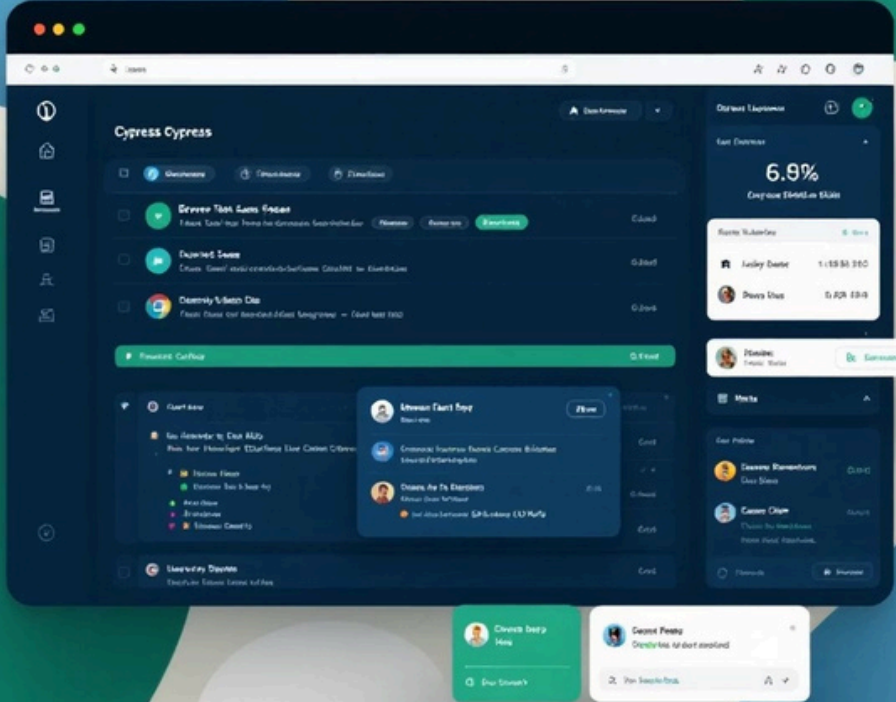


Testes Automatizados com Cypress





O que é Cypress?

Framework de Testes E2E

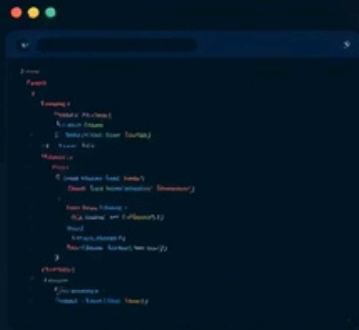
Ferramenta moderna para testes end-to-end de aplicações web. Executa diretamente no navegador, integrando-se a página.

Benefícios Principais

Configuração simples e rápida. Visualização em tempo real dos testes. Depuração facilitada com snapshots automáticos.

Comunidade Ativa

Documentação rica e acessível. Plugins diversos para expandir funcionalidades. Suporte constante e atualizações frequentes.



