

FIC 2 – Módulo IV

Aula 5 – API Rest

Node.js intermediário, CRUD
com sequelize, cookies e
sessões.

Antonio Izo Júnior

Agenda

- O que é API?
- O que é REST?
- Arquitetura REST.
- Funcionamento básico.
- Benefícios da arquitetura REST.

• O que é API?

- API é um acrônimo para Application Programming Interface ou Interface de Programação de Aplicativos.
- É um conjunto de especificações que definem de que forma as aplicações irão se comunicar.

• O que é API?

- Com a API você tem uma interface para que um sistema se comunique com, compartilhando suas ações e ferramentas.
- As APIs são muito úteis e tem diversos benefícios, além disso proporcionam uma troca de informações muito segura, já que somente o proprietário da aplicação define quais informações estarão disponíveis.



● O que é API?

- A autenticação também é importante para proteger de acessos indevidos à informação.
- Para transmitir as informações entre aplicações, geralmente é utilizado o JSON.
- O JSON é muito utilizado para retornar os dados das APIs baseadas em web. Além disso, esse conceito ocupa pouco espaço e é fácil de transportar via rede.



O que é REST?

- Representation State Transfer ou Transferência Representacional de Estado.
- Separa o backend do frontend.
- Nesse caso, o backend depois de receber e processar as requisições, responde ao cliente as informações de forma estruturada (normalmente no padrão JSON).



O que é REST?

- Com isso, o mesmo backend pode atender diversos frontends, podendo ser web, mobile, ou até outras aplicações backends.

Arquitetura REST

- O REST é um modelo de arquitetura com algumas características que podemos citar:
- Client-server: separar as responsabilidades.
- Cache: suporte para uso de cache na API.

Arquitetura REST

- Interface Uniforme: uma espécie de contrato entre o servidor e o cliente, que define os recursos a serem consumidos, com método (HTTP), servidor, recurso (entidade) e parâmetro.
- Código sob demanda: permite que as funcionalidades do cliente sejam estendidas.

Funcionamento básico do REST

1. A requisição é feita por um cliente, que pode ser um browser, um app mobile, etc.
2. Resposta retornada através de uma estrutura de dados (JSON).
3. Cliente recebe a resposta e processa o resultado.
4. As rotas utilizam métodos HTTP, sendo compostas por: método, servidor, recurso (entidade) e parâmetros, ou seja, a rota é o endereço completo da requisição:



Benefícios da Arquitetura REST

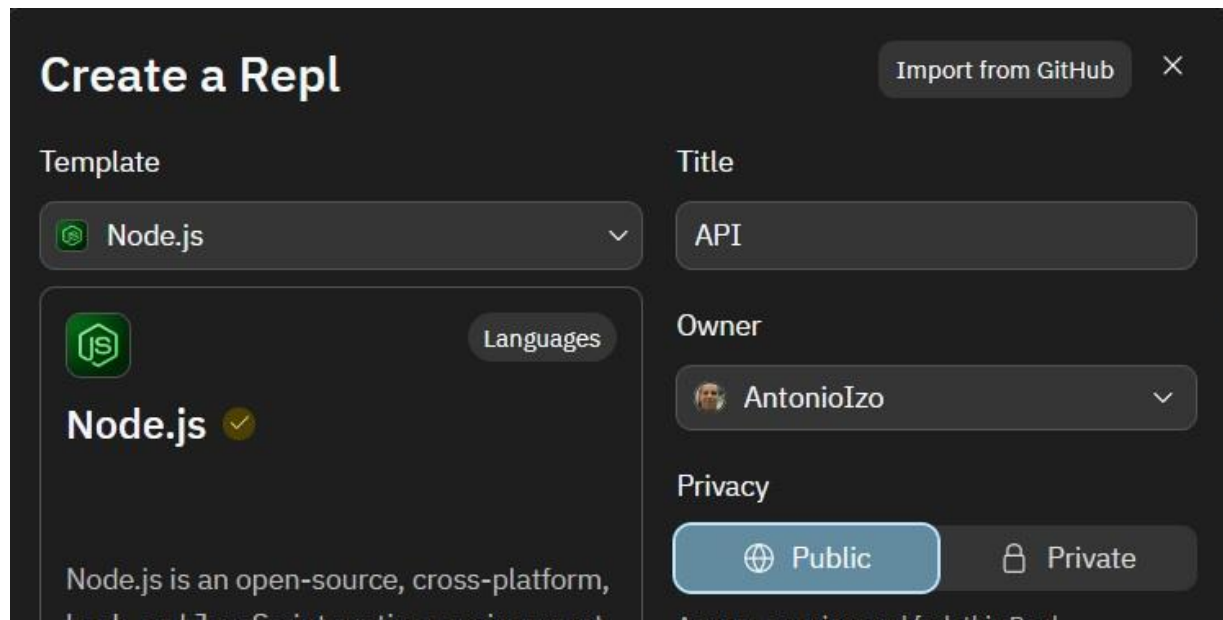
- Múltiplos clientes (frontend) para usufruir de um mesmo backend.
- Protocolo de comunicação padronizado, sendo a mesma estrutura para web e mobile.
- Comunicação com serviços externos.
- A padronização com o JSON faz com que qualquer tecnologia frontend e consiga consumir os dados do backend:

Criando uma API REST básica

Vamos praticar?

