser.json())

// Conexão a base de dados
const uri = `mongodb+srv://${process.env.USER}:${process.env.PASSWORD}@cluster0.cyvyx.mongodb.net/${process.env.DBNAME}?retryWrites=true&w=majority`;

mongoose.connect(uri, {useNewUrlParser: true, useUnifiedTopology: true})
.then(() => {
  console.log('Base de dados conectada!');
}).catch((error) => {
  console.log(error);
});

// motor de templates
app.set('view engine', 'ejs');
app.set('views', __dirname + '/views');

app.use(express.static(__dirname + '/public'));

// Rotas Web
app.use('/', require('./routes/routes-web'));
app.use('/pets', require('./routes/routes-pets'));

// Rota não encontrada
app.use((req, res, next) => {
  // res.status(404).sendFile(__dirname + "/public/404.html");
  res.status(404).render("404", {bodyTitle: "Página não encontrada!", bodyContent: "Essa página não existe!"});
});

app.listen(port, () =>{
  console.log('Servidor rodando na porta ' + port);
});
```

routes\routes-pets.js

```
const express = require('express');
const router = express.Router();
const Pet = require('../models/Pet');

// Rotas

router.get('/', async (req, res) => {

  try {
    const arrayPetsDB = await Pet.find();
    console.log(arrayPetsDB);

    res.render("pets", {arrayPets: arrayPetsDB});

  } catch (error) {
    console.log(error);
  }

});

router.get('/create', (req, res) => {
  res.render("pets/create");
});

router.post('/', async (req, res) => {
  const body = req.body

  try {
    //const petDB = new Pet(body);
    //await petDB.save();

    await Pet.create(body);

    res.redirect('/pets');
    // console.log("Pet cadastrado: ", petDB);
  } catch (error) {
    console.log(error);
  }

});

module.exports = router;
```

views\pets\create.ejs

```
<%- include("../templates/header", {bodyTitle: "Cadastrar um pet" }) %>

<div class="container mb-4 mt-2">

  <div class="row">
    <div class="col-md-2"></div>
    <div class="col-md-8">

      <h1 class="text-center">Cadastrar Pet</h1>

      <form action="/pets" method="POST">
        <label for="nome">Nome:</label>
        <input type="text" name="nome" placeholder="Digite o nome do Pet" class="form-control my-2"><br />

        <label for="especie">Espécie:</label>
        <select name="especie" class="form-control">
          <option value="cachorro">Cachorro</option>
          <option value="gato">Gato</option>
          <option value="coelho">Coelho</option>
          <option value="hamster">Hamster</option>
        </select><br />

        <label for="sexo">Sexo:</label>
        <select name="sexo" class="form-control">
          <option value="M">Macho</option>
          <option value="F">Fêmea</option>
        </select><br /><br />

        <button class="btn btn-success btn-block" type="submit">
          Cadastrar
        </button>
      </form>

    </div>>
  </div class="col-md-2"></div>
</div>

<%- include("../templates/footer") %>
```

views\templates\navbar.ejs

```
<nav class="navbar navbar-expand-md navbar-dark bg-dark">
  <a class="navbar-brand" href="/">Orion3</a>
  <button class="navbar-toggler" type="button" data-toggle="collapse" data-target="#navbarSupportedContent"
    aria-controls="navbarSupportedContent" aria-expanded="false" aria-label="Toggle navigation">
    <span class="navbar-toggler-icon"></span>
  </button>

  <div class="collapse navbar-collapse" id="navbarSupportedContent">
    <ul class="navbar-nav mr-auto">
      <li class="nav-item active">
        <a class="nav-link" href="/">Home <span class="sr-only">(current)</span></a>
      </li>
      <li class="nav-item">
        <a class="nav-link" href="/services">Serviços</a>
      </li>
      <li class="nav-item">
        <a class="nav-link" href="/contact">Contato</a>
      </li>
      <li class="nav-item dropdown">
        <a class="nav-link dropdown-toggle" href="#" id="navbarDropdown" role="button" data-toggle="dropdown"
          aria-haspopup="true" aria-expanded="false">
          Pets
        </a>
        <div class="dropdown-menu" aria-labelledby="navbarDropdown">
          <a class="dropdown-item" href="/pets">Listar</a>
          <a class="dropdown-item" href="/pets/create">Cadastrar</a>
        </div>
      </li>
    </ul>
    <form class="form-inline my-2 my-lg-0">
      <input class="form-control mr-sm-2" type="search" placeholder="Search" aria-label="Search">
      <button class="btn btn-outline-success my-2 my-sm-0" type="submit">Search</button>
    </form>
  </div>
</nav>
```

npm run dev