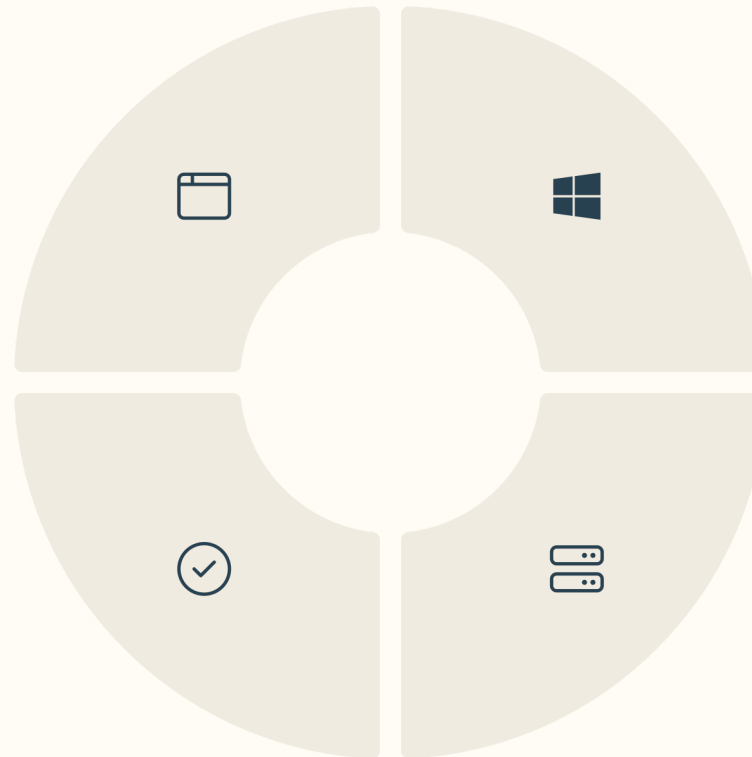
Tipos de Testes com Cypress

Testes E2E

Simulam comportamento real do usuário.
Testam o sistema como um todo,
validando fluxos completos.

Cobertura de Testes

Métricas claras sobre áreas testadas.
Identifica partes críticas da aplicação sem
verificação.



Testes de Interface

Verificam a renderização correta dos
elementos. Garantem funcionamento
adequado dos componentes visuais.

Testes de API

Executados via `cy.request`. Validam
respostas e comportamentos dos
endpoints da aplicação.



Tecnologias do Projeto



Backend

Node.js com Express para API REST. SQLite como banco de dados leve e prático.



Frontend

JavaScript puro com HTML5 e CSS3. Interface simples e funcional para gerenciamento de tarefas.



Ferramentas de Teste

Cypress para automação.

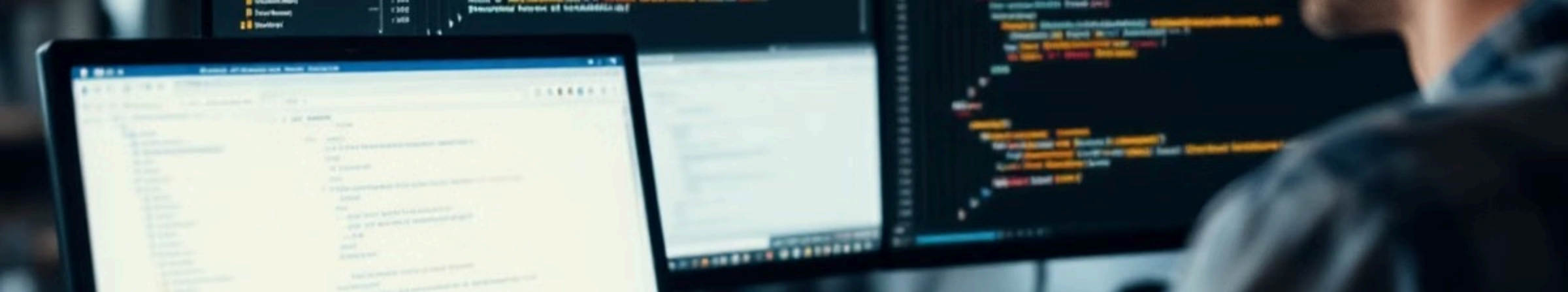


Especificação

Gherkin para escrita de cenários em linguagem natural. Facilita entendimento por não-programadores.

Estrutura do Projeto

- **apps/**
 - **markL/**
 - **api/** - Backend REST com Node.js
 - **web/** - Interface do usuário
- **cypress/**
 - **e2e/** - Testes end-to-end
 - **fixtures/** - Dados de teste
 - **support/** - Comandos personalizados
- **package.json** - Dependências do projeto
- **cypress.config.js** - Configuração do Cypress



Como Executar o Projeto

Configuração Inicial

Clone o repositório do GitHub.
Execute `npm install` para instalar todas as dependências necessárias.

Iniciar Serviços

Execute `npm run api` para iniciar o backend. Em outro terminal, inicie a interface com `npm run web`.

Executar Testes

Abra o Cypress com `npx cypress open`. Selecione os testes E2E e o navegador desejado.

Visualizar Relatórios

Gere relatórios com `npm run allure:generate`. Visualize com `npm run allure:open` após a execução.

Casos de Teste Implementados



Verificação de Status

Testa se a aplicação está online e respondendo corretamente. Valida o texto "No momento você não tem nenhum compromisso".



Cadastro de Tarefas

Testa inclusão de nova tarefa. Verifica validações como campo obrigatório e duplicidade.

Conclusão de Tarefas

Valida se é possível marcar tarefa como concluída. Verifica alteração visual e persistência do estado.

