



UC: Modelos, Métodos e Técnicas da Engenharia de Software - A3	
ALUNOS	RA:
Gustavo de Albuquerque Ramalho	12.822.216.012
João Victor Gomes Uchôa	12.822.212.148
Manoel Santos de Brito	1.282.227.687
Michell Bezerra de Sousa	12.822.125.581

Professores:

Presencial: WANFRANKLIN DE CARVALHO MOREIRA ALVES

On-line: EDYENE CELY AMARO OLIVEIRA

SUMÁRIO

1. INTRODUÇÃO	3
2. OBJETIVO DO PROJETO	3
3. ESCOPO DO PROJETO	3
4. FUNCIONALIDADES PRINCIPAIS.....	3
5. BANCO DE DADOS.....	3
6. TECNOLOGIAS UTILIZADAS	4
6.1. NO FRONT-END: SERÁ UTILIZADO:.....	4
6.2. NO BACK-END SERÁ USADO:	4
6.3. BANCO DE DADOS:	4
7. METODOLOGIA ÁGIL PARA REALIZAÇÃO DO TRABALHO.....	4
7.1. CLIENTE PRESENTE:	5
7.2. PLANEJAMENTO:.....	5
7.3. STAND UP MEETING:.....	5
7.4. PROGRAMAÇÃO EM PAR:	5
7.5. TESTES CONSTANTES:.....	5
7.6. REFATORAÇÃO:	5
8. DESENVOLVEDORES DO SISTEMA	6
9. REQUISITOS DO SISTEMA	6
9.1. REQUISITOS FUNCIONAIS:.....	6
9.2. DIAGRAMA DE CASO DE USO	6
9.3. REQUISITOS NÃO-FUNCIONAIS:.....	7
10. PROTÓTIPO	7
11. MÉTRICAS	8
12. CRONOGRAMA:	8
13. ARQUITETURA DO SISTEMA	9
14. CONCLUSÃO	9

1. INTRODUÇÃO

Este documento visa fornecer um roteiro para o desenvolvimento do site de pedidos, que será utilizado pela DEVELOPER LANCHES®, pequeno carrinho de cachorro-quente, e futura grande empresa do ramo alimentício. Com base nessas informações desenvolvemos um sistema web para pedidos de lanches, oferecendo ao pequeno empreendedor uma experiência intuitiva e eficiente para a organização de suas vendas de cachorro-quente.

2. OBJETIVO DO PROJETO

Fornecer tecnologia aos pequenos empreendedores, para que eles tenham acesso aos dados do seu pequeno estabelecimento, como produtos mais vendidos, o valor total em reais vendido no dia e algumas outras informações. Tudo na palma da sua mão.

3. ESCOPO DO PROJETO

Foi questionado pelo microempreendedor que o mesmo não sabia bater o caixa ao final de sua atividade (carrinho cachorro-quente), o mesmo não sabia o quanto tinha vendido ou se tinha passado troco errado, nem sabe ao certo qual o produto mais vendido ou seja o mais procurado.

De posse dessas informações, foi acordado fazer um pequeno site que tenha os produtos que o pequeno comerciante vende, com os valores de cada item, e com a possibilidade de o cliente escolher vários itens e ao final do pedido o sistema terá uma gestão de carrinho, onde o cliente pode escolher itens, visualizar preços e adicionar/remover produtos conforme sua preferência.

4. FUNCIONALIDADES PRINCIPAIS

O sistema terá uma interface amigável de Seleção de Produtos. Que permitirá aos usuários visualizarem as opções de lanches disponíveis com seus respectivos valores. Cada opção terá uma descrição detalhada, incluindo ingredientes, preço e imagem representativa do produto, o carrinho de compras com os itens de seu pedido e valor total a pagar, o carrinho também exibirá de forma clara os itens selecionados, quantidades, preços individuais e o total acumulado em Reais das vendas feitas no dia, a possibilidade de retirar itens do carrinho será fornecida.

5. BANCO DE DADOS

Utilização de um banco de dados para armazenar informações sobre produtos vendidos, e o valor recebido para que ao final do expediente o pequeno empreendedor saber qual valor

arrecadado e assim fazer a conferência do caixa para saber se houve perdas ou se deu tudo certo. A estrutura do banco de dados incluirá tabelas para produtos vendidos, e valores recebidos.