```
C:\bluuweb\nodejs\crud-mongo>npm run dev

> views_deploy_router@1.0.0 dev C:\bluuweb\nodejs\crud-mongo
> nodemon app

[nodemon] 2.0.7
[nodemon] to restart at any time, enter `rs`
[nodemon] watching path(s): *.*
[nodemon] watching extensions: js,mjs,json
[nodemon] starting `node app.js`
Servidor rodando na porta 3002
Base de dados conectada!
█
```


← → ↻ localhost:3002/pets/create

App Seletores de Formu... jQuery Validation Pl... javascript - Bootstr... Cadastro de pets Tabela de Cores... Lista de leitu...

Orion3 Home Serviços Contato Pets

Search Search

Cadastrar Pet

Nome:

Kim

Espécie:

Cachorro

Sexo:

Fêmea

Cadastrar

Orion3 - By: Roberto Pinheiro - Copyright © - Todos os direitos reservados.

```
{
  _id: 60b91d82e964d3093c00877d,
  nome: 'Kim',
  especie: 'cachorro',
  sexo: 'F',
  __v: 0
}
```

Orion3 Home Serviços Contato Pets

Search Search

Id	Nome	Espécie	Sexo	Ações
60b5d1b5d04d8e38770af3e1	Brisa	gato	Fêmea	Detalhes Editar
60b5d27bd04d8e38770af3e2	Fred	gato	Macho	Detalhes Editar
60b5d2cad04d8e38770af3e3	Téo	gato	Macho	Detalhes Editar
60b5d301d04d8e38770af3e4	Benjamin	gato	Macho	Detalhes Editar
60b5d349d04d8e38770af3e5	Tiquinho	gato	Macho	Detalhes Editar
60b5d393d04d8e38770af3e6	Silloeh	cachorro	Fêmea	Detalhes Editar
60b5d3cbd04d8e38770af3e7	Valentim	cachorro	Macho	Detalhes Editar
60b5d3efd04d8e38770af3e8	Roque	cachorro	Macho	Detalhes Editar
60b91d82e964d3093c00877d	Kim	cachorro	Fêmea	Detalhes Editar

Orion3 - By: Roberto Pinheiro - Copyright © - Todos os direitos reservados.

Cadastrar Pet

Nome:

Renatinho

Espécie:

Cachorro

Sexo:

Macho

Cadastrar

Orion3 - By: Roberto Pinheiro - Copyright © - Todos os direitos reservados.

Id	Nome	Espécie	Sexo	Ações
60b5d1b5d04d8e38770af3e1	Brisa	gato	Fêmea	Detalhes Editar
60b5d27bd04d8e38770af3e2	Fred	gato	Macho	Detalhes Editar
60b5d2cad04d8e38770af3e3	Téo	gato	Macho	Detalhes Editar
60b5d301d04d8e38770af3e4	Benjamin	gato	Macho	Detalhes Editar
60b5d349d04d8e38770af3e5	Tiquinho	gato	Macho	Detalhes Editar
60b5d393d04d8e38770af3e6	Silloeh	cachorro	Fêmea	Detalhes Editar
60b5d3cbd04d8e38770af3e7	Valentim	cachorro	Macho	Detalhes Editar
60b5d3efd04d8e38770af3e8	Roque	cachorro	Macho	Detalhes Editar
60b91d82e964d3093c00877d	Kim	cachorro	Fêmea	Detalhes Editar
60b91f33e964d3093c00877e	Renatinho	cachorro	Macho	Detalhes Editar

Orion3 - By: Roberto Pinheiro - Copyright © - Todos os direitos reservados.

