



Testes Unitarios com Node

Aplicação para estudo de criação de api com node com express para validação do modelo de informações foi usado o yup para teste unitario foi usado o jest

Neste documento tera uma explicação rapida sobre as bibliotecas, processo de criação do projeto , como instalar e iniciar o projeto cada parte sera separada por ordem de uso em modulos

Sumario

Bibliotecas usadas

- Node
- Express
- Multer
- Nodemon
- Uuid
- Yup
- Jest
- Swagger-ui-express

Usando o projeto

- Npm scripts
- Instalando Dependencias
- Abrindo o servidor
- Usando os testes
- Usando o swagger



Testes Unitarios com Node

Bibliotecas usadas

NODE

É um framework de código aberto que usa javascript para trabalharmos no back-end com ele ,podemos criar apis trabalhar com arquivos diversos

para saber mais acesse a documentação

EXPRESS

É uma biblioteca usada para a criação de api, facilita todo o processo de se criar uma api. o express cria um ambiente de teste local para a nossa api para podermos criar nossas rotas e testar o funcionamento dos nossos endpoints também podemos criar midwares a partir deles

para saber mais acesse a documentação

Multer

É um middleware que trabalha com o express para poder pegar arquivos vindo do front
middleware é uma operação que fica entre processos por exemplo a comunicação entre o front-end e o back-end

para saber mais acesse a documentação



Testes Unitarios com Node

NODEMON

É um serviço para reiniciar o nosso servidor caso o nosso código tenha alguma atualização de escrita ele é usado junto com o npm script que será explicado mais para frente para saber mais acesse a documentação

UUID

É usado para criação de id de forma aleatória podendo ser configurado quantos dígitos eu quero para o id e se ele poderá ter apenas números ou textos ou ambos, o motivo para o uso desta biblioteca é para evitar futuros problemas na hora de fazer alguma operação no banco de dados também trazendo maior confiança para saber mais acesse a documentação

YUP

Serve para facilitar a validação de valores vindo de quem consome a api, retornando mensagens de erro para que assim que esta consumindo a api no front possa retornar o erro para o usuário tornando a ui mais clara, pode ser usado para validar outras informações que não necessariamente venha de uma api, pode ser usado no front-end para validação de forms

para saber mais acesse a documentação



Testes Unitarios com Node

Jest

É uma biblioteca de teste unitarios que serve para fazer teste em pedaços do nosso código simulando como fosse o usuário que estive usando aquela parte do código/interface

para saber mais acesse a documentação

Swagger-ui-express

O Swagger é uma biblioteca para a documentação de forma visual e funcional da API e dos endpoints, tornando muito mais fácil para quem for consumir ou dar manutenção entender o que cada endpoint faz e qual é o foco da API.

para saber mais acesse a documentação



Testes Unitarios com Node

Usando o Projeto

npm scripts

É uma forma de usar alguns codigos do terminal em sequencia , podendo usar mais que um codigo para fazer mais de uma operação ao mesmo tempo , tornando mais rapido alguns processos , por exemplo o codigo abaixo ,os npm scripts ficam dentro do "scripts" ,quando é usado no console o comando npm start o terminal le os comandos dentro da tah "start" que executa o comando para executar o nodemon e define o arquivo raiz do projeto que no caso é o index.js

```
"scripts": {  
  "start": "nodemon src/index.js",  
  "test": "clear && jest"  
},
```

para saber mais acesse a documentação

Instalando dependencias

use o comando a baixo para intalar as dependencias do projeto

```
npm install
```

depois disso o terminal ira baixar todas as dependencias do projeto e ele estara pronto para uso



Testes Unitarios com Node

Abrindo o Servidor

tenha certeza de ter instalado as dependencias do projeto
abra o terminal e digite

```
npm start
```

apois isso o servidor estara pronto para teste

usando os testes

para usar os testes voce pode abrir outro terminal ou cancelar o servidor no terminal em que voce estiver,
para usar os testes use no terminal o comando abaixo

```
npm test
```

depois de algum tempo sera mostrado no terminal se os testes passaram ou não

Usando o swagger

para usar o swagger start o servidor depois no seu navegador coloque o link a seguir

```
http://localhost:8080/api-docs/#/User/post_cadastrarUser
```

depois disso é so se divertir S2