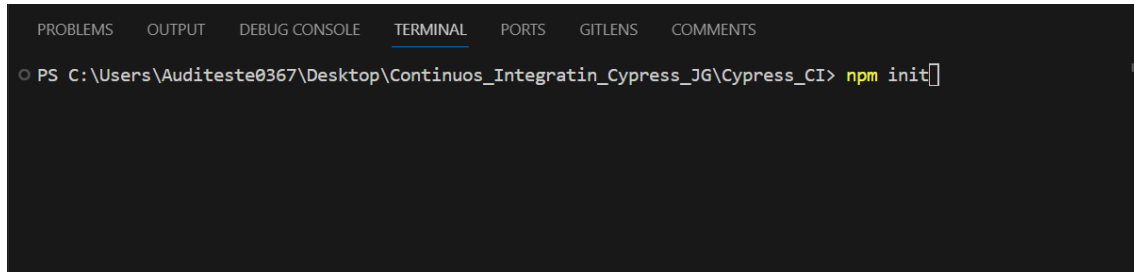


# Automação Cypress

Este documento tem a intenção de demonstrar e padronizar o desenvolvimento de testes automatizados utilizando Cypress.

## 1 – Iniciando Projeto



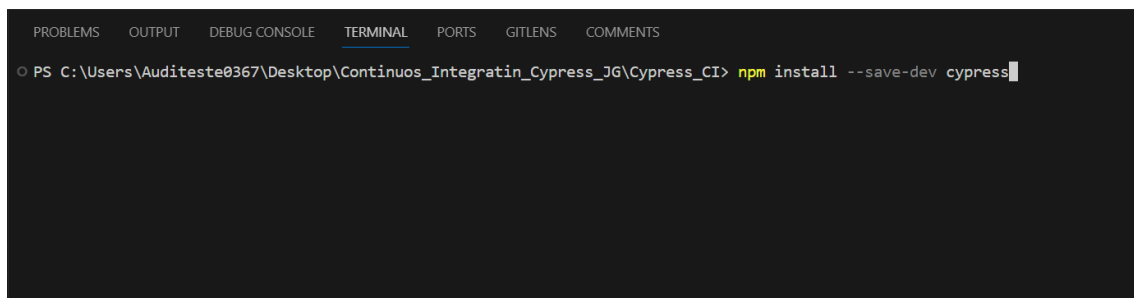
```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS GITLENS COMMENTS
PS C:\Users\Auditeste0367\Desktop\Continuos_Integratin_Cypress_JG\Cypress_CI> npm init
```

O comando **npm init** sinaliza a iniciação do nosso projeto, algumas informações serão pedidas e registradas no arquivo package.JSON



```
{
  "name": "testes_regressivos_jovens_genios",
  "version": "1.0.0",
  "description": "Testes automatizados com Workflow",
  "main": "index.js",
  "scripts": {
    "test": "cypress run"
  },
  "author": "Bruno Dantas (Auditeste).",
  "license": "ISC",
  "devDependencies": {
    "@cypress/xpath": "^2.0.3",
    "cypress": "^13.3.1",
    "cypress-iframe": "^1.0.1"
  }
}
```

