



UNISANTA

**Hospedando a solução
na nuvem**

UNISANTA

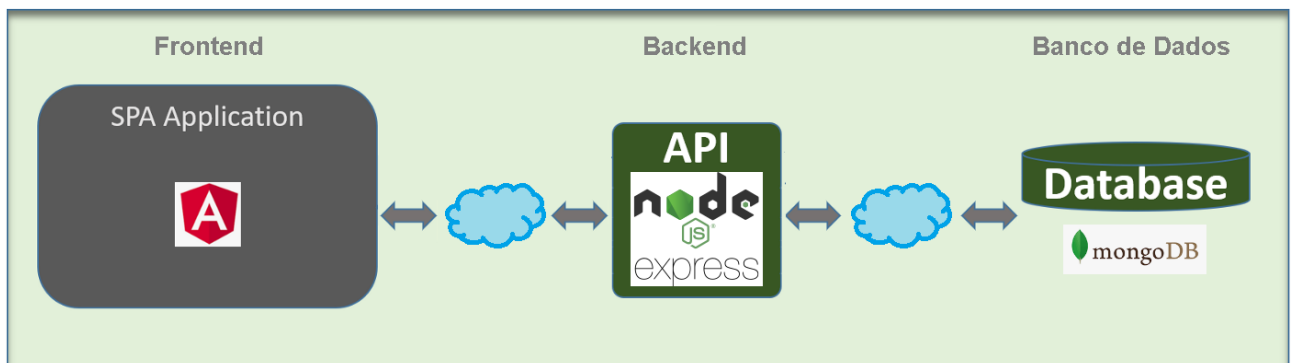
Índice

Revendo a arquitetura da solução.....	3
Definindo a hospedagem.....	4
Hospedagem do Backend.....	5
Preparando a API para rodar no Heroku.....	5
Abrindo sua conta no Heroku.....	6
Criando um novo App no Heroku.....	6
Testando no Browser.....	8
Hospedagem do Frontend.....	9
Preparando o TODOapp para comunicar com a API.....	9
Realizando o Build do App.....	10
Modificando o local base do App.....	11
Transferindo para o Hostinger.....	11
Considerações finais.....	12
Histórico de revisões.....	13

Revendo a arquitetura da solução

Chegou o grande momento onde vamos executar nossa solução em um ambiente de produção (Até o momento estávamos executando nossa solução em um ambiente de desenvolvimento).

Dessa forma é importante revermos a arquitetura definida originalmente:

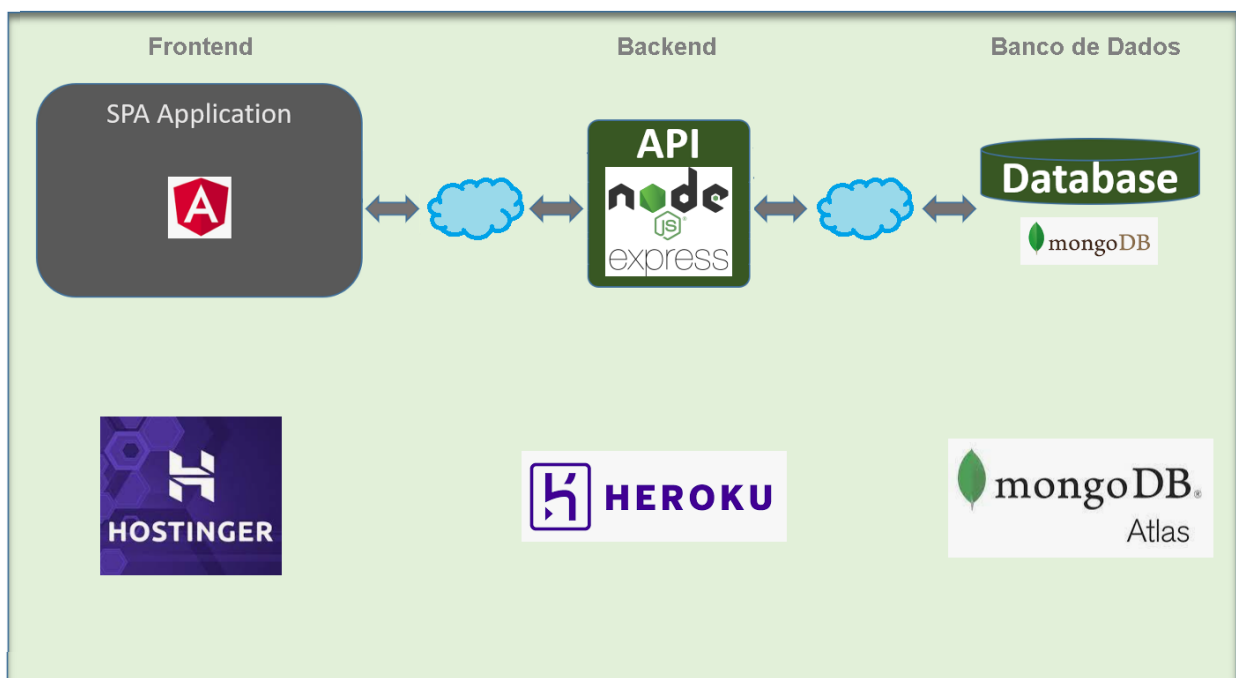


Como podemos ver acima, nossa solução está dividida em três partes básicas:

- Frontend
Representado pelo nosso TODOapp desenvolvido com o Framework Angular
- Backend
Representado pela nossa API desenvolvida com o Framework Express
- Banco de Dados
Utilizando o banco MongoDB

Definindo a hospedagem

Segue a definição de onde será hospedada cada parte da solução:



- **Frontend**
Será hospedado em um servidor HTTP simples da Hostinger. Servirá simplesmente para baixar a SPA (Single Page Application) para o browser do cliente.
- **Backend**
Será hospedado no Heroku, que oferece um servidor de aplicações, rodando o nosso código da API.
- **Banco de Dados**
Já está hospedado no MongoDB Atlas. Será mantida a mesma conta.



Hospedagem do Backend

Considerando que nosso Banco de Dados já está hospedado, vamos partir para hospedar o nosso Backend. É importante saber que esta hospedagem será feita somente para efeito de demonstração e entendimento deste tipo de processo. As ações para realização deste processo podem variar muito ao longo do tempo e dependem exclusivamente dos procedimentos definidos por cada provedor. Em nosso exemplo, usaremos o Heroku, que é uma plataforma de cloud de bastante sucesso e amplamente conhecida pela comunidade. Neste momento, diversas outras plataformas existem no mercado para a mesma finalidade como: Amazon Web Services (AWS), Google Cloud Platform, Microsoft Azure, IBM Cloud, Oracle Cloud etc.

Preparando a API para rodar no Heroku

Ao colocar nossa API em um ambiente cloud, temos que entender como o mesmo fará para executar nossa aplicação sob demanda. No caso do Heroku, ao identificar que nossa aplicação é desenvolvida em Node Express, o Heroku possui a configuração padrão de executar nossa aplicação utilizando o comando:

`npm start`

O comando acima executa um script chamado de "**start**", declarado dentro do arquivo "**package.json**" de nosso projeto.

Dessa forma, vamos modificar esse arquivo, inserindo no conteúdo do script, o comando de inicialização que utilizávamos para executar nossa API no ambiente de desenvolvimento.

Cabe porém observar duas pequenas diferenças:

- Por estarmos executando no ambiente de desenvolvimento, utilizávamos o comando "**nodemon**". No ambiente de produção vamos substituir esse comando por "**node**".
- Como o script necessita ficar entre aspas duplas, vamos substituir as aspas originais da URL por aspas simples.

Segue destacado em verde, a alteração que será necessária realizar no arquivo "**package.json**":

```
{
  "name": "api",
  "version": "1.0.0",
  "description": "",
  "main": "index.js",
  "scripts": {
    "test": "echo \"Error: no test specified\" && exit 1",
    "start": "node index 'sua URL aqui'"
  },
  "author": "",
  "license": "ISC",
  "dependencies": {
    "express": "^4.18.1",
    "mongoose": "^6.4.6"
  }
}
```

Repare que será necessário incluir uma vírgula ao final do script "test"

Abrindo sua conta no Heroku

O Heroku oferece diversos planos conforme sua necessidade de uso. Uma grande vantagem é que ele oferece também um plano básico gratuito que atenderá com muita folga as nossas necessidades durante aprendizado. Dessa forma, entre no site: <https://www.heroku.com/> e crie uma conta FREE pra você.

Criando um novo App no Heroku

Clique no logo do Heroku para aparecer uma página chamada "Personal".

No canto direito você verá um botão chamado "NEW". Ao pressionar este botão selecione a opção "Create New App".

O Heroku irá perguntar qual o nome do App. Entre com o seguinte nome:

apitarefasNNNNxxxxxx

onde: **NNNN** será seu nome e **xxxxxx** será seu RA (tudo minúsculo). Utilize este mesmo procedimento até o final deste tutorial.

Caso surja alguma mensagem dizendo que o nome já existe, então acrescente seu sobrenome ou outras informações que permitam criar um endereço exclusivo.

Em seguida, pressione o botão "**Create App**".