```
{
  _id: 60b91f33e964d3093c00877e,
  nome: 'Renatinho',
  especie: 'cachorro',
  sexo: 'M',
  __v: 0
}
```

Aula 13 - CRUD - Exibir detalhes de um documento em MongoDB

routes\routes-pets.js

```
const express = require('express');
const router = express.Router();
const Pet = require('../models/Pet');

// Rotas

router.get('/', async (req, res) => {

  try {
    const arrayPetsDB = await Pet.find();
    console.log(arrayPetsDB);
    res.render("pets/index", {arrayPets: arrayPetsDB});
  } catch (error) {
    console.log(error);
  }
});

router.get('/create', (req, res) => {
  res.render("pets/create");
});

router.post('/', async (req, res) => {
  const body = req.body

  try {
    //const petDB = new Pet(body);
    //await petDB.save();

    await Pet.create(body);

    res.redirect('/pets');
    // console.log("Pet cadastrado: ", petDB);
  } catch (error) {
    console.log(error);
  }
});

router.get('/:id', async (req, res) => {
  const id = req.params.id
  try {
    const petDB = await Pet.findOne({ _id: id });
    console.log(petDB);
    res.render('pets/show', {
      pet: petDB,
      error: false
    });
  } catch (error) {
    console.log(error);
    res.render('pets/show', {
      error: true,
      message: 'Não encontrado o id selecionado!'
    });
  }
});

module.exports = router;
```

views\show.ejs

```
<%- include("../templates/header", {bodyTitle: "Detalhes do pet" }) %>

<div class="container mb-4 mt-2">
  <h1 class="text-center">Detalhes do pet</h1>
  <% if (error) { %>
    <h3 class="text-danger text-center mb-5">
      <%= message %>
    </h3>
    <a href="/pets" class="btn btn-success btn-block">Listar Pets</a>
  <% } %>

  <% if (!error) { %>
    <table class="table">
      <thead>
        <tr>
          <th scope="col">Id</th>
          <th scope="col">Nome</th>
          <th scope="col">Espécie</th>
          <th scope="col">Sexo</th>
          <th scope="col">Ação</th>
        </tr>
      </thead>
      <tbody>
        <tr>
          <th scope="row">
            <%= pet.id %>
          </th>
          <td>
            <%= pet.nome %>
          </td>
          <td>
            <%= pet.especie %>
          </td>
          <td>
            <% if (pet.sexo==='M') { %>
              Macho
            <% } %>

            <% if (pet.sexo==='F') { %>
              Fêmea
            <% } %>
          </td>
          <td>
            <button class="btn btn-danger btn-sm btn-block" id="btnEliminar" data-id="<%= pet.id %>">
              Excluir
            </button>
          </td>
        </tr>
      </tbody>
    </table>
  <% } %>
</div>

<%- include("../templates/footer") %>
```

views\pets\index.ejs

```
<%- include("../templates/header", {bodyTitle: "Home" }) %>

<div class="container mb-4 mt-2">

  <div class="row">
    <div class="col-md-1"></div>
    <div class="col-md-10">
      <table class="table">
        <thead>
          <tr>
            <th scope="col">Id</th>
            <th scope="col">Nome</th>
            <th scope="col">Espécie</th>
            <th scope="col">Sexo</th>
            <th scope="col">Ações</th>
          </tr>
        </thead>
        <tbody>
          <% arrayPets.forEach(pet=> { %>
            <tr>
              <th scope="row">
                <%= pet.id %>
              </th>
              <td>
                <%= pet.nome %>
              </td>
              <td>
                <%= pet.especie %>
              </td>
              <td>
                <% if (pet.sexo === 'M') { %>
                  Macho
                <% } %>
                <% if (pet.sexo === 'F') { %>
                  Fêmea
                <% } %>
              </td>
              <td><a class="btn btn-primary btn-sm" href="/pets/<%= pet.id %>">Detalhes</a>
                <a class="btn btn-warning btn-sm" href="#">Editar</a>
              </td>
            </tr>
          <% }} %>
        </tbody>
      </table>
    </div>>
    <div class="col-md-1"></div>
  </div>

  <%- include("../templates/footer") %>
```

views\pets\show.ejs

```
<%- include("../templates/header", {bodyTitle: "Detalhes do pet" }) %>

<div class="container mb-4 mt-2">
  <h1 class="text-center">Detalhes do pet</h1>
  <% if (error) { %>
    <h3 class="text-danger text-center mb-5">
      <%= message %>
    </h3>
    <a href="/pets" class="btn btn-success btn-block">Listar Pets</a>
  <% } %>

