

1. Testes Manuais

1.1 Descrição do Sistema

O candidato deve criar e descrever um sistema fictício ou realizar uma análise das soluções disponíveis na seguinte URL: [Adapt Liviú](#)

1.2 Plano de Teste

Cenário 1: Solicitar Atendimento

Caso de uso:

O usuário deve conseguir entrar em contato com a equipe de atendimento da Adapt Liviú por meio do formulário de contato disponível no site.

Critérios de Aceitação:

- O sistema deve exibir um formulário de contato na página "Fale com a nossa equipe".
- O formulário deve conter os seguintes campos obrigatórios:
 - Nome
 - E-mail
 - Empresa
 - Produto
 - Telefone
 - Mensagem
- O formulário deve ter um termo de aceite para que o usuário permita o envio de e-mails da Adapt.
- Após o envio do formulário, o sistema deve exibir uma mensagem de sucesso confirmando o envio da solicitação.
- O sistema deve enviar um e-mail automático para a equipe de atendimento com as informações preenchidas pelo usuário.

Passos Detalhados:

1. Acessar o site [Adapt Liviú](#).
2. Clicar na opção "Fale com a nossa equipe".
3. Preencher todos os campos obrigatórios com informações válidas:
 - Nome: "João Silva"
 - E-mail: "joao.silva@email.com"
 - Empresa: "Empresa Teste Ltda"
 - Produto: Selecionar um dos produtos disponíveis
 - Telefone: "(11) 99999-9999"
- 1.
- 2.
- 3.

4. Mensagem: "Gostaria de obter mais informações sobre os serviços."
5. Marcar a opção "Aceito receber e-mails da Adapt" (se for um campo obrigatório).
6. Clicar no botão "Entrar em contato".

Resultado Esperado:

- O sistema exibe uma mensagem de sucesso, confirmando que o formulário foi enviado corretamente.

Resultado real:

- Passou: O sistema exibe uma mensagem de sucesso, confirmando que o formulário foi enviado corretamente.

Cenário 2: Acessar Outros Produtos da Adapt

Caso de uso:

O usuário deve conseguir navegar entre os produtos da Adapt, incluindo o AdaptEdtech, a partir do site principal.

Critérios de Aceitação:

- O sistema deve permitir que o usuário acesse outros produtos da Adapt a partir do site principal.
- Ao selecionar o produto AdaptEdtech, o usuário deve ser redirecionado corretamente para a página inicial desse produto.

Passos Detalhados:

1. Acessar o site Adapt Liviu.
2. Rolar a página até o final e clicar na opção "Quem Somos".
3. Clicar no link ou botão correspondente ao produto AdaptEdtech.

Resultado Esperado:

- O usuário é redirecionado corretamente para a página inicial do AdaptEdtech.
- O site carrega sem erros e exibe as informações do produto AdaptEdtech.

Resultado real:

- O usuário é redirecionado corretamente para a página inicial do AdaptEdtech.
- O site carrega sem erros e exibe as informações do produto AdaptEdtech.

Cenário 3: Acessar a Tela de Vagas no AdaptEdtech

Caso de uso:

O usuário deve conseguir acessar a página de vagas disponíveis no site do AdaptEdtech.

Critérios de Aceitação:

- O sistema deve permitir que o usuário navegue até a seção de carreiras.
- O usuário deve conseguir acessar a página de Vagas Internas.
- O sistema deve exibir corretamente a lista de vagas disponíveis.

Passos Detalhados:

1. Acessar o site AdaptEdtech.
2. No menu principal, clicar na opção "Carreiras".
3. Na página de carreiras, clicar em "Vagas Internas".
4. Clicar em "Ver todas as oportunidades"

Resultado Esperado:

- O usuário seja redirecionado corretamente para a tela de Vagas Internas.
- O sistema exibe a página contendo a lista de vagas disponíveis.
- O usuário pode visualizar e buscar oportunidades dentro da empresa.

Resultado real:

- O usuário seja redirecionado corretamente para a tela de Vagas Internas.
- O sistema exibe a página contendo a lista de vagas disponíveis.
- O usuário pode visualizar e buscar oportunidades dentro da empresa.

Cenário 4: Buscar Vagas Internas

Caso de uso:

O usuário deve conseguir pesquisar as vagas internas disponíveis na empresa.

Critérios de Aceitação:

- O sistema deve permitir que o usuário acesse a seção de vagas internas.
- O sistema deve disponibilizar um campo de busca para facilitar a pesquisa de vagas.

