

# Rotina de Cadastro

Projeto criado para testar rotina de cadastro, para facilitar a execução, foi criado o arquivo “variaveis.json” com todas as informações usadas para a execução dos testes como: Nome, Email, URL, etc...(como mostrado na imagem 1.1) e configurado para que seja como padrão assim apenas sendo necessário executar o comando “npx cypress open” para executar os testes utilizando o arquivo de configuração com as variáveis necessárias (como mostrado na imagem 1.2).

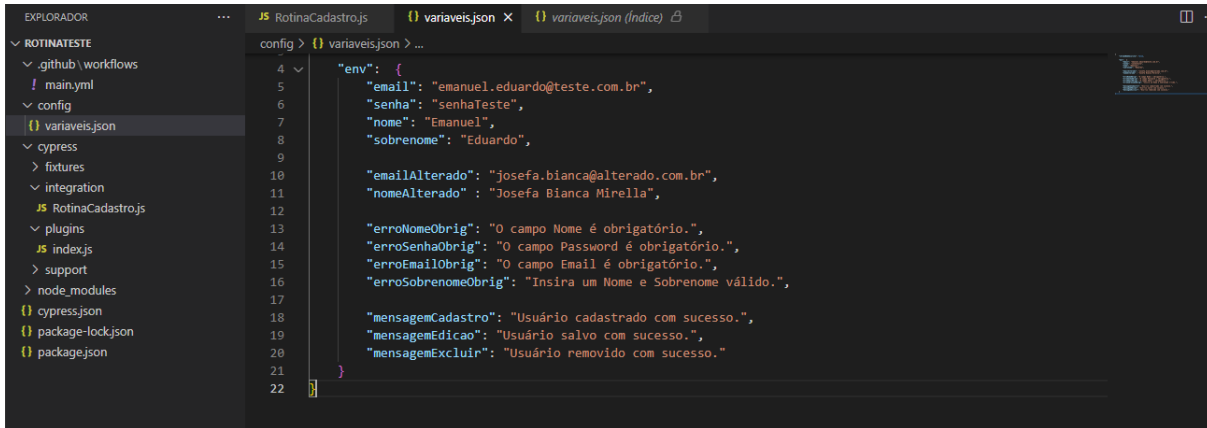


Imagem 1.1

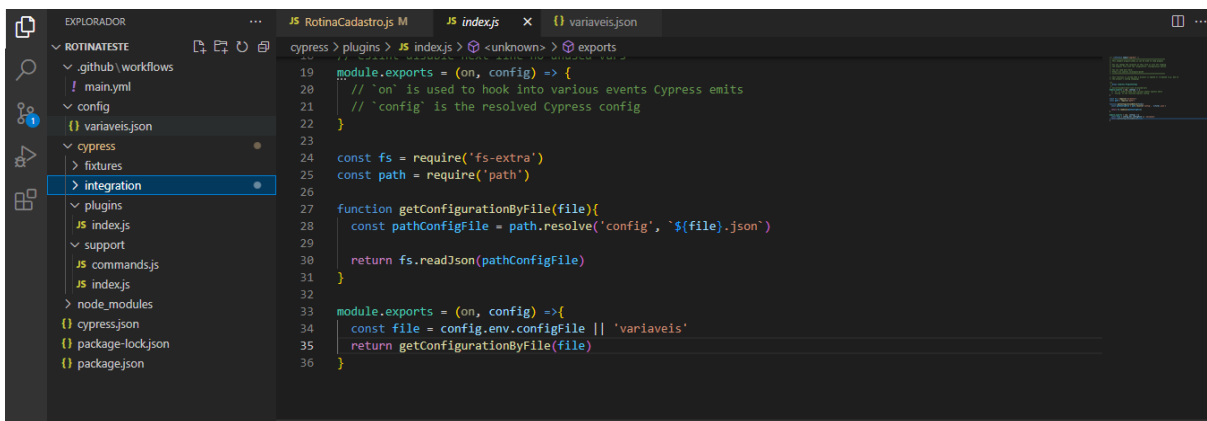


Imagem 1.2

Feito um tratamento para que caso receba qualquer retorno de erro da aplicação que não ocasiona erro de testes do cypress, não interrompa o mesmo onde apenas será interrompido caso aconteça algum cenário fora do programado em seu script (como mostrado na imagem 1.3).

