**Desafio QA**

1. **Na tela de inserção e listagem de Conteúdos, identifique no mínimo 3 requisitos básicos da funcionalidade.**

* Adicionar Conteúdos
* Listar Conteúdos
* Listar Conteúdo por categoria
* Pesquisa conteúdo por ID
* Pesquisar conteúdo por nome
* Excluir conteúdo
* Editar conteúdo

1. **Crie no mínimo 4 casos de testes relacionados a listagem, inserção, edição e exclusão de conteúdo, utilizando a técnica.**

* **Caso de Teste 1:** Adicionar Conteúdo  
    
  Acessar a página Conteúdo do site

Clicar no botão **“+ Adicionar”**

Preencher o campo **“Nome”**

Preencher a duração de exibição

Selecionar o arquivo a ser exibido

Selecionar a categoria

Clicar em **“Avançar”**

Preencher as datas de validade

Preencher as horas iniciais e finais

Selecionar os dias da semana

Clicar em **“Adicionar”**

Conferir se o conteúdo recém cadastrado foi listado no site

* **Caso de Teste 2:** Listar Conteúdo Não Cadastrado

Acessar a página Conteúdo do site

Clicar no botão “Limpar Dados”

Preencher o campo ID/Nome com o dado **“Teste 2”**

Clicar no botão **“Pesquisar”**

Conferir a mensagem **“Nenhum registro encontrado.”**

**Caso de Teste 3:** Listar Conteúdo Cadastrado  
  
Acessar a página Conteúdo do site

Clicar no botão **“Limpar Dados”**

Preencher o campo ID/Nome com o dado **“Teste 1”**

Clicar no botão **“Pesquisar”**

Conferir se o conteúdo pesquisado foi listado no site

* **Caso de Teste 4:** Edição de Conteúdo

Acessar a página Conteúdo do site  
Clicar no botão **“Alterar”** do conteúdo **“Teste 1”** listado no site

Alterar a duração para 1:30

Clicar em **“Alterar”**

Conferir se a alteração realizada é exibida nos dados do conteúdo listado

* **Caso de Teste 5:** Exclusão de Conteúdo

Acessar a página Conteúdo do site  
Clicar no botão **“Excluir”** do conteúdo **“Teste 1”** listado no site

Clicar em **“OK”** na mensagem de confirmação

Conferir a mensagem **“Registro excluído com sucesso.”**

1. **Escrever os testes automatizados para os cenários acima criados, utilizando uma linguagem para automatização de software (PHPUnit, JUnit, Cypress, Protractor, etc).**Os testes automatizados encontra-se nos arquivos **“Resource.robot”** e **“Conteudo.robot”.** Ambos podem ser abertos utilizando o Visual Studio Code
2. **Crie um relatório resumo de testes para apresentar os resultados dos testes executados. É importante que neste documento contenha os resultados das atividades de testes realizadas e avaliações baseadas nesses resultados, juntamente com sugestões de alterações que possam ser aplicadas, visando a qualidade do produto.**

**Testes Manuais:** Os testes criados na questão anterior foram executados com sucesso no modo manual e os prints dos resultados encontram-se no documento de evidências anexados ao email de resposta.

**Testes Automatizados:** Para realizar os testes foram utilizados os seguintes recursos: Robot Framework, Visual Studio Code, Cmder e Node JS.

Para ajudar na preparação do ambiente, coloquei junto à documentação um vídeo explicando toda a configuração de instalação das ferramentas necessárias.

Os testes automatizados foram criados na ferramenta **“Robot Framework”** e todos rodaram com sucesso no Google Chrome.

Testei os scripts individualmente, pois não consegui executar todos juntos, tal como gostaria. O motivo foi que tive dificuldade de automatizar as ações **“Anexar um arquivo"** e **“Clicar em OK na mensagem de confirmação”** (na exclusão de um conteúdo da lista.

Estas duas ações abrem elementos que não fazem parte do HTML e sim do sistema operacional.

Como estava preocupada em não perder o prazo para entregar o teste, não consegui pensar em uma solução para fazer o robot interagir com estes elementos.

Depois de rodar os testes, o robot cria os arquivos **“log.html”**, **“output.html”** e **“report.html”**, este último tem o relatório com o resultado dos testes.