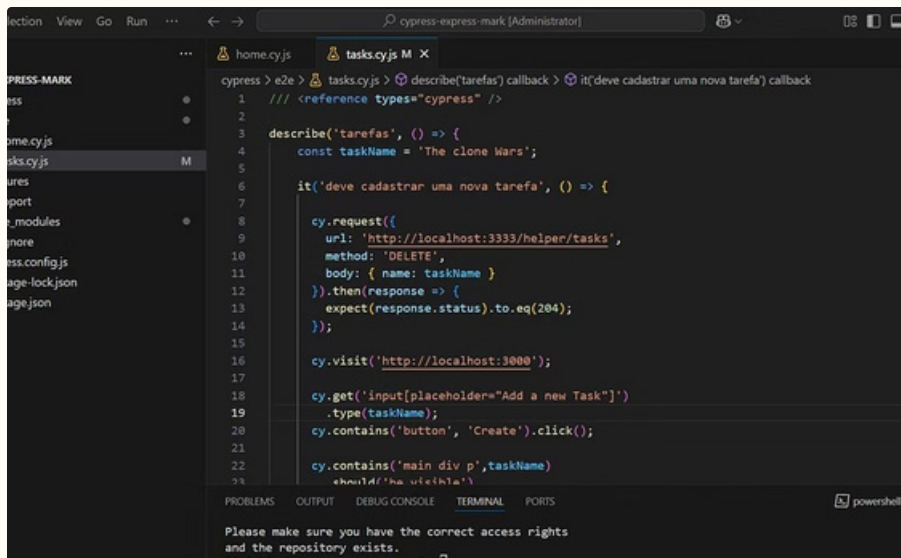


Exclusão de Tarefas

Testa remoção de tarefa da lista. Confirma que a tarefa não aparece após exclusão.

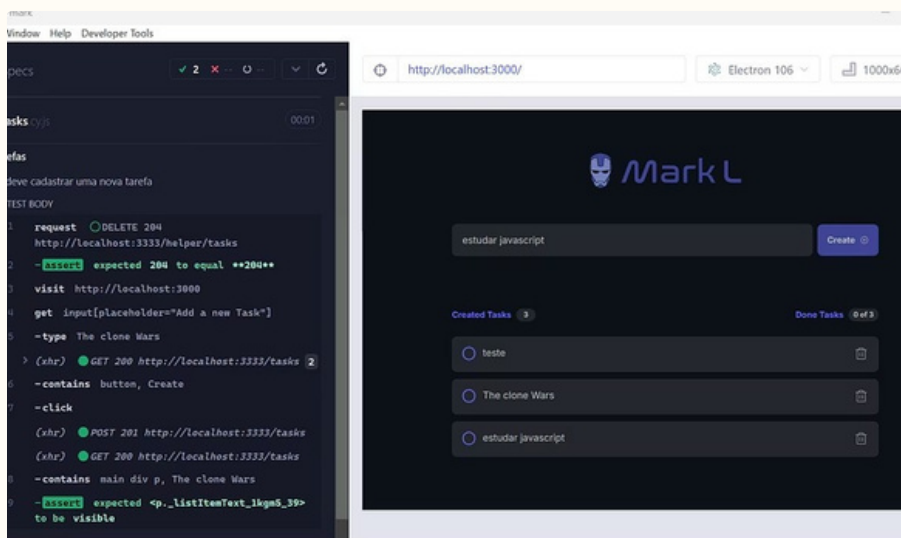


Demonstração Prática



Interface de Testes

O Cypress oferece visualização em tempo real da execução. Cada passo é documentado com screenshot automático.



Interação com Elementos

Testes localizam elementos da interface automaticamente. Clicks, entradas de texto e verificações são registrados.



Relatórios de Testes

100%

Cobertura

Porcentagem de funcionalidades cobertas pelos testes automatizados.

4

Suítes

Conjuntos de testes organizados por funcionalidade.

12

Casos

Total de casos de teste implementados no projeto.

1.5s

Tempo Médio

Duração média da execução de cada teste individual.

Conclusão



Qualidade Garantida

Testes automatizados previnem regressões e garantem funcionalidades.



Eficiência no Desenvolvimento

Deteção rápida de problemas e feedback imediato para desenvolvedores.



Base para Aprendizado

Projeto prático demonstra aplicação real dos conceitos teóricos.