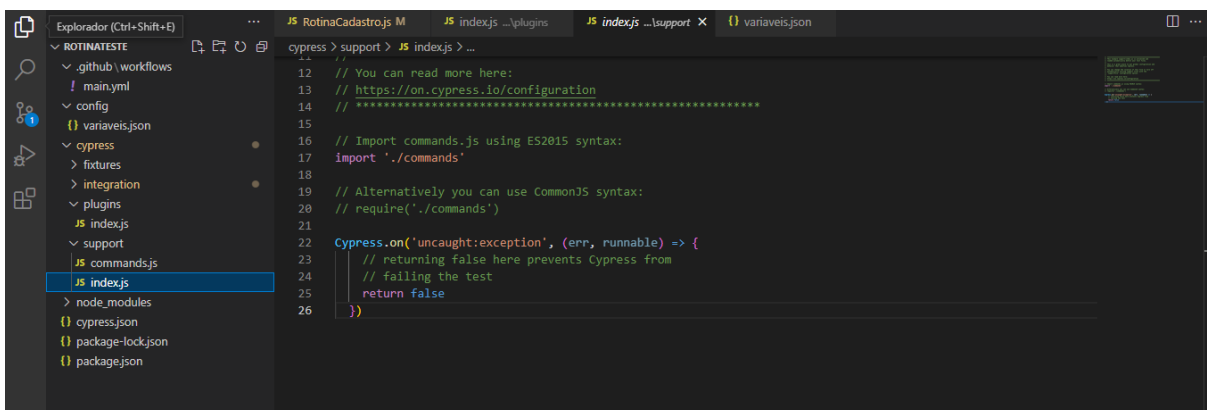


Imagem 1.3

O Primeiro Contexto de Testes foi com o intuito de verificar as mensagens de erro e testar as regras de cadastros.

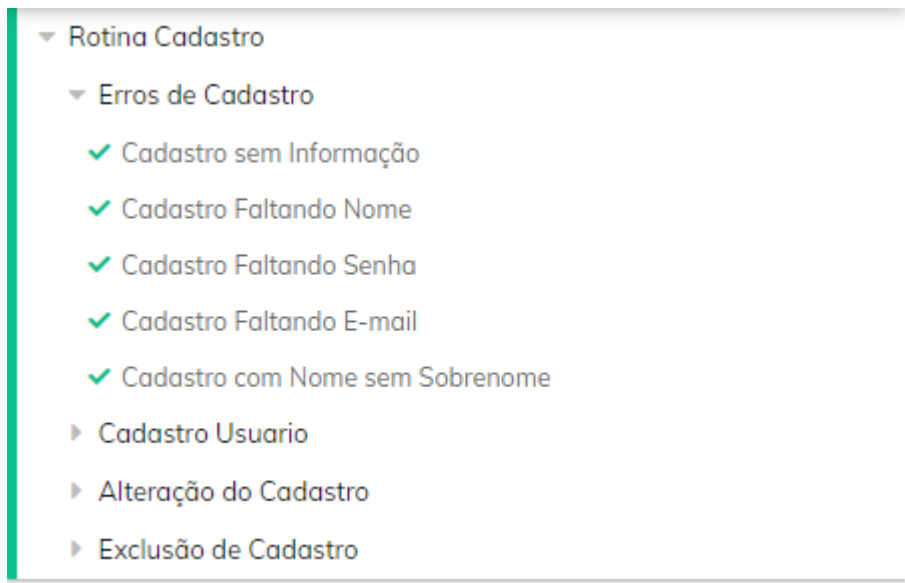


Imagem 2.1

Como pode se verificar, para se realizar um cadastro correto é necessário Nome, Senha, E-mail e o Nome deve conter Sobrenome (Onde utiliza de regra ter uma string, uma espaço em branco e pelo menos mais alguma string exemplo: xxxxx xxxx).