- O sistema deve exibir as vagas disponíveis conforme os critérios de busca aplicados.
- O usuário deve conseguir visualizar os detalhes da vaga selecionada.

Passos Detalhados:

1. Acessar o site AdaptEdtech.
2. No menu principal, clicar na opção "Carreiras".
3. Na página de carreiras, clicar em "Vagas Internas".
4. Clicar em "Ver todas as oportunidades" para visualizar a lista de vagas disponíveis.
5. Utilizar o campo de busca para pesquisar vagas específicas.
6. Selecionar uma vaga da lista para visualizar seus detalhes.

Resultado Esperado:

- O usuário deve visualizar o seu resultado de busca
- O sistema deve exibir a vaga caso o resultado pesquisado for encontrado
- O sistema deve exibir uma mensagem informativa caso o resultado não for encontrado.

Cenário 5: Candidatar-se a uma Vaga

Caso de uso:

O usuário deve conseguir se candidatar a uma vaga interna disponível na empresa.

Critérios de Aceitação:

- O sistema deve permitir que o usuário se candidate a uma vaga disponível.
- O sistema deve solicitar as informações necessárias para a candidatura.
- O sistema deve exibir uma confirmação de envio da candidatura.

Passos Detalhados:

1. Seguir os passos do **Cenário 4** para acessar a lista de vagas.
2. Selecionar uma vaga disponível e visualizar seus detalhes.
3. Clicar na opção "**Candidatar-se**".
4. Preencher os campos obrigatórios do formulário de candidatura.
5. Confirmar o envio da candidatura.

Resultado Esperado:

- O sistema exibe uma confirmação de candidatura enviada com sucesso.
- A candidatura do usuário é registrada corretamente no sistema.
- O usuário pode visualizar o status da sua candidatura futuramente.

Resultado real:

- O sistema exibe uma confirmação de candidatura enviada com sucesso.
- A candidatura do usuário é registrada corretamente no sistema.
- O usuário pode visualizar o status da sua candidatura futuramente.

1.3 Testes Exploratórios

Título do bug: Redundância de contato

Ao acessar a tela <https://adaptliviu.com.br/liviu-content/#contact-form> a ação para entrar em contato com a equipe e solicitar demonstração fica redundante enviando para o mesmo formulário.

1. Acessar <https://adaptliviu.com.br/>
2. Em “Conheça nossas soluções” clicar em “Saiba mais”



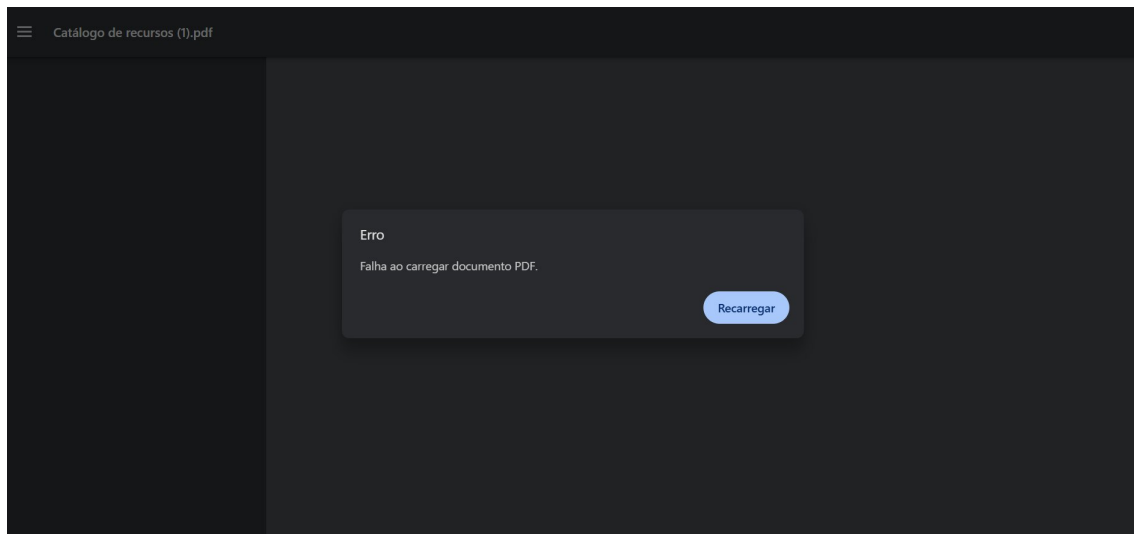
Severidade Baixo

Título do bug: Erro ao abrir arquivo em PDF

Ao baixar o arquivo em PDF da tela de catálogo de recursos exibe um erro.

