

Tema do produto/empresa:

- *Aplicativo de gestão de lista de compras com integração com mercados*

Conceito geral do tema:

- Este aplicativo permitiria aos usuários criar listas de compras personalizadas com base nas refeições que desejam preparar.

1. Requisitos:**1.1 Requisitos Funcionais:**

- Adicionar item.
- Remover item.
- Editar quantidade de itens.
- Escolher mercado.
- Remover mercado.
- Escolher o período para a lista ser comprada(diariamente, semanalmente, mensalmente, anualmente) e a duração do período.
- Checklists de se um item foi comprado.
- Criar lista de compra.
- Apagar a lista de compras.
- Editar lista de compra.

1.2 Requisitos não funcionais:

- A aplicação deve ter até **5** Listas individuais gratuitas.
- A aplicação deve ser responsiva e rápida, mesmo quando operando com grandes listas/quantidade de listas de compra ou em internet de baixa velocidade.
- A aplicação deve estar disponível para uso a qualquer momento, sem interrupções significativas devido a manutenção ou problemas técnicos.
- A aplicação deve ser capaz de lidar com um aumento no número de usuários e no volume de dados sem comprometer o desempenho.
- A aplicação deve ser compatível com uma ampla variedade de dispositivos.

1.3 Regras de Negócio:

- Política de Conteúdo: Os usuários são responsáveis pelo conteúdo das listas de compras que criam.
- Suporte ao Cliente: A empresa deve fornecer suporte ao cliente eficaz para lidar com consultas, relatórios de bugs e solicitações de recursos dos usuários de forma oportuna.
- Atualizações do Software: A empresa deve fornecer atualizações regulares de software para corrigir bugs, melhorar o desempenho e adicionar novos recursos à aplicação.

2. Plano de projeto com estratégias de testes unitários.

2.1 Introdução:

- Este aplicativo permitiria aos usuários criar listas de compras personalizadas com base nas refeições que desejam preparar. Tendo os testes em mente para que nenhum usuário tenha que lidar com erros ou bugs quando manejando suas listas de compras.

2.2 Escopo:

Queremos que estas 10 funcionalidades(As principais do projeto) estejam sempre em dia com nenhuma falta ou problema em suas ações.

10 funcionalidades Para testes:

1. Adicionar item *(se os itens estão sendo adicionados corretamente)*.
2. Remover o item *(se os itens estão sendo removidos corretamente)*.
3. Editar quantidade de itens *(se a quantidade dos itens estão sendo editados corretamente)*.
4. Escolher mercado *(se os mercados estão sendo escolhidos corretamente)*.
5. Remover mercado *(se os mercados estão sendo removidos corretamente)*.
6. Escolher período para a lista ser comprada seja diariamente, semanalmente, mensalmente, anualmente *(se a periodicidade está sendo escolhida corretamente)*.
7. Checklists de se um item foi comprado *(se os itens da checklists estão sendo marcados corretamente)*.
8. Criar lista de compra *(se as listas estão sendo criadas corretamente)*.
9. Apagar lista de compras *(se as listas estão sendo apagadas corretamente)*.
10. Editar lista de compra *(se as listas estão sendo editadas corretamente)*.

2.3 Estratégia de Teste:

- Testagem do máximo de funções possíveis(conhecido como Caixa Branca)
- Possíveis ferramentas de teste: Cypress, Jest, Mocha, Chai, etc.
- Seguindo padrões recomendados, tentaremos nos manter dentro desses parâmetros: Passar todos os casos de teste, Cobertura de código, Isolamento de dependências, Tempo de execução, Fácil manutenção, Reprodutibilidade, Identificação de erros, Integração com ferramentas de CI/CD, Conformidade com padrões de codificação, Documentação.

2.4 Ambiente de Teste:

- Hardware: qualquer PC médio comum.
- Software: Qualquer sistema que rode as ferramentas de teste, o projeto e Github no navegador.
- Configs: Garantido que funcione com Windows 10 e 11, VSCODE, arrumado com Commitlint, husky, Commitizen, Npm do Node.js(20.10.0) no Github com o Workflow do GitHub Actions.

2.5 Procedimentos de Teste:

- Os testes seguirão de acordo com o recomendado pela documentação da ferramenta, guias da internet e o suporte de um profissional.
 - Os testes das funcionalidades serão escritos e documentados de acordo com os métodos mais utilizados publicamente.
 - Se aplicável, os casos de teste serão automatizados seguindo os padrões da ferramenta escolhida.
 - Os resultados dos testes serão gerenciados via a ferramenta, suas funções e qualquer recomendação encontrada.
 - Qualquer defeito será comunicado a todos os membros, encontrado e resolvido da maneira mais prática e rápida possível em grupo via a ferramenta escolhida.

2.6 Cronograma de Teste:

- 3 Semanas para o desenvolvimento dos testes e aplicação da ferramenta.
- Necessária a entrega até dia 16/05/2024

2.7 Responsabilidades:

- As responsabilidades serão divididas igualmente entre todos os membros e dentro do possível cada etapa será feita em conjunto com todos os membros para maior eficiência.

2.8 Riscos e Mitigações:

- Os riscos dos testes são o entendimento da aplicação, como suas funcionalidades afetam uma à outra e a possibilidade de um erro massivo desmantelando o código por inteiro.
Possíveis Mitigações: Backups e um estudo do código, suas funções e ferramentas utilizadas.

2.9 Gerenciamento de Mudanças:

- Qualquer alteração será discutida em como afetará o projeto, por onde seria o mais correto começar a implementar, se é realmente necessária, quais outras alterações serão necessárias ser feitas para que tudo funcione corretamente, e tudo será idealmente feito em grupo ou com alguém que se voluntarie.

2.10 Aprovação:

- Paulo Roberto de Souza Carneiro Filho - 1272215237
- Lucca D'avilla Bacelar - 1272218370