A imagem mostra um formulário de cadastro com o título "Formulário". Há três campos de entrada: "Nome", "E-mail" e "Password". Cada campo tem um ícone de lupa à esquerda e um ícone de seta à direita. Abaixo de cada campo, há uma mensagem de erro em vermelho: "O campo Nome é obrigatório.", "O campo Email é obrigatório." e "O campo Password é obrigatório.". No final do formulário, há um botão azul com o texto "Cadastrar".

Imagem 2.2

# Formulário

Nome

Insira um Nome e Sobrenome válido.

E-mail

Password

**Cadastrar**

Imagem 2.3

Como pode ser visto na Imagem 2.1, os testes foram feitos de maneiras unitárias, verificando se alguma interação pode-se alterar o resultado (Exemplo se ao adicionar Email, a mensagem de erro sobre a Senha ser Obrigatória não ser visível)

O Segundo Contexto de Teste foi para ser realizado o cadastro e verificar se o mesmo foi salvo corretamente

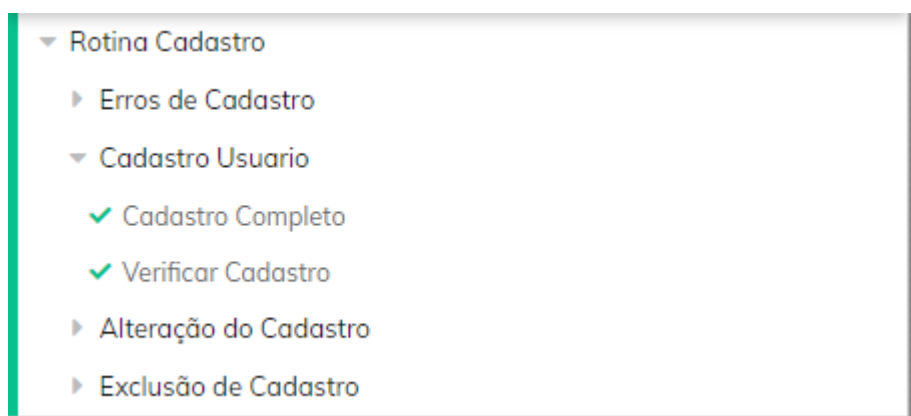


Imagem 3.1

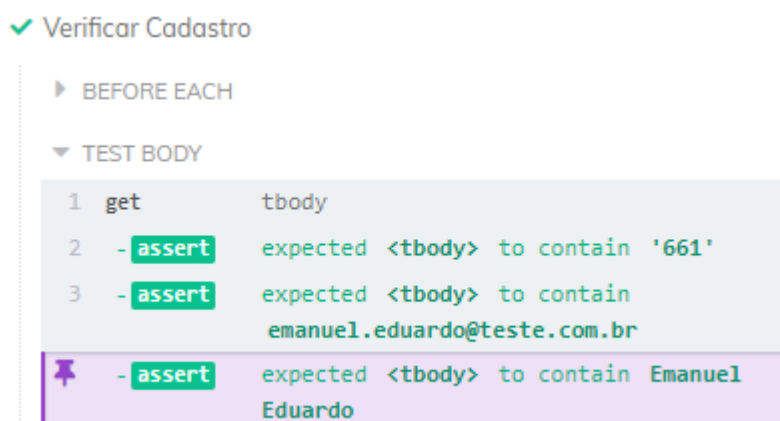


Imagem 3.2

Para a verificação do cadastro foi analisado o elemento “tbody” onde contém os cadastros realizados e analisado se continha as informações cadastradas (como mostrado na imagem 3.2)