  <% if (!error) { %>
    <table class="table">
      <thead>
        <tr>
          <th scope="col">Id</th>
          <th scope="col">Nome</th>
          <th scope="col">Espécie</th>
          <th scope="col">Sexo</th>
          <th scope="col">Ação</th>
        </tr>
      </thead>
      <tbody>
        <tr>
          <th scope="row">
            <%= pet.id %>
          </th>
          <td>
            <%= pet.nome %>
          </td>
          <td>
            <%= pet.especie %>
          </td>
          <td>
            <% if (pet.sexo==='M') { %>
              Macho
            <% } %>

            <% if (pet.sexo==='F') { %>
              Fêmea
            <% } %>
          </td>
          <td>
            <button class="btn btn-danger btn-sm btn-block" id="btnEliminar" data-id="<%= pet.id %>">
              Excluir
            </button>
          </td>
        </tr>
      </tbody>
    </table>
  <% } %>
</div>

<%- include("../templates/footer") %>
```

Orion3

Home

Serviços

Contato

Pets

Search

Search

Id	Nome	Espécie	Sexo	Ações	>
60b5d1b5d04d8e38770af3e1	Brisa	gato	Fêmea	<div>Detalhes</div> <div>Editar</div>	
60b5d27bd04d8e38770af3e2	Fred	gato	Macho	<div>Detalhes</div> <div>Editar</div>	
60b5d2cad04d8e38770af3e3	Téo	gato	Macho	<div>Detalhes</div> <div>Editar</div>	
60b5d301d04d8e38770af3e4	Benjamin	gato	Macho	<div>Detalhes</div> <div>Editar</div>	
60b5d349d04d8e38770af3e5	Tiquinho	gato	Macho	<div>Detalhes</div> <div>Editar</div>	
60b5d393d04d8e38770af3e6	Silloeh	cachorro	Fêmea	<div>Detalhes</div> <div>Editar</div>	
60b5d3cbd04d8e38770af3e7	Valentim	cachorro	Macho	<div>Detalhes</div> <div>Editar</div>	
60b5d3efd04d8e38770af3e8	Roque	cachorro	Macho	<div>Detalhes</div> <div>Editar</div>	
60b91d82e964d3093c00877d	Kim	cachorro	Fêmea	<div>Detalhes</div> <div>Editar</div>	
60b91f33e964d3093c00877e	Renatinho	cachorro	Macho	<div>Detalhes</div> <div>Editar</div>	

Orion3 - By: Roberto Pinheiro - Copyright © - Todos os direitos reservados.

http://localhost:3002/pets/60b5d27bd04d8e38770af3e2

The screenshot shows a web browser at the URL `localhost:3002/pets/60b5d27bd04d8e38770af3e2`. The browser's address bar and tabs are visible. The website has a dark header with the logo "Orion3" and navigation links: "Home", "Serviços", "Contato", and "Pets". A search bar is on the right. The main content area is titled "Detalhes do pet" and contains a table with the following data:

Id	Nome	Espécie	Sexo	Ação
60b5d27bd04d8e38770af3e2	Fred	gato	Macho	Excluir

The footer of the page reads: "Orion3 - By: Roberto Pinheiro - Copyright © - Todos os direitos reservados."

http://localhost:3002/pets/11111111111111111111

The screenshot shows a web browser at the URL `localhost:3002/pets/11111111111111111111`. The website's header and navigation are the same as in the previous screenshot. The main content area is titled "Detalhes do pet" and displays a red error message: "Não encontrado o id selecionado!". Below the message is a green button labeled "Listar Pets". The footer of the page reads: "Orion3 - By: Roberto Pinheiro - Copyright © - Todos os direitos reservados."

Aula 14 - CRUD - Eliminar documentos em MongoDB

routes\routes-pets.js

```
const express = require('express');
const router = express.Router();
const Pet = require('../models/Pet');

router.get('/', async (req, res) => {
  try {
    const arrayPetsDB = await Pet.find();
    console.log(arrayPetsDB);
    res.render("pets", {arrayPets: arrayPetsDB});
  } catch (error) {
    console.log(error);
  }
});

router.get('/create', (req, res) => {
  res.render("create");
});

router.post('/', async (req, res) => {
  const body = req.body
  try {
    //const petDB = new Pet(body);
    //await petDB.save();
    await Pet.create(body);
    res.redirect('/pets');
    // console.log("Pet cadastrado: ", petDB);
  } catch (error) {
    console.log(error);
  }
});

router.get('/:id', async (req, res) => {
  const id = req.params.id
  try {
    const petDB = await Pet.findOne({ _id: id });
    console.log(petDB);
    res.render('show', {
      pet: petDB,
      error: false
    });
  } catch (error) {
    console.log(error);
    res.render('show', {
      error: true,
      message: 'Não encontrado o id selecionado!'
    });
  }
});
```