Escolha a opção "**Heroku git**" como sendo o método de **Deployment**. Lembrando que instalamos o Git em nosso computador na primeira aula deste curso.

Abra o prompt de comando, dentro da pasta de nossa API, e execute o seguinte comando para instalar o Heroku CLI (caso ele nunca tenha sido instalado antes).

```
npm install -g heroku
```

Verifique se o Heroku foi instalado com sucesso através do seguinte comando:

```
heroku --version
```

Deve aparecer o número da versão instalada.

Realize o login pela linha de comando:

```
heroku login
```

Pressione ENTER para ele redirecioná-lo para uma página de login via browser. Ao concluir o login, volte ao prompt de comandos e perceba que o login foi reconhecido.

Defina um novo repositório **git** local, utilizando o seguinte comando:

```
git init
```

Garanta que seu email e nome de usuário estejam definidos para o git, executando os seguintes comandos:

```
git config --global user.email "seu email"
```

```
git config --global user.name "seu nome"
```

Associe o seu projeto local com o App criado remotamente no Heroku:

```
heroku git:remote -a apitarefasNNNNxxxxxx
```

Defina que todos os arquivos fazem parte do seu repositório:

```
git add .
```

Defina um commit da versão inicial do seu sistema:

```
git commit -am "versao inicial"
```

Transfira seu projeto para o repositório remoto:

```
git push heroku master
```



Testando no Browser

Abra o browser utilizando o endereço:

`https://apitarefasNNNNxxxxxx.herokuapp.com/api/getAll`

Se tudo estiver funcionando perfeitamente, o browser apresentará um json de todas as tarefas gravadas em seu banco de dados.

Hospedagem do Frontend

Por último, resta fazermos a hospedagem do Frontend para que nossa solução esteja disponibilizada 100% na nuvem.

Preparando o TODOapp para comunicar com a API

Agora que temos nossa API hospedada na nuvem, basta atualizar sua nova localização dentro do TODOapp.

Durante o desenvolvimento nossa API estava disponível no endereço:

`http://localhost:3000`

Ao hospedarmos a API no Heroku, seu novo endereço passou a ser:

`https://apitarefasNNNNxxxxxx.herokuapp.com`

Dessa forma, segue destacado em verde a alteração que deve ser feita no arquivo "**app.component.ts**" de nosso App:

```
...
export class AppComponent {
  title = 'TODOapp';
  arrayDeTarefas: Tarefa[] = [];
  apiURL: string;
  constructor(private http: HttpClient) {
    this.apiURL = 'https://apitarefasNNNNxxxxxx.herokuapp.com';
    this.READ_tarefas();
  }

  CREATE_tarefa(descricaoNovaTarefa: string) {
    ...
  }
}
```

Realizando o Build do App

Abra o prompt de comando dentro da pasta TODOapp.

Execute o seguinte comando para realizar o Build da aplicação:

ng build

O angular irá compilar seu projeto apresentando o seguinte resultado:

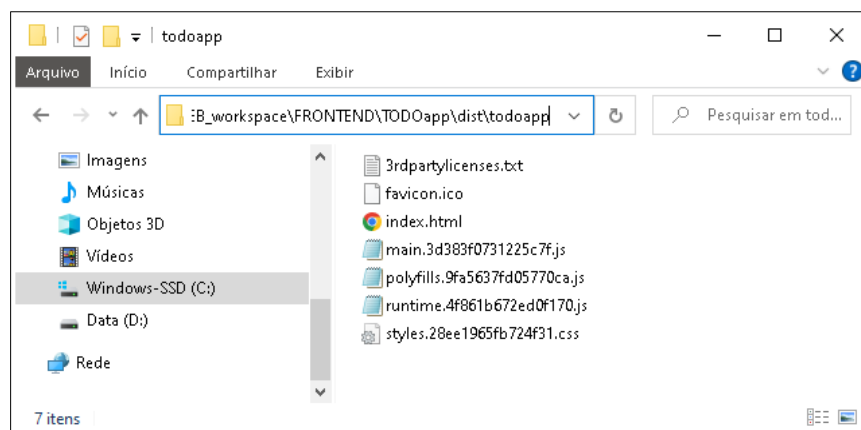
```
C:\WINDOWS\system32\cmd.exe
Microsoft Windows [versão 10.0.19043.1826]
(c) Microsoft Corporation. Todos os direitos reservados.

C:\WEB_workspace\FRONTEND\TODOapp>ng build
✓ Browser application bundle generation complete.
✓ Copying assets complete.
✓ Index html generation complete.

Initial Chunk Files | Names | Raw Size | Estimated Transfer Size
main.3d383f0731225c7f.js | main | 125.57 kB | 35.95 kB
polyfills.9fa5637fd05770ca.js | polyfills | 33.07 kB | 10.63 kB
runtime.4f861b672ed0f170.js | runtime | 1.04 kB | 596 bytes
styles.28ee1965fb724f31.css | styles | 648 bytes | 243 bytes
| Initial Total | 160.31 kB | 47.39 kB

Build at: 2022-07-21T14:16:51.739Z - Hash: 56dc8a2451dd7dae - Time: 6348ms
C:\WEB_workspace\FRONTEND\TODOapp>
```