O Terceiro contexto foi para a edição do cadastro feito anteriormente, para realizar essa alteração, durante a etapa de cadastro foi armazenada a informação de “ID” do cadastro onde através dela se acessa o cadastro realizado (como mostrado na Imagem 4.1) e assim entrando na tela de edição e alterando os dados cadastrado inicialmente (como mostrado na Imagem 4.2)

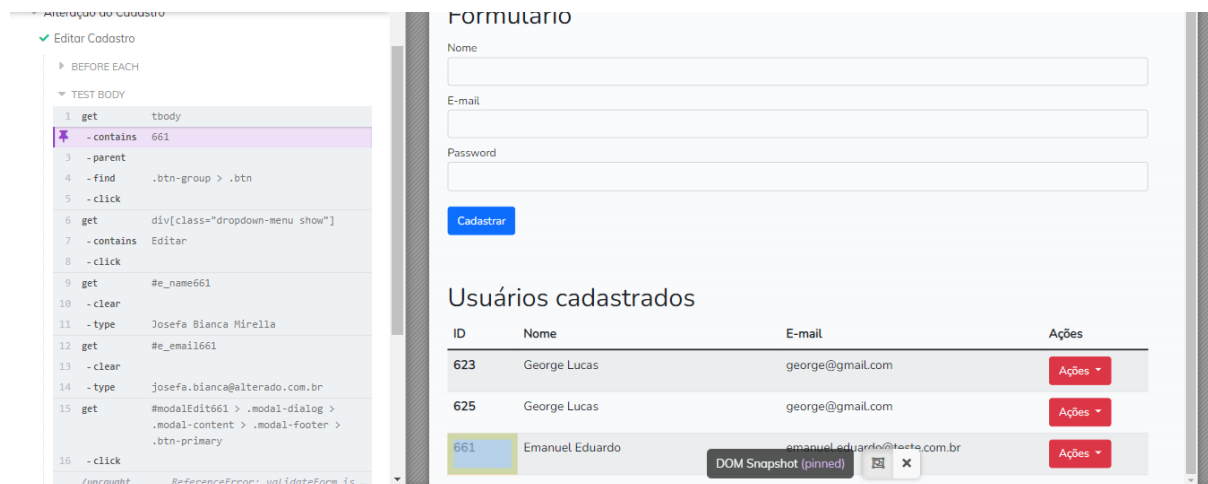


Imagem 4.1

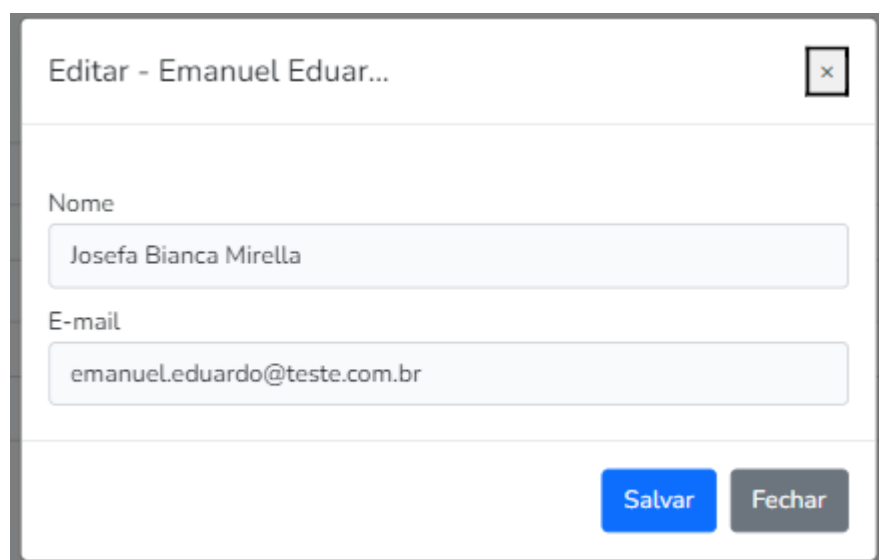


Imagem 4.2

Para a checagem se a edição foi realizada foi usada a mesma lógica do cadastro, pesquisando no elemento “tbody” as informações alteradas (como mostrado na Imagem 4.3)

