Integrando Jasmine ao NodeJS

Engenharia de Software II

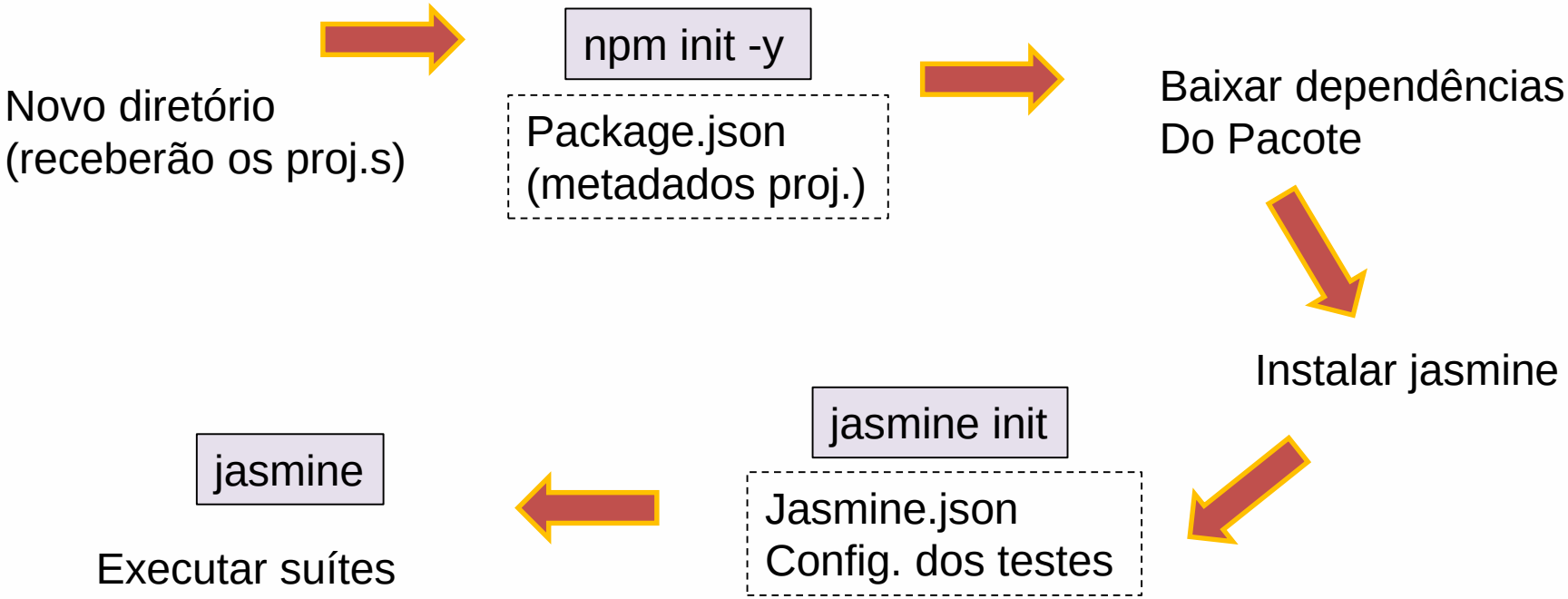
INSTITUTO FEDERAL
Mato Grosso do Sul

NodeJS

- é uma plataforma de desenvolvimento server-side em JavaScript
- é construído utilizando a Chrome V8 JavaScript Engine do Google, que permite executar código JavaScript no servidor com performance muito boa
- NodeJS é altamente escalável e orientado a eventos

INSTITUTO FEDERAL
Mato Grosso do Sul

Diagrama Geral das Configurações



INSTITUTO FEDERAL
Mato Grosso do Sul

Por que Jasmine com NodeJS?

- Permite criação de aplicação totalmente em JavaScript
- O Jasmine se integra perfeitamente com o NodeJS
- A integração de ambos permite a automatização de processos, como execução automática dos testes durante o desenvolvimento, processo de build e deploy.