6. TECNOLOGIAS UTILIZADAS

6.1. NO FRONT-END: SERÁ UTILIZADO:

O HTML: (linguagem de marcação utilizada na construção de páginas na Web, significa (HyperText Markup Language). Linguagem de Marcação de Hipertexto —, o HTML é o componente base da web. Isso quer dizer que ele permite a construção de websites e a inserção de novos conteúdos, como imagens e vídeos, por meio dos hipertextos.).

O CSS (sigla para Cascading Style Sheets, ou seja, Folhas de Estilo em Cascatas. É uma maneira de dar estilo ao código criado por linguagens como HTML, XML ou XHTML, por exemplo.)

O framework JQuery (que é uma biblioteca livre que contém funções da linguagem de programação JavaScript que interage com páginas em HTML, desenvolvida para simplificar os scripts executados/interpretados no navegador de internet do usuário.)

6.2. NO BACK-END SERÁ USADO:

Linguagem de programação Javascript, Framework de web API Nestjs.

6.3. BANCO DE DADOS:

Banco de dados não relacional MongoDB, utilizando o framework mongoose.

7. METODOLOGIA ÁGIL PARA REALIZAÇÃO DO TRABALHO

Para a organização e realização do projeto, será utilizada a metodologia XP (Extreme Programming), onde o foco é voltado para a agilidade na entrega e no aprendizado mútuo. Adaptamos a metodologia para o uso online do grupo, o XP possui um conjunto de princípios e valores, onde os princípios tendem a ser mais concretos que os valores. O conjunto de valores servem como um critério que norteiam as pessoas envolvidas no desenvolvimento do software, além de se complementarem. São eles: comunicação, simplicidade, feedback, coragem e respeito.

Além desses valores, existe um conjunto de princípios que deve ser seguido por equipes que forem usar XP em projetos, sendo o feedback rápido, presumir simplicidade, abraçar mudanças, trabalho de alta qualidade, pequenos passos, melhoria, diversidade, reflexão. As práticas consistem no núcleo principal do processo. Elas evidenciam os valores que nos ajudarão a ter sucesso no projeto. São elas:

7.1. CLIENTE PRESENTE:

O cliente deve participar ativamente do processo de desenvolvimento. Tudo precisa da comunicação com o cliente. Ele deve receber o melhor resultado possível a cada semana, ver o progresso no sistema, ser informado de mudanças de planos, etc. Escute, para que saiba qual é o problema a ser resolvido.

7.2. PLANEJAMENTO:

O desenvolvimento utilizando o XP é feito em iterações. Uma iteração é um período curto de tempo (1 ou 2 semanas) onde a equipe desenvolve um conjunto de funcionalidades. Sendo assim, no início da semana, desenvolvedores e clientes se reúnem para priorizar as funcionalidades. Essa reunião chama-se jogo de planejamento e nela já devem estar criadas as histórias. Se uma história for muito grande, ela deve ser dividida em tarefas com duração máxima de alguns dias. Essas histórias devem ser escritas pelo cliente, pois assim ele consegue pensar melhor em cada funcionalidade. O planejamento é importante para que você sempre faça a coisa mais importante a ser feita.

7.3. STAND UP MEETING:

São reuniões feitas em pé e de curta duração - mas muito produtiva, para que o time se mantenha alinhado, para saber o que cada um está fazendo exatamente, em que ponto está o projeto e se alguém está tendo problemas para executar suas tarefas. Ainda que apareça algum problema, essa reunião não tem o propósito de pensar em soluções.

7.4. PROGRAMAÇÃO EM PAR:

É uma programação em par (dupla) em um único computador. Como é apenas um computador, o software sempre é revisto por duas pessoas diminuindo assim a possibilidade de falhas. Busca-se sempre a evolução da equipe melhorando a qualidade do código fonte. Ela é uma das práticas primordiais do XP, pois dois programadores fazendo o trabalho juntos acaba agregando muito para o trabalho em equipe.

7.5. TESTES CONSTANTES:

É utilizado o Desenvolvimento Orientado a Testes (Test Driven Development), o conhecido TDD. Primeiro crie os testes unitários e depois crie o código para que o teste funcione, essa abordagem é complexa no início, mas os testes unitários são essenciais para que a qualidade do projeto seja mantida.

7.6. REFATORAÇÃO:

É um processo que permite a melhoria contínua da programação, o mínimo de introdução de erros e mantendo compatibilidade com o código já existente. Refatorar

melhora a clareza, leitura do código e facilita a manutenção. Além disso, o código fica mais coeso e você tem um melhor aproveitamento, evitando duplicação no código fonte.

