**Alisson V. Morais de Almeida**

**Sistema de Gerenciamento de Tarefas (To-Do List)**

**Objetivo do Software:** Desenvolver um aplicativo de lista de tarefas simples para ajudar os usuários a acompanhar suas tarefas diárias.

**Requisitos Funcionais:**

1. O usuário deve ser capaz de criar uma nova tarefa, especificando um título e uma descrição.
2. As tarefas devem ser exibidas em uma lista na ordem em que foram criadas.
3. O usuário deve poder marcar uma tarefa como concluída.
4. O usuário deve poder editar uma tarefa existente.
5. Deve ser possível excluir uma tarefa.
6. O software deve permitir que o usuário filtre as tarefas por status (concluídas ou pendentes).
7. As tarefas concluídas devem ser arquivadas e podem ser visualizadas em uma seção separada.
8. O usuário deve poder definir uma data de vencimento para uma tarefa.
9. Deve haver uma opção de classificação para organizar as tarefas por data de vencimento, prioridade ou título.
10. O software deve ser acessível via web e dispositivos móveis (responsivo).

**Requisitos Não Funcionais:**

1. O aplicativo deve ser responsivo e funcionar bem em diferentes navegadores e dispositivos móveis.
2. O tempo de resposta do aplicativo deve ser rápido, com carregamento rápido das tarefas.
3. O sistema deve ser seguro, protegendo os dados do usuário e autenticando os usuários adequadamente.

**Com base no que você aprendeu sobre como testar aplicações, quais os testes você faria para validar os requisitos apresentados acima?**

Teste de criação de tarefa, onde o sistema testará a função usando mocks, e caso a operação seja realizada com sucesso o teste passará, de forma análogo o sistema testará a edição de tarefas.

Teste da listagem de tarefas onde o sistema esperará uma determinada lista e verificará sua equivalência com o resultado da listagem.

O sistema simulará ao marcar uma tarefa como concluída, e depois testará se esta tarefa esta realmente concluída.

O sistema excluirá uma tarefa e ira verificar se a mesma foi deletada na base de dados.

O sistema contará com testes de filtros onde ele irá usar uma lista (array) e ao aplicar os filtros na mesma ela deverá retornar os dados filtrados em uma nova lista.

O sistema testará se uma tarefa concluída realmente esta sendo arquivada, realizando uma assertion no banco de dados.

Será construído também testes de integração entre as funcionalidades da tarefa, validando se seu funcionamento conjunto esta conforme esperado.

Será feito também o teste de ponta a ponta, que validará o funcionamento gráfico e funcional do sistema.

Serão realizados testes no lighthouse, plataforma que fornece informações sobre a qualidade de um sistema web, com base nele serão propostas melhorias.

**Aplicativo de Entrega de Comida**

**Descrição**: Uma startup de entrega de comida planeja desenvolver um aplicativo para clientes pedirem comida online. Os clientes podem procurar restaurantes, fazer pedidos, pagar online e rastrear a entrega em tempo real. Os restaurantes parceiros também devem ter acesso a um painel para gerenciar pedidos.

* Realizem uma simulação de Reunião de Planejamento da Sprint, selecionando itens de um Product Backlog do aplicativo de Entrega de Comida descrito anteriormente .
* Criar um Sprint Backlog com os itens selecionados e definir um objetivo para a Sprint.

**Product Backlog:**

1. **Pesquisa de Mercado**: Coletar dados sobre as preferências dos clientes, concorrência e demanda local.
2. **Cadastro de Clientes:** Desenvolver a funcionalidade de cadastro de clientes, incluindo nome, endereço, e-mail e senha.
3. **Pesquisa de Restaurantes**: Implementar um mecanismo de busca para os clientes encontrarem restaurantes nas proximidades.
4. **Cardápio de Restaurantes:** Criar uma interface para os restaurantes cadastrarem seus cardápios com itens, preços e descrições.
5. **Fazer Pedidos:** Permitir que os clientes escolham itens do cardápio, adicionem ao carrinho e façam pedidos.
6. **Pagamento Online:** Integração de pagamento online, incluindo cartões de crédito e serviços de pagamento.
7. **Rastreamento de Pedidos:** Implementar um sistema de rastreamento em tempo real para os clientes acompanharem a entrega.
8. **Painel do Restaurante:** Criar um painel de controle para os restaurantes acompanharem os pedidos e gerenciarem estoque.
9. **Avaliações e Comentários:** Permitir que os clientes avaliem e deixem comentários sobre os restaurantes e pedidos.
10. **Seleção de Itens para a Sprint:** Para esta simulação, selecionaremos os seguintes itens do Product Backlog para a Sprint:
11. **Cadastro de Clientes:** Esta é uma funcionalidade fundamental para que os clientes possam começar a usar o aplicativo. Isso inclui criar uma conta e fazer login.
12. **Pesquisa de Restaurantes:** Para permitir que os clientes encontrem restaurantes, é importante implementar a funcionalidade de pesquisa de restaurantes.
13. **Fazer Pedidos:** A capacidade de fazer pedidos é crucial, pois é o principal objetivo do aplicativo. Vamos incluir a seleção de itens do cardápio e a adição ao carrinho.

**Objetivo da Sprint:** O objetivo desta Sprint será criar as funcionalidades básicas para os clientes iniciarem a experiência de uso do aplicativo. Isso inclui a capacidade de criar uma conta, fazer login, pesquisar restaurantes e fazer pedidos. Com essas funcionalidades em vigor, os clientes poderão começar a explorar os restaurantes, escolher refeições e fazer pedidos.

**Sprint Backlog:** Com base nos itens selecionados, o Sprint Backlog incluirá tarefas específicas para cada um dos itens:

**Cadastro de Clientes:**

Criar a interface de cadastro de clientes.

Implementar a lógica de validação dos dados do cliente.

Desenvolver a funcionalidade de armazenamento de dados do cliente.

**Pesquisa de Restaurantes:**

Criar a interface de pesquisa de restaurantes.

Conectar-se à API de geolocalização para encontrar restaurantes nas proximidades.

Exibir resultados da pesquisa na interface do cliente.

**Fazer Pedidos:**

Criar a interface de seleção de itens do cardápio.

Implementar a lógica de adicionar itens ao carrinho.

Integrar o carrinho de compras com o processo de pagamento.

Parte superior do formulário