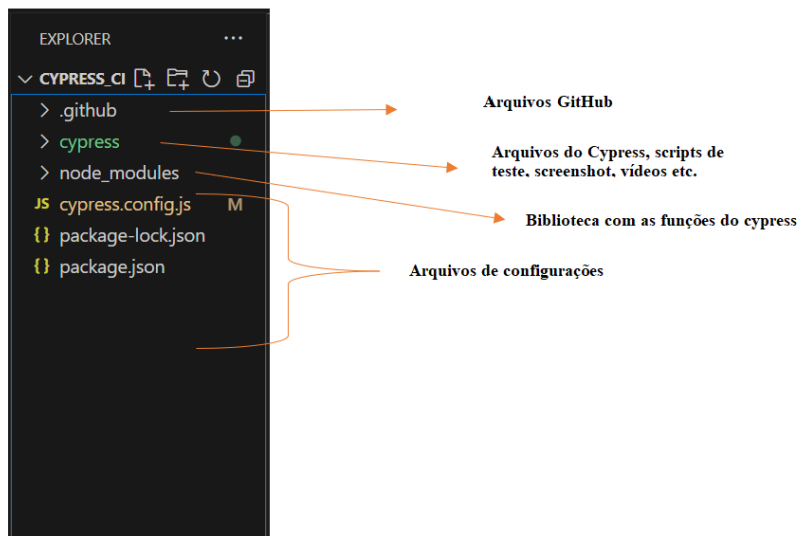
## 2 -Instalação do Cypress



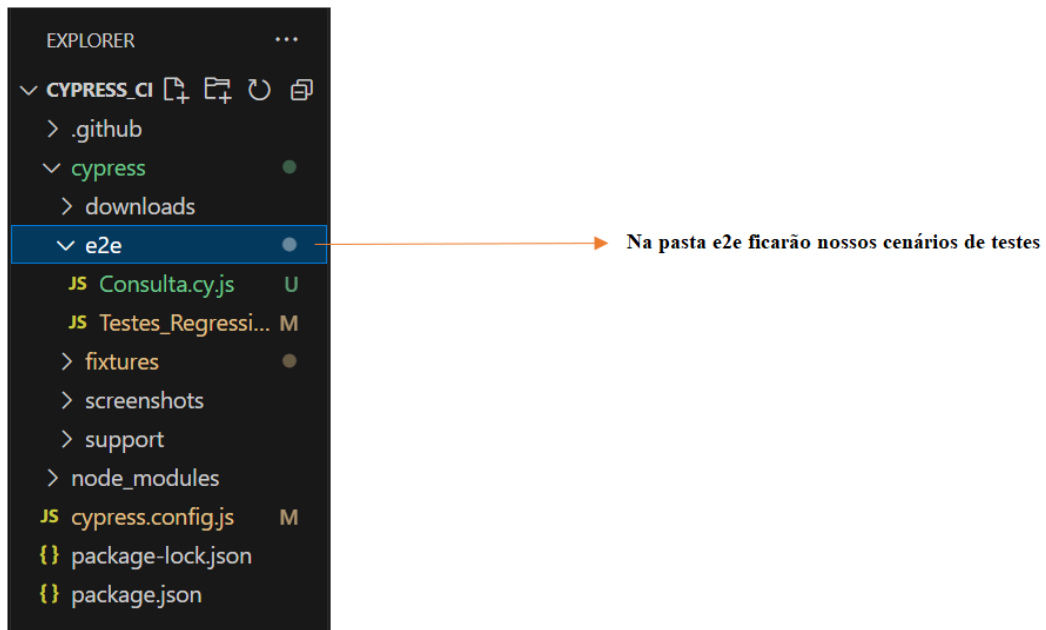
```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS GITLENS COMMENTS
PS C:\Users\Auditeste0367\Desktop\Continuos_Integratin_Cypress_JG\Cypress_CI> npm install --save-dev cypress
```

O comando **npm install --save-dev cypress** irá baixar os módulos do cypress

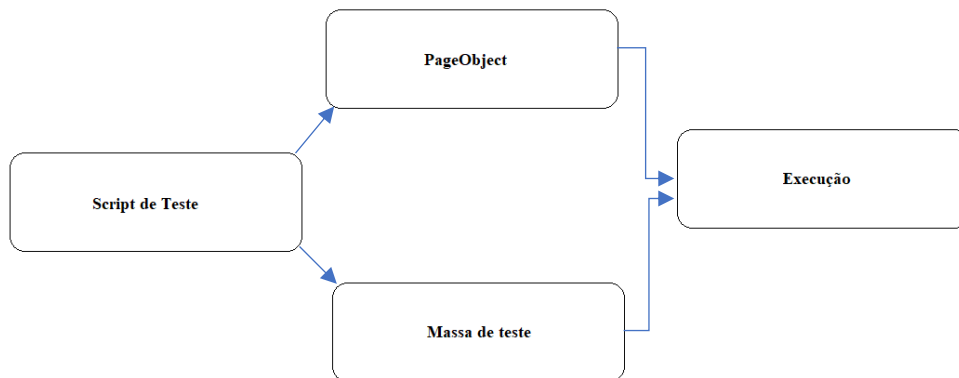
### 3 – Estrutura do Cypress



### 4 – Dentro da pasta Cypress



## 5 - A estrutura do projeto



Script de teste é onde escrevemos as ações a serem tomadas nos testes, os identificadores dos elementos são atribuídos no arquivo PageObject, os dados usados no teste ficam no arquivo massa de teste, mapeados esses arquivos a execução começa.

No script de teste não deixamos valores hardcoded

No comando `cy.visit()`, estamos passando um valor da variável que está no arquivo `massa.json`

No comando `cy.get()`, passamos uma variável que está no arquivo `pageobject.json`

```
it('CT001 - Se a escola estiver dentro do período de Olimpíadas, deverá aparecer a mensagem de aviso', () => {

  cy.visit(mt.Url)
  cy.xpath(po.ID_User).should('be.visible').click()

  cy.xpath(po.ID_User).type(mt.Usuario);
  cy.get(po.ID_password).type(mt.Senha);
  cy.get(po.ID_Botao).click();

});
```

### Massa.json

```
cypress > fixtures > {} MassaTeste.json > Senha
You, last month | 1 author (You)
1 {
2   "Url": "https://dev-educadores.jovensgenios.com/login",
3   "Usuario": "educadortesteqa@jovensgenios.com",
4   "Senha": "jovensgenios"
5 }
```