1. Acessar a tela <https://clientes.livioadapt.com.br/websites/recursos/#/s/4f93f130-bcb1-4bf1-9540-f337a02bf8d3>

2. Abrir o menu catálogo de recurso
3. Clicar em “Baixar PDF”
4. Abrir na própria máquina o arquivo



Severidade Crítico

Título do bug: Ajuste em link

Ao acessar a tela <https://adaptliviu.com.br/liviu-lms/> em “Tenha diferenciais que vai engajar seus alunos de forma interativa” em “Agende uma demonstração” está fora do formato de botão como nas demais telas.

1. Acessar <https://adaptliviu.com.br/liviu-lms/>
2. Rolar a página até em “Tenha diferenciais que vai engajar seus alunos de forma interativa”.



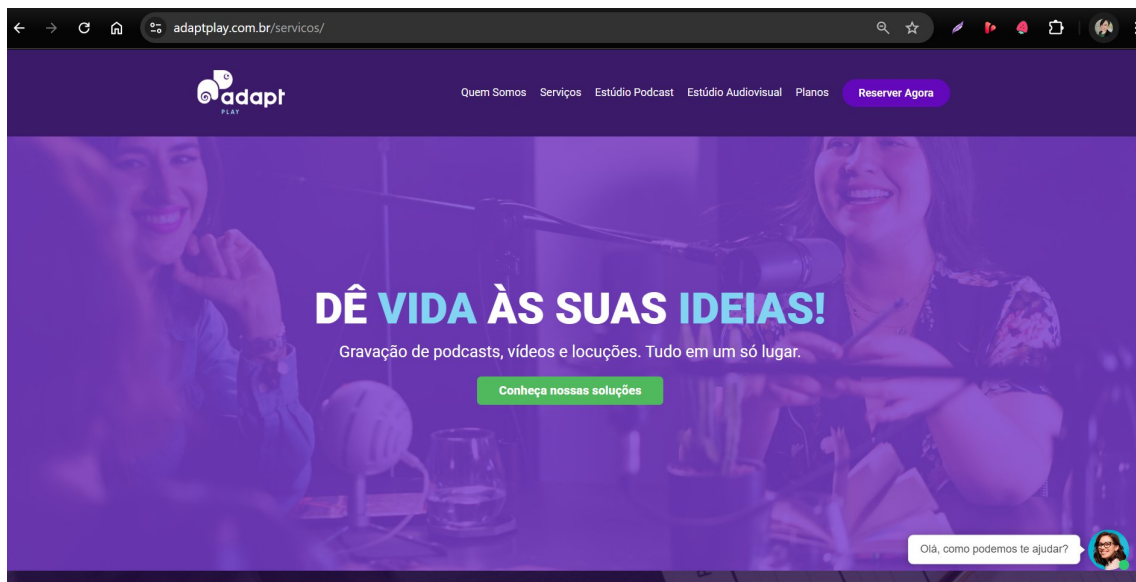


Severidade Baixo

Título do bug: Não direciona para a página de soluções

Ao acessar a tela <https://adaptplay.com.br/servicos/> não esta direcionando para as soluções do serviço

1. Acessar a tela <https://adaptplay.com.br/servicos/>
2. Clicar em “Conheça nossas soluções”



Severidade Alto

2. Proposta de Melhoria Contínua na Área de QA

Objetivo: Melhorar a eficiência, a cobertura e a qualidade dos testes realizados pela equipe de QA, além de otimizar os processos e reduzir o esforço manual.

1. Novas Ferramentas para Otimizar Testes

A introdução de novas ferramentas pode ser essencial para otimizar os testes, principalmente em áreas que exigem maior velocidade e precisão. Algumas sugestões são:

- **Cypress:** Para testes end-to-end, é uma ferramenta poderosa para testar interfaces de usuário de forma rápida e eficiente. Ela permite a automação dos testes com uma API simples e possui uma excelente integração com CI/CD.
- **Selenium (com WebDriverIO):** Para testes de interface em múltiplos navegadores e dispositivos, o Selenium, combinado com o WebDriverIO, pode proporcionar uma automação mais flexível.
- **TestRail:** Para gerenciar e documentar os testes de forma eficiente, TestRail pode ser uma boa opção para integrar as execuções manuais e automatizadas, aumentando a rastreabilidade.
- **Jira + Xray:** Para integração entre testes e gerenciamento de tarefas, permitindo que os testes automatizados e manuais sejam vinculados diretamente aos tickets do Jira.