8. DESENVOLVEDORES DO SISTEMA

Responsabilidades dos Desenvolvedores Front-end: Manoel Santos e Michell Bezerra: Implementação da interface do usuário (UI) do sistema. Desenvolvimento de componentes visuais interativos e responsivos. Colaboração com designers para garantir a consistência visual. Garantir uma experiência do usuário intuitiva e eficiente. Integração de elementos de design com o back-end.

Responsabilidades dos Desenvolvedores Back-end: Gustavo Albuquerque e João Victor: Desenvolvimento do Banco de Dados e Implementação de APIs para comunicação entre o front-end e o banco de dados. Garantir a segurança e integridade dos dados. Otimização do desempenho do sistema. Colaboração com desenvolvedores front-end para integração eficiente.

Responsabilidades dos Analista de Documentação: Manoel Santos, Michell Bezerra, Gustavo Albuquerque e João Victor: elaboração e manutenção da documentação do projeto com uma descrição detalhada da arquitetura do sistema, APIs e fluxos de trabalho.

9. REQUISITOS DO SISTEMA

9.1. REQUISITOS FUNCIONAIS:

RF1: Fornecer um carrinho de compras.

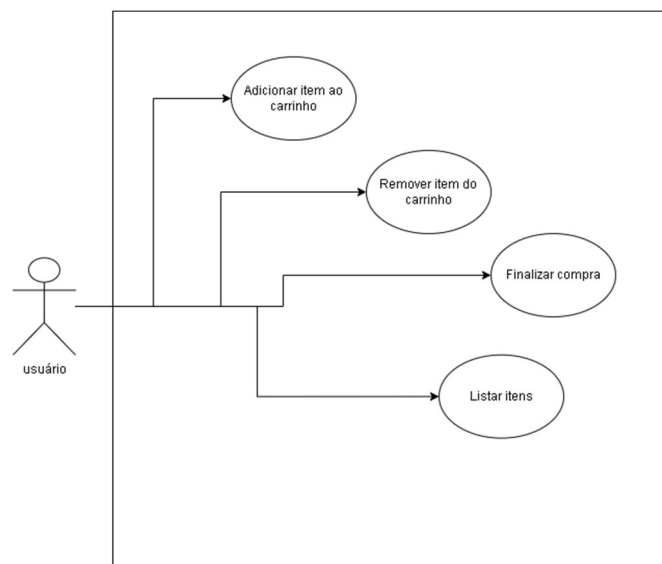
RF2: Permitir que o usuário adicione um item ao carrinho por vez.

RF3: Permitir que o usuário exclua um item do carrinho por vez.

RF4: Finalização da compra, clicando em um botão (Finalizar Compra) no carrinho.

RF5: Armazenar os itens do pedido, para relatórios futuros.

9.2. DIAGRAMA DE CASO DE USO



9.3. REQUISITOS NÃO-FUNCIONAIS:

Desempenho: Garantir uma resposta rápida da API, minimizando o tempo de carregamento.






Escalabilidade: Projetar a API para lidar com um aumento no número de usuários e pedidos sem comprometer a velocidade do web site.

Disponibilidade: Assegurar alta disponibilidade da API, minimizando tempo de inatividade por meio de redundância, backups regulares e procedimentos de recuperação de desastres.

Manutenibilidade: Desenvolver a API com código modular e documentação clara para facilitar a manutenção, atualizações e correções de bugs no futuro.

Compatibilidade: Garantir a compatibilidade da API com diferentes navegadores e dispositivos, assegurando uma experiência consistente para os usuários finais.

10. PROTÓTIPO

Developer Lanches		
Lugar de lanche e Programar Faça seu pedido		
Tradicional  Carne, Frango, Salcicha, Verduras R\$ 5,20 - 0 + Adicionar R\$ 0,00	Camarão  Camarão, Frango, Salcicha R\$ 8,00 - 0 + Adicionar R\$ 0,00	Carne de Sol  Carne de Sol, Salcicha, Cebola R\$ 8,00 - 0 + Adicionar R\$ 0,00
Gigante  Carne, Frango, Salcicha, Verduras R\$ 10,00 - 0 + Adicionar R\$ 0,00	Hot Dog Simples  Frango, Salcicha, Batata Palha R\$ 5,00 - 0 + Adicionar R\$ 0,00	Especial Dev.Jr  Salcicha, com Graxa R\$ 3,00 - 0 + Adicionar R\$ 0,00

11. MÉTRICAS

Taxa de Bugs Corrigidos:

Indica a eficiência da equipe na identificação e correção de problemas ao longo do desenvolvimento.