Criando uma API REST básica

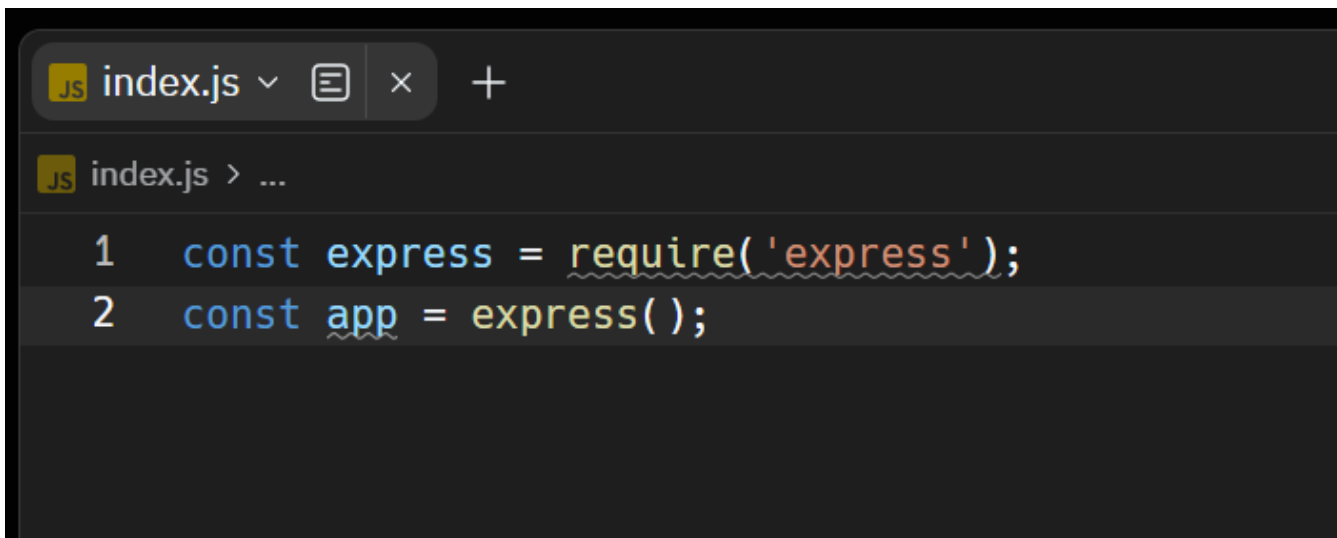
Abra o replit e crie um novo projeto com o nome de API.



The screenshot shows the 'Create a Repl' interface on the Replit website. At the top right, there is a button labeled 'Import from GitHub' with a close icon. The form is divided into two main columns. The left column has a 'Template' section with a dropdown menu currently showing 'Node.js'. Below this is a larger section for selecting a language, featuring the Node.js logo, the text 'Node.js' with a checkmark, and a 'Languages' button. The right column has a 'Title' field containing 'API', an 'Owner' dropdown menu showing 'AntonioIzo', and a 'Privacy' section with two buttons: 'Public' (selected) and 'Private'.

Criando uma API REST básica

iniciamos fazendo a importação do express para dentro de uma constante e criamos uma variável app para inicializar o express.



```
JS index.js v [icon] × +  
JS index.js > ...  
1  const express = require('express');  
2  const app = express();
```

Criando uma API REST básica

Vamos criar agora uma rota get que retornará um send para nosso browser.

```
4 ✓ app.get('/', (req, res) => {  
5   return res.send('Olá meu usuário.')  
6 })  
7 |  
8 app.listen(3000)
```



Criando uma API REST básica

Antes de testar nossa aplicação, vamos instalar via shell o express.

NPM install express

Olá meu usuário.

Criando uma API REST básica

- Uma API retorna um JSON estruturado para que a aplicação cliente pegue esses dados e faça o uso.
- Vamos então alterar o retorno da nossa rota para que seja retornado um JSON.
- Lembrando que um JSON pode retornar ou um array ou um objeto.



Criando uma API REST básica

```
4  app.get('/', (req, res) => {  
5    return res.json({ message: 'Olá meu usuário.' });  
6  })
```

Vamos testar novamente nossa aplicação.

Criando uma API REST básica

← → ↻ api.antonioizo.repl.co

```
{"message": "Olá meu usuário."}
```

Agora recebemos um JSON.

Criando uma API REST básica

- Repare que poucos códigos são novos para você.
- A grande maioria do processo de criação de uma API você já sabe o funcionamento.
- Uma API é exatamente o que fizemos.

Até a próxima semana!

Referências

- Aluizio && Developer, 2022.
- Construindo aplicações com NodeJS. Novatec editora, 2021.



**INSTITUTO
FEDERAL**
Espírito Santo