INSTITUTO FEDERAL
Mato Grosso do Sul

NPM (Node Package Manager)

- é o gerenciador de pacotes do Node
- NPM também permite a execução de scripts personalizados, como por exemplo a execução dos testes
- os pacotes adicionais são armazenados no diretório node_modules na raiz do projeto

INSTITUTO FEDERAL
Mato Grosso do Sul

Arquivo package.json

- é um arquivo para armanezamento dos metados do projeto, todas as informações (nome, versão, descrição)
- fica localizado na pasta raiz do projeto
- pode ser criado com o comando "npm init"

INSTITUTO FEDERAL
Mato Grosso do Sul

Criando o package.json

- Sequência de criação do projeto no NodeJS
- Com o uso do Assistente
  - *npm init*
  - ou
- Sem o uso do Assistente (arquivo padrão)
  - *npm init -y*

INSTITUTO FEDERAL
Mato Grosso do Sul

Criando o package.json

no terminal

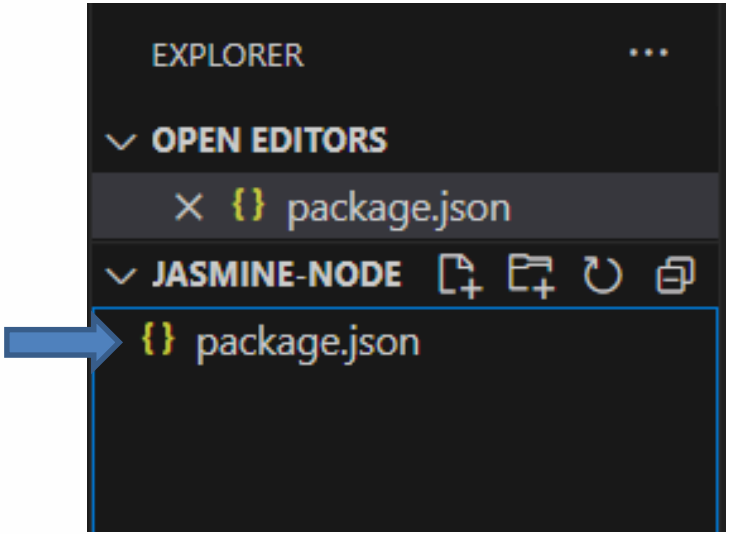
- crie o diretório do projeto
- *mkdir jasmine-node* (linux) ou *md jasmine-node* (windows)

- acesse o diretório
- *cd jasmine-node*

- crie o arquivo package.json
- *npm init -y*

- exiba o conteúdo do diretório
- *ls* (linux) ou *dir* (windows)