2. Processos para Aumentar a Eficiência e Cobertura dos Testes

A eficiência e a cobertura dos testes podem ser melhoradas com a implementação de processos de testes bem definidos:

- **Planejamento de Testes Baseado em Riscos:** Identificar áreas críticas do sistema que mais impactam os usuários e garantir que essas áreas sejam testadas com maior intensidade.
- **Testes de Regresso:** Implementar uma suíte de testes de regressão automatizada para garantir que mudanças no sistema não afetem funcionalidades já testadas.
- **Testes Exploratórios:** Incentivar a equipe a realizar testes exploratórios com base nas funcionalidades críticas do sistema. Estes testes podem ser realizados de forma manual, com o objetivo de encontrar bugs que não foram previstos.
- **Testes de Performance e Stress:** Integrar testes de performance desde as fases iniciais do desenvolvimento, usando ferramentas como JMeter ou Gatling.

3. Uso de Métricas e Indicadores de Qualidade

Para melhorar o processo de QA, o uso de métricas e indicadores pode ser essencial para avaliar e ajustar a qualidade das entregas. Algumas métricas a serem adotadas incluem:

- **Cobertura de Testes:** Mede a porcentagem de código que foi testada. Quanto maior a cobertura, menor a chance de bugs não detectados. Ferramentas como **Jacoco** ou **Istanbul** podem ser utilizadas para gerar relatórios de cobertura.

- **Taxa de Defeitos:** Medir o número de defeitos encontrados por fase (desenvolvimento, teste, produção). Isso ajuda a identificar pontos fracos no processo de desenvolvimento e testes.
- **Custo por Defeito:** Quantifica o impacto financeiro dos defeitos encontrados. Isso ajuda a justificar investimentos em ferramentas e processos para melhorar a qualidade do software.
- **Tempo de Execução dos Testes:** Monitorar o tempo total que a suíte de testes leva para rodar. O objetivo é manter esse tempo baixo sem comprometer a qualidade, utilizando testes paralelos e ferramentas de CI/CD.
- **Taxa de Automação:** A porcentagem de testes automatizados em relação ao total de testes. Um aumento nesta métrica pode liberar tempo da equipe de QA para testar casos mais complexos manualmente.

4. Automação de Tarefas Repetitivas para Reduzir o Esforço Manual

A automação é uma parte crucial para aumentar a eficiência da equipe de QA e reduzir o esforço manual. Algumas ações incluem:

- **Automatizar Testes de Regressão:** Criar e manter uma suíte de testes de regressão automatizada para garantir que as funcionalidades existentes não sejam impactadas por mudanças no código.
- **Integração Contínua (CI) e Entrega Contínua (CD):** Implementar pipelines de CI/CD para automatizar a execução de testes sempre que houver uma nova alteração no código. Ferramentas como **Jenkins**, **GitLab CI** ou **CircleCI** são indicadas para automatizar a execução de testes em múltiplos ambientes.
- **Automatizar Testes de Performance:** Ferramentas como **JMeter** podem ser integradas aos pipelines de CI/CD para rodar testes de carga sempre que houver alterações que possam impactar a performance.
- **Testes de API Automatizados:** Implementar ferramentas como **Postman** ou **Rest Assured** para automatizar testes de integração e API.

5. Exemplos de Boas Práticas já Aplicadas em Outros Projetos

A adoção de boas práticas pode ser crucial para garantir a qualidade e o sucesso dos testes. Alguns exemplos de boas práticas incluem:

- **Testes de Integração no Processo de Desenvolvimento:** Em projetos anteriores, implementamos testes de integração para garantir que diferentes partes do sistema funcionem juntas corretamente. Isso foi feito com o uso de ferramentas como **Postman** para testar APIs e

Cypress para testar a interação entre diferentes componentes de frontend.

- **Uso de Mocks e Stubs em Testes de API:** Utilizamos **WireMock** e **Mockito** para simular APIs externas durante os testes, garantindo que o sistema seja testado de forma independente de serviços externos.
- **Documentação Clara e Rastreabilidade de Testes:** Em outro projeto, integramos o **TestRail** com o Jira para rastrear casos de teste e garantir que todas as funcionalidades desenvolvidas fossem cobertas por testes. A documentação de cada caso de teste foi mantida atualizada para que qualquer membro da equipe pudesse entender o que estava sendo testado.