```

router.delete('/:id', async (req, res) => {
  const id = req.params.id;
  console.log('id desde backend', id);
  try {
    const petDB = await Pet.findByIdAndDelete({ _id: id });
    console.log(petDB);
    if(!petDB){
      res.json({
        status: false,
        message: "Erro ao excluir o documento!"
      });
    } else {
      res.json({
        status: true,
        message: "Eliminado com sucesso!"
      });
    }
  } catch (error) {
    console.log(error);
  }
});

```

```

module.exports = router;

```

views\show.ejs

```

<%- include("../templates/header", {bodyTitle: "Detalhes do pet" }) %>

```

```

<div class="container mb-4 mt-2">
  <h1 class="text-center">Detalhes do pet</h1>
  <% if (error) { %>
    <h3 class="text-danger text-center mb-5">
      <%= message %>
    </h3>
    <a href="/pets" class="btn btn-success btn-block">Listar Pets</a>
  <% } %>

```

```

<% if (!error) { %>
  <table class="table">
    <thead>
      <tr>
        <th scope="col">Id</th>
        <th scope="col">Nome</th>
        <th scope="col">Espécie</th>
        <th scope="col">Sexo</th>
        <th scope="col">Ação</th>
      </tr>
    </thead>
    <tbody>
      <tr>
        <th scope="row">
          <%= pet.id %>
        </th>
        <td>
          <%= pet.nome %>
        </td>
        <td>
          <%= pet.especie %>

```

```

    </td>
    <td>
      <% if (pet.sexo==='M' ) { %>
        Macho
      <% } %>

      <% if (pet.sexo==='F' ) { %>
        Fêmea
      <% } %>
    </td>
    <td>
      <button class="btn btn-danger btn-sm btn-block" id="btnEliminar" data-id="<%= pet.id %>">
        Excluír
      </button>
    </td>
  </tr>
</tbody>
</table>
<% } %>
</div>

```

```

<%- include("../templates/footer") %>

```

```

<script>
const btnEliminar = document.querySelector('.btn-danger')
btnEliminar.addEventListener('click', async () => {
  console.log(btnEliminar.dataset.id)
  try {
    const data = await fetch(`/pets/${btnEliminar.dataset.id}`, {
      method: 'delete'
    })
    const res = await data.json()
    console.log(res)
    if (res.status) {
      window.location.href = '/pets'
    } else {
      console.log(res)
    }
  } catch (error) {
    console.log(error)
  }
})
</script>

```

http://localhost:3002/pets/60b91f33e964d3093c00877e

← → ↻ ⓘ localhost:3002/pets/60b91f33e964d3093c00877e ☆ (x) 📄 🌟 📌 ⋮

Apps Seletores de Formul... jQuery Validation Pl... javascript - Bootstr... Cadastro de posts Tabela de Cares... ** Lista de leitura

Orion3 Home Serviços Contato Pets ▾ Search Search

Detalhes do pet

Id	Nome	Espécie	Sexo	Ação
60b91f33e964d3093c00877e	Renatinho	cachorro	Macho	Excluir

Orion3 - By: Roberto Pinheiro - Copyright © - Todos os direitos reservados.

- Clique no botão "Excluir"

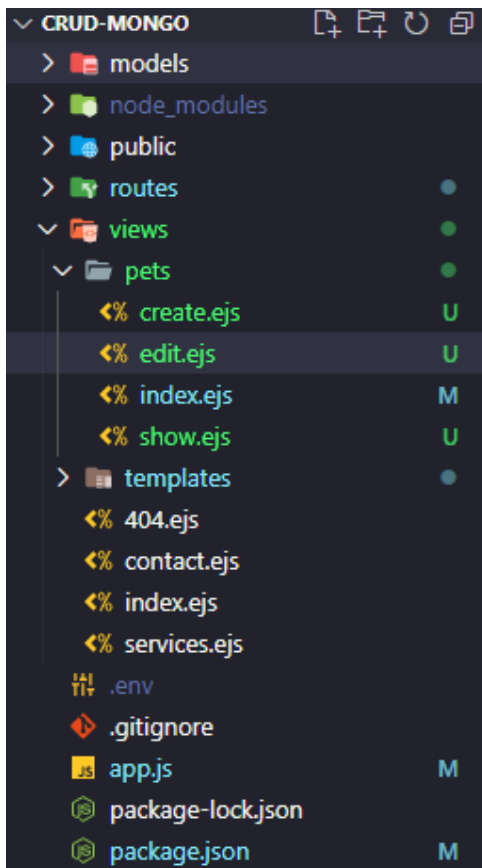
Orion3 Home Serviços Contato Pets ▾ Search Search

Id	Nome	Espécie	Sexo	Ações
60b5d1b5d04d8e38770af3e1	Brisa	gato	Fêmea	Detalhes Editar
60b5d27bd04d8e38770af3e2	Fred	gato	Macho	Detalhes Editar
60b5d2cad04d8e38770af3e3	Téo	gato	Macho	Detalhes Editar
60b5d301d04d8e38770af3e4	Benjamin	gato	Macho	Detalhes Editar
60b5d349d04d8e38770af3e5	Tiquinho	gato	Macho	Detalhes Editar
60b5d393d04d8e38770af3e6	Silloeh	cachorro	Fêmea	Detalhes Editar
60b5d3cbd04d8e38770af3e7	Valentim	cachorro	Macho	Detalhes Editar
60b5d3efd04d8e38770af3e8	Roque	cachorro	Macho	Detalhes Editar
60b91d82e964d3093c00877d	Kim	cachorro	Fêmea	Detalhes Editar

Orion3 - By: Roberto Pinheiro - Copyright © - Todos os direitos reservados.

Aula 15 - CRUD - Editar documento MongoDB e Deploy a Heroku

- Na pasta `views\pets` adicione um arquivo chamado `edit.ejs`:



`routes\routes-pets.js`