- abra o arquivo e visualize seu conteúdo



INSTITUTO FEDERAL
Mato Grosso do Sul

Criando o package.json

```
PS C:\Users\marci\Desktop\jasmine\jasmine-standalone-4.5.0\jasmine-node> npm init -y
npm WARN config global `--global`, `--local` are deprecated. Use `--location=global` instead.
Wrote to C:\Users\marci\Desktop\jasmine\jasmine-standalone-4.5.0\jasmine-node\package.json:

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

INSTITUTO FEDERAL
Mato Grosso do Sul

Instalando o Jasmine para NodeJS

- *npm install --save-dev jasmine*

- o **"--save-dev"** possibilita adicionar a informação do pacote no arquivo package.json criado anteriormente

- Verifique então se o arquivo package.json contém a entrada para o pacote do Jasmine em "devDependences"

- instale o jasmine
- *npm install -g jasmine*

- verifique a versão do jasmine para validar a instalação
- *jasmine -v*

INSTITUTO FEDERAL
Mato Grosso do Sul

Inicializando Jasmine no Projeto

- o comando *jasmine init* é responsável por criar os diretórios e arquivos base para a utilização do Jasmine em um projeto NodeJS

- Após sua execução o diretório 'spec' será criado, assim como o arquivo de configuração 'jasmine.json'

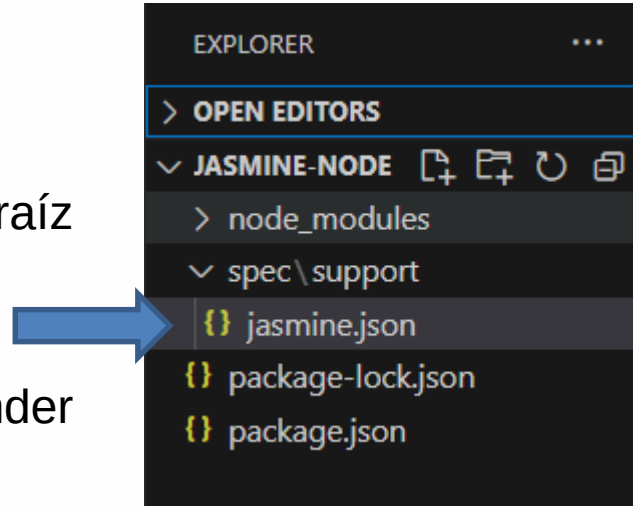
- o arquivo *jasmine.json* contém informações de execução dos testes, como localização dos arquivos de testes

INSTITUTO FEDERAL
Mato Grosso do Sul

Inicializando Jasmine no Projeto

- Inicializar o Jasmine
- *jasmine init*

- certifique que o diretório spec foi criado na raíz do projeto



- visualize o arquivo jasmine.json para entender as configurações de execução dos testes

- os testes ficarão dentro do diretório "spec"

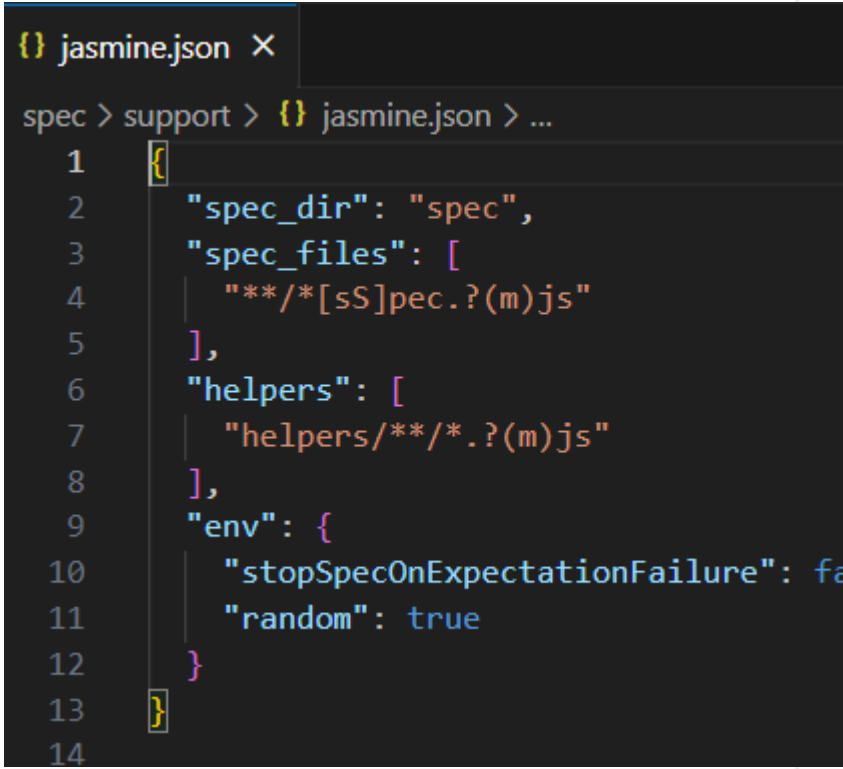
INSTITUTO FEDERAL
Mato Grosso do Sul

Inicializando Jasmine no Projeto

- a expressão regular significa que o jasmine irá procurar por aquivos dentro do spec com o padrão'*Spec.js' ou '*spec.js'.

- o diretório Helper poderá ser usado com arquivos auxiliares da aplicação.

- Por padrão, o Jasmine irá executar os testes mesmo que falhem e sem ser na ordem aleatória.



INSTITUTO FEDERAL
Mato Grosso do Sul

Criando o primeiro teste

- Criar o diretório exemplo dentro da pasta spec
  - criar o arquivo *exemploSpec.js*

- criar a suíte:
- *describe("Suíte de teste de exemplo com NodeJS", function(){*
- *it("deve demonstrar o uso do Jasmine com o NodeJS", function(){*
- *expect(true).toEqual(true);*
- *});*
- *});*

- Para executar, apenas digitar *jasmine*

INSTITUTO FEDERAL
Mato Grosso do Sul

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL

 1 spec, 0 failures
 Finished in 0.008 seconds
 Randomized with seed 39021 (jasmine --random=true --seed=39021)
PS C:\Users\marci\Desktop\jasmine\jasmine-node>
```

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL

1) Suíte de teste de exemplo com NodeJS deve demonstrar o uso do
  Message:
    Expected true to equal false.
  Stack:
        at <Jasmine>
        at UserContext.<anonymous> (C:\Users\marci\Desktop\jasmi
16)
        at <Jasmine>