### PageObject.json

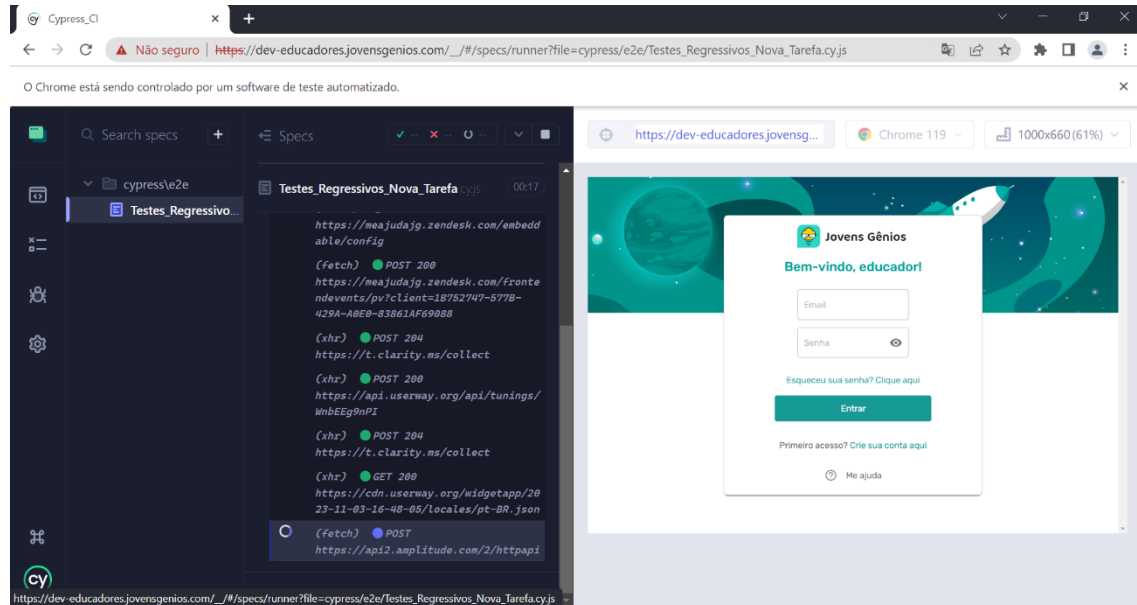
```
{
  "ID_User" : "/html/body/div[7]/div/div[1]/main/div/div[1]/div/div/div/div[3]/div/div/div[1]/div/input",
  "ID_password" : "[data-cy=password]",
  "ID_Botao" : ".v-btn__content > .row > .d-flex",
  "ID_Nova_Atividade" : "[data-cy='home-create'] > .row > .fs-18",
  "ID_Nova_Tarefa" : "[data-cy=TAREFA] > .rounded-lg",
  "ID_Botao_Criar" : "[data-cy='criar'] > .btn"
```

## 6 – A Execução

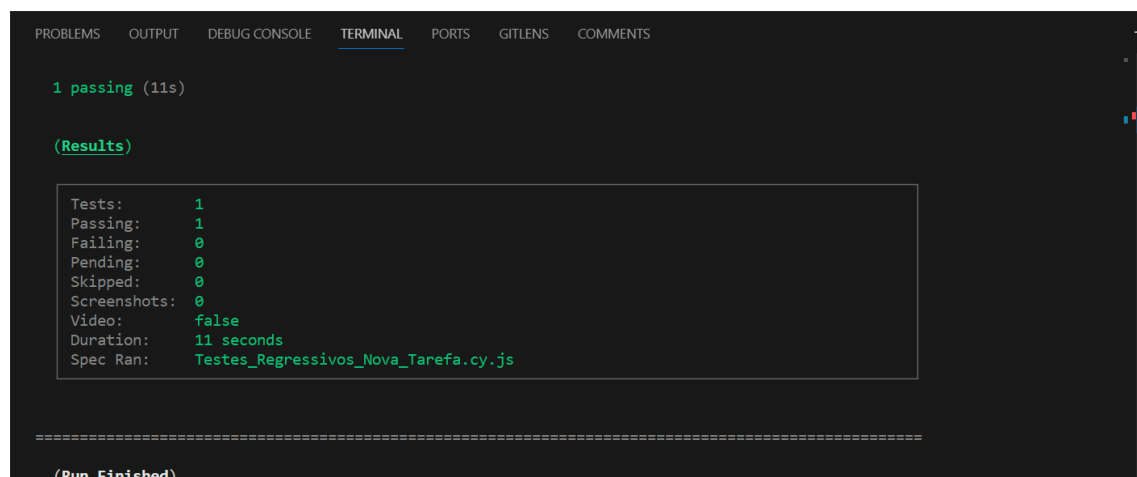
Podemos utilizar dois comandos para a execução dos testes:

**npx cypress open**

Irá abrir o IDE do Cypress.



Ou utilizar o comando **npx cypress run** que vai executar em background.



Um material vasto e de qualidade é disponibilizado pelo site do cypress:

<https://docs.cypress.io/guides/getting-started/installing-cypress>