```
const express = require('express');
const router = express.Router();
const Pet = require('../models/Pet');

// Rotas

router.get('/', async (req, res) => {

  try {
    const arrayPetsDB = await Pet.find();
    console.log(arrayPetsDB);
    res.render("pets/index", {arrayPets: arrayPetsDB});
  } catch (error) {
    console.log(error);
  }
});

router.get('/create', (req, res) => {
  res.render("pets/create");
});
```

```

router.post('/', async (req, res) => {
  const body = req.body

  try {
    //const petDB = new Pet(body);
    //await petDB.save();

    await Pet.create(body);

    res.redirect('/pets');
    // console.log("Pet cadastrado: ", petDB);
  } catch (error) {
    console.log(error);
  }

});

router.get('/:id', async (req, res) => {
  const id = req.params.id
  try {
    const petDB = await Pet.findOne({ _id: id });
    console.log(petDB);
    res.render('pets/show', {
      pet: petDB,
      error: false
    });
  } catch (error) {
    console.log(error);
    res.render('pets/show', {
      error: true,
      message: 'Não encontrado o id selecionado!'
    });
  }
});

router.delete('/:id', async (req, res) => {
  const id = req.params.id;
  console.log('id desde backend', id);
  try {
    const petDB = await Pet.findByIdAndDelete({ _id: id });
    console.log(petDB);
    if(!petDB){
      res.json({
        status: false,
        message: "Erro ao excluir o documento!"
      });
    } else {
      res.json({
        status: true,
        message: "Eliminado com sucesso!"
      });
    }
  } catch (error) {
    console.log(error);
  }
});

```

```
router.get('/edit/:id', async (req, res) => {
  const id = req.params.id
  try {
    const petDB = await Pet.findOne({ _id: id });
    console.log(petDB);
    res.render('pets/edit', {
      pet: petDB,
      error: false
    });
  } catch (error) {
    console.log(error);
    res.render('pets/edit', {
      error: true,
      message: 'Não encontrado o id selecionado!'
    });
  }
});

router.put('/:id', async (req, res) => {
  const id = req.params.id
  const body = req.body

  try {

    const petDB = await Pet.findByIdAndUpdate(id, body, { useFindAndModify: false })
    console.log(petDB)

    res.json({
      estado: true,
      mensaje: 'Editado'
    })