Cobertura de Testes:

Avaliar uma porcentagem do código que é testado, ajudando a garantir uma cobertura abrangente.

Tempo Médio de Resolução de Problemas:

Mede o tempo necessário para corrigir bugs ou resolver problemas identificados durante o desenvolvimento.

Satisfação do Cliente:

Obtida por meio de feedbacks e pesquisas, a satisfação do cliente é uma métrica chave para avaliar o sucesso do projeto.

Tempo de Resposta do Sistema:

Mede o tempo que o sistema leva para responder às interações do usuário, influenciando diretamente na experiência do cliente.

12. CRONOGRAMA:

Fase de Planejamento (Semana 1):

Definição de requisitos.

Elaboração do escopo do projeto.

Identificação de tecnologias e ferramentas.

Fase de Desenvolvimento Back-end e Front-end (Semana 2):

Implementação da lógica de negócios e funcionalidades.

Desenvolvimento de APIs para comunicação.

Implementação da interface do usuário.

Integração com o desenvolvimento back-end.

Fase de Testes e Documentação (Semana 3):

Execução de testes de unidade, integração e sistema.

Resolução de bugs identificados.

Atualização da documentação técnica.

Fase de Implementação (Semana 4):

Lançamento do sistema.

Monitoramento e ajustes pós-implantação.

13. ARQUITETURA DO SISTEMA

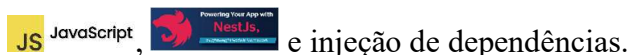
Banco de Dados:  mongoDB.

Ambiente de Desenvolvimento:  Visual Studio Code.

Ferramentas de Desenvolvimento Front-end:



Ferramentas de Desenvolvimento Back-end:



e injeção de dependências.

Protocolos de Comunicação: HTTP/HTTPS.

14. CONCLUSÃO

Ao longo deste documento, delineamos a concepção, desenvolvimento e implementação do sistema de pedidos para a DEVELOPER LANCHES®. Nosso objetivo principal era oferecer uma solução tecnológica acessível aos pequenos empreendedores, proporcionando-lhes uma gestão eficiente das vendas de seus estabelecimentos.

O projeto focou em suprir as necessidades específicas do microempreendedor, que enfrentava desafios na contabilidade e gestão de seu carrinho de cachorro-quente. A solução proposta consiste em um site intuitivo, possibilitando ao cliente visualizar, selecionar e organizar seus pedidos de forma descomplicada.

As funcionalidades principais, destacadas na interface amigável de Seleção de Produtos, incluem um carrinho de compras claro e informativo, apresentando detalhes dos itens escolhidos e o valor total das vendas diárias. Com a integração de um banco de dados não relacional MongoDB e tecnologias front-end como HTML, CSS e jQuery, garantimos uma experiência fluida e eficiente.

A metodologia ágil XP (Extreme Programming) foi adotada para promover agilidade, feedback constante e qualidade no desenvolvimento. A participação ativa do cliente, reuniões de planejamento, programação em par, testes constantes e refatoração contribuíram para a excelência do processo.

A equipe de desenvolvimento, composta pelos dedicados Manoel Santos, Michell Bezerra, Gustavo Albuquerque e João Victor, desempenhou papéis cruciais na implementação bem-sucedida. Os analistas de documentação também desempenharam um papel fundamental na elaboração e manutenção desta documentação detalhada.

Os requisitos do sistema, tanto funcionais quanto não funcionais, foram considerados meticulosamente para assegurar desempenho, escalabilidade, disponibilidade, manutenibilidade e compatibilidade com diferentes ambientes.

O cronograma delineado, desde a definição de requisitos até a implementação e monitoramento pós-implantação, foi seguido para garantir o sucesso do projeto. Além disso, métricas como taxa de bugs corrigidos, cobertura de testes, tempo médio de resolução de problemas, satisfação do cliente e tempo de resposta do sistema foram estabelecidas para avaliar e aprimorar continuamente o sistema.

Em conclusão, a criação deste sistema de pedidos representa não apenas uma solução tecnológica eficaz, mas também um compromisso com o fortalecimento dos pequenos empreendedores. Estamos confiantes de que este projeto proporcionará benefícios significativos à DEVELOPER LANCHES®, contribuindo para o crescimento e sucesso do negócio.