1 spec, 1 failure
Finished in 0.007 seconds
Randomized with seed 87933 (jasmine --random=true --seed=87933)
PS C:\Users\marci\Desktop\jasmine\jasmine-node>
```

INSTITUTO FEDERAL
Mato Grosso do Sul

Executando teste no NodeJS

- o node permite a configuração de scripts no arquivo package.json
- os scripts ficam no bloco **scripts**
- eles possuem um nome e comando a ser executado
- o comando é executado utilizando o comando "*npm run nome_scrip*"

INSTITUTO FEDERAL
Mato Grosso do Sul

Executando teste no NodeJS

- Configurando o arquivo package.json
  - Edite o arquivo e substitua o comando encontrado em 'scripts/test' para 'jasmine'

"echo \"Error: no test strified\" && exit 1"

Trocar Para

“jasmine”

- No terminal, na raiz do projeto execute os testes
  - npm run test

INSTITUTO FEDERAL
Mato Grosso do Sul

Executando teste no NodeJS

```
PS C:\Users\marci\Desktop\jasmine\jasmine-node> npm run test
npm WARN config global `--global`, `--local` are deprecated. Use `--location=global` instead.

> jasmine-node@1.0.0 test
> jasmine

Randomized with seed 79741
Started
.


1 spec, 0 failures
Finished in 0.006 seconds
Randomized with seed 79741 (jasmine --random=true --seed=79741)
PS C:\Users\marci\Desktop\jasmine\jasmine-node>
```

INSTITUTO FEDERAL
Mato Grosso do Sul

Adicionando testes de exemplo do Jasmine no projeto

- O Jasmine contém uma suíte de testes que pode ser adicionada ao projeto
- É um modo simples de testar se o projeto está configurado corretamente
- Para baixar os testes de exemplo basta executar o comando 'jasmine examples'
- Serão criados os diretórios 'lib' na raíz do projeto e 'jasmine_examples' em spec
- Esses diretórios contém alguns testes implementados com alguns recursos do Jasmine

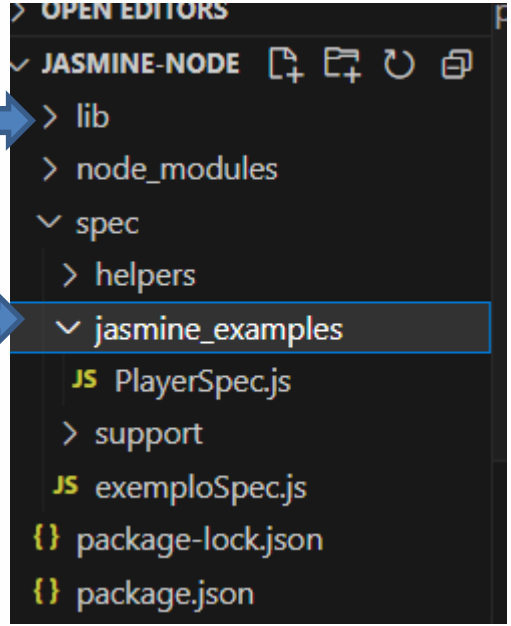
INSTITUTO FEDERAL
Mato Grosso do Sul

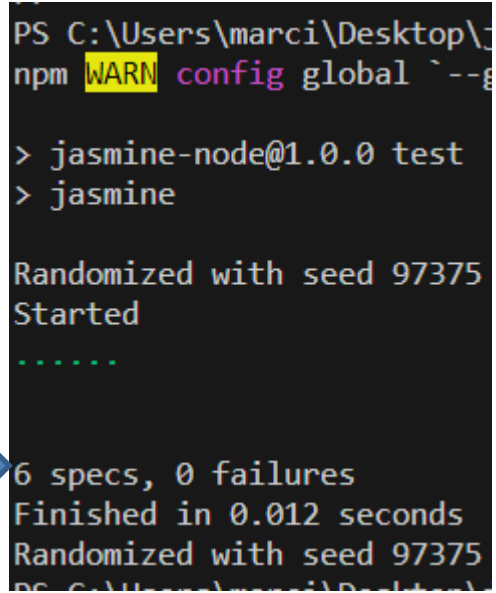
Adicionando testes de exemplo do Jasmine no projeto

- No terminal
- Baixar os exemplos
- *jasmine examples*

- visualizar os exemplos

- testá-los
- *npm run test*





INSTITUTO FEDERAL
Mato Grosso do Sul

Adicionando testes de exemplo do Jasmine no projeto

- O ideal era a pasta lib ser renomeada para src (source)

- *O arquivo de testes de exemplo foi criado no diretório:*
  - *spec/jasmine_examples/PlayerSpec.js*

- *O arquivo do Song não foi criado por que não tem implementação*