

Evolução do Plano de Testes: API ServeRest

(Pós-Challenge 2)

Este documento detalha as melhorias e ajustes aplicados ao Plano de Testes da API ServeRest, seguindo o feedback recebido e os requisitos da tarefa de preparação para a automação com o **Robot Framework**.

1. Rastreabilidade e Gestão de Testes (Jira e QAlity)

O Plano de Testes foi migrado para uma ferramenta de Gestão de Ciclo de Vida de Testes (TCLM) para maior rastreabilidade e visão gráfica de progresso:

- **Instalação e Migração para o QAlity:** O plugin **QAlity - Test Management for Jira** foi instalado e configurado no projeto individual.
- **Conversão de Cenários:** Todos os Cenários de Teste (CTs) mapeados no Challenge anterior foram convertidos para o tipo de item **QAlity Test** no Jira.
- **Ciclo de Teste:** Foi criado um **Ciclo de Teste (Test Cycle)** no QAlity para agrupar os testes. Este ciclo será usado para registrar a execução manual e gerar relatórios de progresso antes da implementação da automação.

2. Melhorias em Cobertura e Robustez (Aplicação do Feedback)

O feedback sobre "**testes de limite e reforço de mensagens de erro**" foi atendido com a expansão da cobertura e ajustes na priorização e evidências.

A. Inclusão de Novos Cenários de Teste

ID	Módulo	Descrição do Novo CT	Justificativa (Aplicação do Feedback)
CT-LOGIN-03	Login	Falha na autenticação ao usar a senha com tamanho abaixo do limite mínimo e acima do limite máximo.	Aplica a técnica de Análise de Valor Limite para testar as fronteiras de segurança.
CT-PRODUTO-04	Produtos	Validação do Status Code 400 e da Mensagem de Erro Específica ao tentar cadastrar um produto com nome duplicado.	Reforça as mensagens de erro, garantindo clareza na resposta da API, conforme feedback.

CT-CARRINHO-02	Integração	Falha ao tentar excluir um produto que está ativo em um carrinho (validação de integridade de dados).	Aumenta a cobertura das Regras de Negócio e de Dependências Lógicas (visão de risco).
-----------------------	-------------------	--	--

B. Ajustes em Prioridade e Evidências

- **Priorização:** O cenário de validação de segurança **CT-PRODUTO-03** (Usuário comum é barrado em rotas de admin - **403 Forbidden**) foi promovido para a prioridade **P1 - Crítico**. Este teste é vital e será incluído na suíte inicial de **Smoke Test**.
- **Evidências:** Para aprimorar a qualidade das **evidências**, foi padronizado que o registro de execução de cada teste deve incluir, obrigatoriamente, a captura do **Status Code HTTP**, **Headers de Resposta** e o **Corpo de Resposta (Payload JSON)**, fornecendo uma evidência estrutural completa.

3. Estratégia de Automação com Robot Framework

A estratégia de automação foi definida para otimizar o desenvolvimento no próximo Challenge:

- **Ferramenta:** A automação será desenvolvida utilizando o **Robot Framework** em conjunto com a biblioteca **RequestsLibrary**.
- **Escopo:** Serão automatizados 100% dos testes **P1 (Críticos)** e **P2 (Altos)**, que formarão a **Suíte de Regressão de API**.
- **Gerenciamento de Teste:** O Robot será configurado para utilizar as *Keywords* de Setup e Teardown. Isso garantirá a criação e limpeza de dados (usuário, produto, token) dinamicamente, mantendo a independência e a estabilidade dos testes (automação bem construída).
- **Validação Detalhada:** O foco da automação será a validação em camadas, verificando explicitamente o **Status Code**, a **Estrutura do JSON** e o **Conteúdo da Mensagem de Erro**, garantindo que as validações do "caminho triste" sejam robustas.