Challenge ao longo da semana

Pré-Challenge:

- **Remoção de Cenários de Testes:**
  - **Dos Cenários de Testes para US 001 - [API] Usuários foi removido:**
    - **Usuários (US001):**
      - **CT08:** Listar Todos os Usuários
        - **Motivo:** Por que não compromete a qualidade dos testes, já que o comportamento básico da listagem será incidentalmente testado durante a execução de outros cenários relacionados ao endpoint `/usuarios`.
      - **CT14:** Excluir Usuário Existente
        - **Motivo:** A funcionalidade de exclusão pode ser validada através de outros cenários que testam endpoints relacionados à exclusão ou comportamento com IDs inexistentes.
      - **CT15:** Tentar Excluir Usuário Inexistente
        - **Motivo:** O cenário de teste **US001-CT15** foi removido porque a validação de exclusão de um usuário inexistente segue um padrão genérico implementado pela API para qualquer recurso não encontrado, como o retorno de um status HTTP 404 (Not Found). Testar essa lógica específica para o endpoint de usuários não é essencial.
  - **Dos Cenários de Testes para US 003 - [API] Produtos foi removido:**
    - **Produtos (US003):**
      - **CT04:** Listar Todos os Produtos Cadastrados
        - **Motivo:** O cenário **US003-CT04** foi removido por se tratar de uma funcionalidade básica e de baixo impacto, cuja execução bem-sucedida é implicitamente validada por outros cenários mais críticos envolvendo a manipulação e gerenciamento de produtos.
      - **CT05:** Buscar Produto por ID Válido
        - **Motivo:** O cenário **US003-CT05** foi removido por ser uma operação simples e de baixo impacto, cuja validação é incidentalmente realizada em outros testes mais relevantes que dependem da busca de produtos por ID válido.
      - **CT06:** Buscar Produto por ID Inexistente
        - **Motivo:** O cenário **US003-CT06** foi removido por ser uma validação genérica e de baixo impacto, já coberta implicitamente pela lógica padrão da API para retornos de recursos não encontrados (HTTP 404).
      - **CT12:** Tentar Excluir Produto Inexistente
        - **Motivo:** O cenário **US003-CT12** foi removido por validar um comportamento padrão da API para exclusão de recursos inexistentes, considerado de baixo risco e amplamente coberto por outros testes gerais.
  - **Dos Cenários de Testes para US 004 - [API] Carrinhos foi removido:**
    - **Carrinhos (US004):**
      - **CT01:** Listar carrinhos cadastrados com sucesso.
        - **Motivo:** O cenário **US004-CT01** foi removido por ser uma operação básica de baixa complexidade e impacto, cuja funcionalidade é validada indiretamente em testes mais críticos do fluxo de carrinhos.
      - **CT02:** Listar carrinhos quando nenhum carrinho está vinculado ao usuário.
        - **Motivo:** O cenário **US004-CT02** foi removido por apresentar baixo impacto funcional e não influenciar diretamente os fluxos críticos da aplicação.
      - **CT06:** Buscar carrinho por ID com sucesso.
        - **Motivo:** O cenário **US004-CT06** foi removido por ser uma operação simples e de baixo risco, cuja funcionalidade é validada indiretamente em testes mais abrangentes.
      - **CT08:** Concluir compra com sucesso.

        - **Motivo:** O cenário **US004-CT08** foi removido por representar um fluxo com baixo impacto em comparação a outros cenários prioritários que cobrem validações mais críticas.
    - **CT10:** Cancelar compra com sucesso.
        - **Motivo:** O cenário **US004-CT10** foi removido por ter impacto limitado nos fluxos principais e baixa probabilidade de causar falhas críticas.
    - **CT11:** Tentar cancelar compra sem carrinho ativo.
        - **Motivo:** O cenário **US004-CT11** foi removido por validar um caso de erro de baixo impacto e baixa probabilidade de ocorrência

- **Consequentemente, após a remoção dos cenários de testes:**
    - Renomeei os cenários de testes planejados com seus ID atualizados.
    - Atualizei a matriz de risco removendo os cenários de testes inexistentes e atualizando o ID dos necessários.
    - Atualizei a priorização da execução dos cenários de testes removendo os cenários inexistentes, atualizando os ID dos necessários e atualizando a baixa prioridade com cenários de testes que estavam na média prioridade.
    - Atualizei os testes candidatos a automação removendo os cenários inexistentes e atualizando o ID dos necessários.