Os arquivos finais da construção o seu projeto vão estar disponíveis dentro da pasta "**C:\WEB_workspace\FRONTEND\TODOapp\dist\todoapp**". Ex:



Esses são os arquivos que deveremos hospedar no servidor do Frontend.

Modificando o local base do App

Considerando que o Angular prepara o App para rodar na pasta raiz de um site, vamos fazer uma pequena mudança para que possa ser colocado em uma sub-pasta. Para isso, edite o arquivo "**index.html**" modificando o elemento:

```
<base href="/">
```

Por:

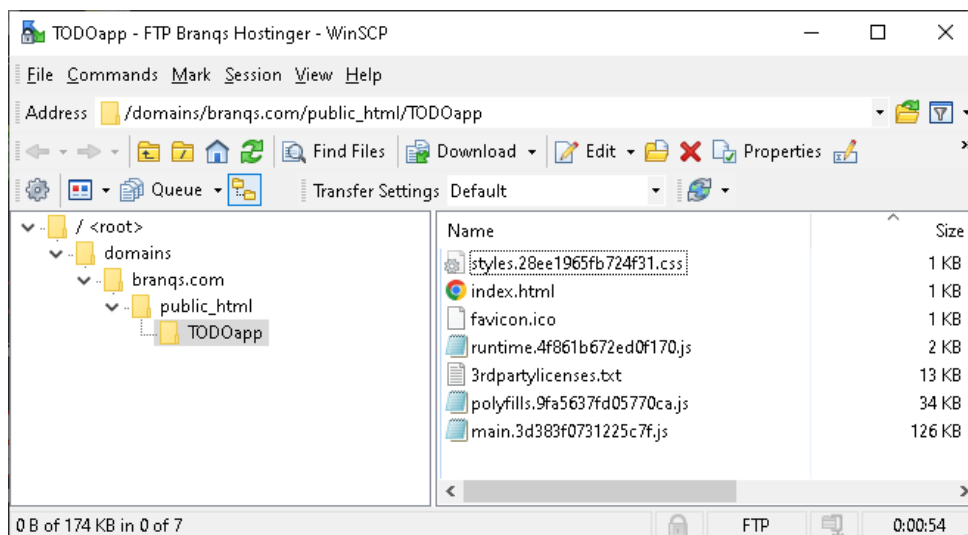
```
<base href=". /">
```

Ou seja, basta inserir um ponto antes da barra.

Transferindo para o Hostinger

Diferentemente do Backend, o Frontend não necessita de uma configuração especial de recursos, bastando realizar uma simples transferência de arquivos para o servidor.

Os diversos servidores do mercado oferecem diferentes métodos para transferência desses arquivos. Em nosso exemplo, vou estabelecer a conexão FTP com uma conta que possuo no Hostinger para hospedar o domínio "branqs.com". Em seguida vou criar uma sub-pasta do domínio chamada de "TODOapp" e arrastar para dentro dela os arquivos criados durante o build do App. Ex:



Pronto, neste exemplo, nosso App já está pronto para uso a partir da URL:

<http://branqs.com/TODOapp/>



Considerações finais

Se você chegou até aqui... PARABÉNS !!!

Você conseguiu publicar uma solução WEB contemplando FRONTEND, BACKEND e BANCO DE DADOS, ou seja, o famoso FULL STACK.

Seu App agora pode ser executado rapidamente a partir do browser de qualquer tipo de dispositivo como: Computadores, Smartphones, Tablets etc. Tudo isso sem necessitar nem mesmo de instalação prévia.

Espero muito que você tenha gostado do conteúdo e que sirva de inspiração para você pesquisar detalhes de funcionamento e melhorias que podem ser implementadas, como por exemplo Autenticação de usuários e Autorizações de Acesso.

Muito obrigado por todos os momentos em que passamos juntos nesta matéria e parabéns por você se dedicar aos estudos para obter aprimoramento profissional. São pessoas assim que ajudam a criar um mundo cada vez melhor.

SUCESSO !!!!!!!!!!!

Histórico de revisões

Revisão: 00
Data: 19/07/2022
Descrição das alterações:
Documento original