  } catch (error) {
    console.log(error)

    res.json({
      estado: false,
      mensaje: 'Fallamos!!'
    })
  }
})

module.exports = router;
```

views\pets\edit.ejs

```
<%- include("../templates/header", {bodyTitle: "Detalhes do pet" }) %>

<div class="container mb-4 mt-2">

  <div class="row">
    <div class="col-md-2"></div>
    <div class="col-md-8">
      <h1 class="text-center mb-3">Atualização de dados do pet</h1>
      <% if (error) { %>
        <h3 class="text-danger text-center mb-5">
          <%= message %>
        </h3>
        <a href="/pets" class="btn btn-success btn-block">Listar Pets</a>
      <% } %>

      <% if (!error) { %>
        <form id="formularioEditar" data-id="<%= pet.id %>">
          <label for="nome">Nome</label>
          <input type="text" class="form-control my-2" name="nome" value="<%= pet.nome %>"
id="nomeInput"><br />

          <label for="especie">Espécie:</label>
          <select name="especie" id="especieSelect" class="form-control">

            <% if (pet.especie==='cachorro' ) { %>
              <option value="cachorro" selected="selected">Cachorro</option>
            <% } else { %>
              <option value="cachorro">Cachorro</option>
            <% } %>

            <% if (pet.especie==='gato' ) { %>
              <option value="gato" selected="selected">Gato</option>
            <% } else { %>
              <option value="gato">Gato</option>
            <% } %>

            <% if (pet.especie==='coelho' ) { %>
              <option value="coelho" selected="selected">Coelho</option>
            <% } else { %>
              <option value="coelho">Coelho</option>
            <% } %>

            <% if (pet.especie==='hamster' ) { %>
              <option value="hamster" selected="selected">Hamster</option>
            <% } else { %>
              <option value="hamster">Hamster</option>
            <% } %>

          </select><br /><br />

          <label for="sexo">Sexo:</label>
          <select name="sexo" id="sexoSelect" class="form-control">
            <% if (pet.sexo==='F' ) { %>
              <option value="F" selected="selected">Fêmea</option>
            <% } else { %>
              <option value="F">Fêmea</option>
            <% } %>

          </select>
        </form>
      <% } %>
    </div>
  </div>
</div>
```



```

        <% if (pet.sexo==='M' ) { %>
            <option value="M" selected="selected">Macho</option>
        <% } else { %>
            <option value="M">Macho</option>
        <% } %>
    </select><br /><br />

    <button class="btn btn-warning btn-block type=" submit">
        Gravar
    </button>
</form>
<% } %>
</div>
</div>
<div class="col-md-2"></div>
</div>

```

```

<%- include("../templates/footer") %>

```

```

<script>

```

```

const formularioEditar = document.querySelector('#formularioEditar')

```

```

formularioEditar.addEventListener('submit', async (e) => {
    e.preventDefault()

```

```

    const nome = formularioEditar.elements['nome'].value
    const especie = document.querySelector('#especieSelect').value
    const sexo = document.querySelector('#sexoSelect').value
    const id = formularioEditar.dataset.id

```

```

    try {

```

```

        const data = await fetch(`/pets/${id}`, {
            method: 'put',
            headers: {
                'Content-Type': 'application/json'
            },
            body: JSON.stringify({ nome, especie, sexo })
        })

```

```

        const res = await data.json()

```

```

        if (res.estado) {
            window.location.href = '/pets'
        } else {
            console.log(res)
        }
    } catch (error) {
        console.log(error)
    }
}

```