#### ✓ Verificar Edição Cadastro

▶ BEFORE EACH

▼ TEST BODY

	get	tbody
2	- assert	expected <tbody> to contain '661'
3	- assert	expected <tbody> to contain Josefa Bianca Mirella
4	- assert	expected <tbody> to contain josefa.bianca@alterado.com.br

Imagem 4.3

O quarto contexto foi realizado a exclusão dos dados cadastrados, utilizando a mesma lógica da edição realizando a pesquisa através da “ID”, foi acessado o cadastro realizado durante os testes e executado sua exclusão. Ao confirmar a exclusão e analisado se a mensagem de confirmação da exclusão é vista (Imagem 5.3)

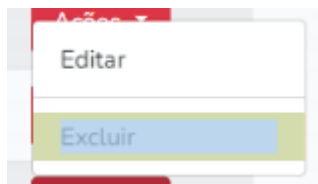


Imagem 5.1



Imagem 5.2

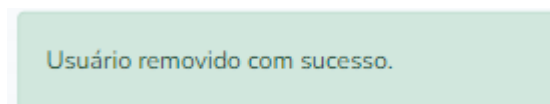


Imagem 5.3

Para a verificação da exclusão, além da mensagem verificada no passo de exclusão, é criado o teste de verificação onde se mantém a mesma lógica de verificação dos testes anteriores, se observando o elemento “tbody” se não contém os dados que haviam sido cadastrados (Imagem 5.4).

#### ✓ Verificar Se o Cadastro foi Apagado

##### ▶ BEFORE EACH

##### ▼ TEST BODY

```
1 get      tbody
2 - assert expected <tbody> not to contain '661'
3 - assert expected <tbody> not to contain
   Josefa Bianca Mirella
4 - assert expected <tbody> not to contain
   josefa.bianca@alterado.com.br
```

Imagem 5.4

Foi criado no repositório do GitHub uma Action para que sempre que for realizado um push, será rodado o cypress verificando os testes com o intuito de segurança para impedir algum incidente de commit que possa interferir na estrutura dos scripts “quebrando” o código (Imagem 6.1, Imagem 6.2 e Imagem 6.3). E caso necessário informar qual o commit e alteração será necessária correção.

12 lines (12 sloc) | 270 Bytes

RawBlame✎📄🗑

```
1 name: Cypress Tests
2 on: [push]
3 jobs:
4   cypress-run:
5     runs-on: ubuntu-latest
6     steps:
7       - name: Checkout
8         uses: actions/checkout@v2.4.2
9       - name: Cypress.io
10        uses: cypress-io/github-action@v4.2.0
11        with:
12          start: npx cypress open
```

Imagem 6.1

<> Code🕒 Issues🔗 Pull requests🔧 Actions📁 Projects🛡 Security📊 Insights⚙ Settings

Workflows

New workflow

All workflows

Cypress Tests

All workflows

Showing runs from all workflows

Filter workflow runs

3 workflow runs

EventStatusBranchActor

test: :art: Adição de Teste

Cypress Tests #9: Commit 2fbf286 pushed by MateusGogh

master

2 minutes ago

1m 20s

refactor: :recycle: Alteração código para Teste de Github ...

Cypress Tests #8: Commit 051e2e2 pushed by MateusGogh

master

1 hour ago

57s

Update main.yml

Cypress Tests #7: Commit 2801fab pushed by MateusGogh

master

1 hour ago

1m 1s

Imagem 6.2

test: :art: Adição de Teste Cypress Tests #9

Re-run all jobs⋮

Summary

Jobs

cypress-run

cypress-run

succeeded now in 1m 8s

Search logs⚙

Set up job

Checkout

Cypress.io

Post Checkout

Complete job

2s

1s

1m 4s

0s

0s

Imagem 6.3