- **Com o feedback obtido da apresentação:**
    - Consumi o conteúdo necessário de cobertura de testes e melhorei a minha cobertura, que estava no estado inicial com uma conta errada.

Dia 01:

- **Migração dos cenários de testes mapeados para o Jira usando a ferramenta QAlity:**
    - Usando o QAlity, criei Cycle Tests para cada endpoint da API ServeRest.
    - Usando o QAlity, migrei os cenários de testes mapeados para Test Cases.
    - Após criar os Test Cases, atribuí cada um ao Cycle Test correspondente ao endpoint a que pertencia.

Dia 02:

- **Melhoria na estrutura de testes da API ServeRest no Postman:**
    - Organizei melhor a estrutura de testes no Postman:
        - Criei pastas para cada endpoint da API. Dentro de cada uma dessas pastas, adicionei uma subpasta com o nome correspondente ao conjunto de cenários de testes, como "CTs para US 002 - [API] Login". Dentro dessas subpastas, organizei pastas individuais para cada cenário de teste, nomeadas, por exemplo, como "CT01 - Login com Credenciais Válidas", contendo as requests e fluxos relacionados.
    - Atualizei a cobertura de testes após validar todos os endpoints, com os testes refinados no Postman.

Dia 03:

- **Refinei os testes candidatos à automação:**
    - Analisei os cenários de testes e adicionei novos cenários de testes como candidatos a automação. Os novos candidatos são:
        - **US001 - CT03:** Criar usuário com e-mail inválido.
        - **US003 - CT08:** Tentar excluir produto que está associado a carrinhos.
        - **US004 - CT06:** Tentar cancelar compra sem token ou com token inválido.

- **Executei os cenários de teste (CTs US 001 - Usuários & CTs US 002 - Login) no Postman:**
  - Após a execução dos testes, relatei os resultados no Cycle Test correspondente na ferramenta QAlity Plus.
- **Refinei cenários de teste mapeados:**
  - Melhorei a gramática e o entendimento de alguns cenários que estavam confusos.

Dia 04:

- Criei e configurei toda a estrutura para executar os testes candidatos à automação no Robot Framework
  - A estrutura da pasta ficou:
    - Testes candidatos à automação no Robot Framework
      - ServeRest API
        - **keywords**
          - `carrinhos_keywords.resource`
          - `login_keywords.resource`
          - `produtos_keywords.resource`
          - `usuarios_keywords.resource`
        - **support**
          - `base.robot`
          - **common**
            - `common.resource`
          - **fixtures**
            - **static**
              - `json_login.json`
              - `json_usuarios.json`
          - **variables**
            - `variaveis_api.resource`
        - **tests**
          - `carrinhos_tests.robot`
          - `login_tests.robot`
          - `produtos_tests.robot`
          - `usuarios_tests.robot`

Dia 05:

- Relatei os resultados dos testes manuais restantes realizados no Postman, dentro do Cycle Test correspondente. (Dentro da ferramenta QAlity Plus.)
- Melhorei o entendimento do plano de testes e o que eu estava de fato fazendo.
  - Havia informações vagas em alguns pontos, como no “Objetivo do Plano” e “Escopo dos Testes”.
- Melhorei a cobertura de testes dos endpoints utilizando “heatmap” para facilitar a visualização.
- Adicionei cobertura usando heatmap para:
  - Cenários de Teste US 001 - [API] Usuários.
  - Cenários de Teste US 002 - [API] Login.
  - Cenários de Teste US 003 - [API] Produtos.
  - Cenários de Teste US 004 - [API] Carrinhos.

    - Incluí a cobertura geral dos cenários de testes manuais e dos cenários de testes candidatos a automação, apresentando o percentual de sucesso.
- Relatei os resultados dos testes candidatos à automação realizados no Robot Framework, organizando-os em um Cycle Test específico para os cenários automatizados.
- Relatei minha interpretação dos resultados obtidos nos testes manuais e automatizados.