

Plano de Teste para Google.com.br

Este plano de teste visa garantir a qualidade do site Google.com.br, focando em sua funcionalidade principal: pesquisa de informações.

1. Plano de Teste Funcional (PDF)

1.1 Escopo do Teste:

* **Funcionalidades:** Busca de informações, navegação no site, acessibilidade, segurança, idioma.

* **Plataformas:** Chrome, Firefox, Safari, Edge.

* **Dispositivos:** Desktop, Mobile.

* **Linguagens:** Português (Brasil).

1.2 Ambientes de Teste:

* **Ambiente de Desenvolvimento:** Utilizado para testes iniciais e desenvolvimento.

* **Ambiente de Homologação:** Utilizado para testes de integração e validação final.

* **Ambiente de Produção:** Utilizado para testes de regressão após implementação de novas funcionalidades.

1.3 Casos de Teste:

| Caso de Teste | Descrição | Critérios de Aceitação |

|---|---|---|

| CT-01 | Realizar uma busca simples | A pesquisa deve retornar resultados relevantes e relacionados à palavra-chave. |

| CT-02 | Realizar uma busca avançada com filtros | A pesquisa deve retornar resultados filtrados de acordo com os parâmetros selecionados. |

| CT-03 | Acessar a página de "Configurações" | A página de configurações deve ser acessível e permitir modificações nas preferências do usuário. |

| CT-04 | Acessar a página de "Ajuda" | A página de ajuda deve ser acessível e fornecer informações relevantes para o usuário. |

| CT-05 | Acessar o site em diferentes idiomas | O site deve ser acessível em diferentes idiomas, com conteúdo traduzido e funcionalidade adequada. |

| CT-06 | Realizar login com conta do Google | O login deve ser realizado com sucesso, permitindo acesso a funcionalidades específicas. |

| CT-07 | Verificar a acessibilidade do site | O site deve atender aos padrões de acessibilidade, com textos alternativos, contraste de cores e navegação por teclado. |

| CT-08 | Validar a segurança do site | O site deve utilizar medidas de segurança adequadas, como HTTPS e autenticação de usuários. |

| CT-09 | Validar a funcionalidade do site em diferentes dispositivos | O site deve ser funcional e responsivo em diferentes dispositivos, incluindo desktops e dispositivos móveis. |

****1.4 Critérios de Aceitação:****

- * O teste é considerado bem-sucedido se todos os critérios de aceitação forem cumpridos.
- * Os critérios de aceitação são específicos para cada caso de teste e devem ser definidos com clareza.

****1.5 Resultados Esperados:****

- * Todos os casos de teste devem ser executados com sucesso, sem erros ou falhas.
- * O site deve ser funcional, responsivo e seguro.

****1.6 Documentação:****

- * Todos os casos de teste devem ser documentados com detalhes, incluindo descrição, passos, resultados esperados e critérios de aceitação.
- * Os resultados dos testes devem ser registrados em um relatório de testes.

2. Código de Automação em Cypress (pasta "codigo_automacao")

```
````javascript
// cypress/integration/google_search.spec.js
describe('Google Search', () => {
 it('should perform a simple search', () => {
 cy.visit('https://www.google.com.br/');
 cy.get('input[name="q"]').type('Cypress{enter}');
 cy.get('h3').should('contain', 'Cypress');
 });

 it('should perform a search with filters', () => {
 cy.visit('https://www.google.com.br/');
 cy.get('input[name="q"]').type('Cypress');
 cy.get('.QS5gu').click(); // Click on "Tools" button
 cy.get('.hdtb-mn-hd').should('be.visible'); // Verify "Tools" menu is visible
 });

 it('should access the "Settings" page', () => {
 cy.visit('https://www.google.com.br/');
 cy.get('.gb_d').click(); // Click on "Google apps" button
 cy.get('.gb_1').click(); // Click on "Settings" link
 cy.url().should('include', '/search?q='); // Verify the URL includes "search?q="
 });

 // Add more tests for other functionalities...
});
````
```

3. Instruções para Configuração e Execução (README.md)

3.1 Configuração:

1. **Instalar Node.js:** Baixe e instale o Node.js em seu sistema.
2. **Instalar Cypress:** Abra o terminal e execute o comando ``npm install cypress --save-dev``.
3. **Abrir o Cypress:** Execute ``npx cypress open``.

3.2 Executar Testes:

1. Abra o Cypress através do terminal.
2. Selecione o arquivo de teste ``google_search.spec.js`` na interface do Cypress.
3. Os testes serão executados automaticamente e os resultados serão exibidos na interface do Cypress.

3.3 Interpretar Resultados:

- * **Testes Passados:** Os testes que foram executados com sucesso serão marcados como "Pass".
- * **Testes Falhados:** Os testes que falharam serão marcados como "Fail".
- * **Erros:** Qualquer erro ocorrido durante a execução dos testes será exibido em detalhes na interface do Cypress.

3.4 Dicas:

- * Utilize a funcionalidade de "regravar" do Cypress para gerar automaticamente o código de teste para uma ação específica.
- * Utilize os comandos ``cy.debug()`` e ``cy.pause()`` para depurar o código de teste.
- * Utilize os recursos de "assert" do Cypress para verificar se o site está funcionando como esperado.

3.5 Considerações:

- * Este plano de teste é um exemplo e pode ser adaptado para atender às necessidades específicas do projeto.
- * É importante manter o plano de teste atualizado com as novas funcionalidades e mudanças no site.

3.6 Próximos Passos:

- * Adicionar mais casos de teste para cobrir todas as funcionalidades do site.
- * Implementar testes de desempenho e segurança.
- * Automatizar a execução dos testes.