```

</script>

```

http://localhost:3002/pets

Orion3 Home Serviços Contato Pets ▾					Search <input type="text"/>		Search
Id	Nome	Espécie	Sexo	Ações			
60b5d1b5d04d8e38770af3e1	Brisa	gato	Fêmea	Detalhes Editar			
60b5d27bd04d8e38770af3e2	Fred	gato	Macho	Detalhes Editar			
60b5d2cad04d8e38770af3e3	Téo	gato	Macho	Detalhes Editar			
60b5d301d04d8e38770af3e4	Benjamin	gato	Macho	Detalhes Editar			
60b5d349d04d8e38770af3e5	Tiquinho	gato	Macho	Detalhes Editar			
60b5d393d04d8e38770af3e6	Silloeh	cachorro	Fêmea	Detalhes Editar			
60b5d3cbd04d8e38770af3e7	Valentim	cachorro	Macho	Detalhes Editar			
60b5d3efd04d8e38770af3e8	Roque	cachorro	Macho	Detalhes Editar			
60b91d82e964d3093c00877d	Kim	cachorro	Fêmea	Detalhes Editar			

Orion3 - By: Roberto Pinheiro - Copyright © - Todos os direitos reservados.

http://localhost:3002/pets/edit/60b5d27bd04d8e38770af3e2

← → ↻ ⓘ localhost:3002/pets/edit/60b5d27bd04d8e38770af3e2 🔍 ☆ (w) 📄 🌟 ⚙️ ⋮

📱 Apps 📄 Seletores de Formu... 📄 jQuery Validation PL... 📄 javascript - Bootstr... 📄 Cadastro de posts 📄 Tabela de Cores... 📄 Lista de leitura

Orion3 Home Serviços Contato Pets ▾					Search <input type="text"/>		Search		
Atualização de dados do pet									
Nome									
<input type="text" value="Fred"/>									
Espécie									
<div>Gato ▾</div>									
Sexo									
<div>Macho ▾</div>									
<div>Gravar</div>									

Orion3 - By: Roberto Pinheiro - Copyright © - Todos os direitos reservados.

- Altere o nome de "Fred" para "Frederico":

localhost:3002/pets/edit/60b5d27bd04d8e38770af3e2

App: Seletores de Formas... jQuery Validation PL... javascript - Bootstr... Cadastro de posts Tabela de Cores... Lista de feitura

Orion3 Home Serviços Contato Pets

Search Search

Atualização de dados do pet

Nome

Frederico

Espécie

Gato

Sexo

Macho

Gravar

Orion3 - By: Roberto Pinheiro - Copyright © - Todos os direitos reservados.

- Clique no botão "Gravar":

Orion3 Home Serviços Contato Pets

Search Search

Id	Nome	Espécie	Sexo	Ações
60b5d1b5d04d8e38770af3e1	Brisa	gato	Fêmea	Detalhes Editar
60b5d27bd04d8e38770af3e2	Frederico	gato	Macho	Detalhes Editar
60b5d2cad04d8e38770af3e3	Téo	gato	Macho	Detalhes Editar
60b5d301d04d8e38770af3e4	Benjamin	gato	Macho	Detalhes Editar
60b5d349d04d8e38770af3e5	Tiquinho	gato	Macho	Detalhes Editar
60b5d393d04d8e38770af3e6	Silloeh	cachorro	Fêmea	Detalhes Editar
60b5d3cbd04d8e38770af3e7	Valentim	cachorro	Macho	Detalhes Editar
60b5d3efd04d8e38770af3e8	Roque	cachorro	Macho	Detalhes Editar
60b91d82e964d3093c00877d	Kim	cachorro	Fêmea	Detalhes Editar

Orion3 - By: Roberto Pinheiro - Copyright © - Todos os direitos reservados.

Deploy

git status

```
beto1@DESKTOP-85HCHH0 MINGW64 /c/bluuweb/nodejs/crud-mongo (master)
$ git status
On branch master
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
    modified:   app.js
    modified:   package.json
    modified:   routes/routes-pets.js
    modified:   views/pets/index.ejs
    modified:   views/templates/navbar.ejs

Untracked files:
  (use "git add <file>..." to include in what will be committed)
    views/pets/create.ejs
    views/pets/edit.ejs
    views/pets/show.ejs

no changes added to commit (use "git add" and/or "git commit -a")
```

git add .

```
beto1@DESKTOP-85HCHH0 MINGW64 /c/bluuweb/nodejs/crud-mongo (master)
$ git add .
```

git status

```
beto1@DESKTOP-85HCHH0 MINGW64 /c/bluuweb/nodejs/crud-mongo (master)
$ git status
On branch master
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    modified:   app.js
    modified:   package.json
    modified:   routes/routes-pets.js
    new file:   views/pets/create.ejs
    new file:   views/pets/edit.ejs
    modified:   views/pets/index.ejs
    new file:   views/pets/show.ejs
    modified:   views/templates/navbar.ejs
```

git commit -m "CRUD Mongo"

```
beto1@DESKTOP-85HCHH0 MINGW64 /c/bluuweb/nodejs/crud-mongo (master)
$ git commit -m "CRUD Mongo"
[master d82c7d1] CRUD Mongo
8 files changed, 397 insertions(+), 35 deletions(-)
create mode 100644 views/pets/create.ejs
create mode 100644 views/pets/edit.ejs
rewrite views/pets/index.ejs (93%)
create mode 100644 views/pets/show.ejs
```

git push heroku master