

Fatec Araras “Antônio Brambilla”

Documentação de Software
Casa Sônia Fashion
Projeto Integrador - 2º Semestre - DSM

| <i>Integrantes</i> |
|--|
| <i>Lucas Eduardo Rosolem</i> |
| <i>Renan Augusto Eugênio Marques</i> |
| <i>Jefferson Felipe de Moura</i> |
| <i>Felipe Rafael Rocha</i> |
| <i>Wilson Geraldo Donizeti Pereira</i> |
| <i>Bruna Cardoso</i> |

Cronograma Geral do Projeto

| Integrantes | Descrição | Data de Início | Data de Término |
|-----------------|--|----------------|------------------------------------|
| Lucas | Criação do repositório no Github | 02/10/2024 | 24/10/2024 |
| Grupo | Definição do projeto (Votação) | 08/10/2024 | 10/10/2024 |
| Renan | “Quadro de Trabalho”... Metodologia Ágil (Scrum) Ferramenta (Trello) | 12/10/2024 | Utilização até o final do projeto! |
| Lucas & Wilson | Codificação Front End | 05/10/2024 | ? |
| Renan Jefferson | Prototipação (Figma) | 14//10/2024 | 28/10/2024 |
| Lucas & Wilson | Diagramas UML | 11/10/2024 | 24/10/2024 |
| Lucas | Criação e Deploy da API | 09/11/2024 | 18/11/2024 |
| Lucas | Testes de Requisição HTTP | 16/11/2024 | 18/11/2024 |
| Grupo | Documentação | 12/10/2024 | 16/11/2024 |
| Lucas | Deploy FrontEnd | 20/11/2024 | 20/11/2024 |

Sumário

0.0 Cronograma Geral do Projeto

1. ESPECIFICAÇÃO

1.1 Introdução

1.2 Objetivo

1.3 Escopo

1.4 Descrição do Software

1.4.1 Visão Geral

1.4.2 Público-Alvo

1.5 Requisitos

1.5.1 Levantamento de Requisitos (LR)

1.5.2 Requisitos Funcionais (RF)

1.5.3 Requisitos Não Funcionais (RNF)

1.6 Modelagem de Software

Diagramas UML

Diagrama de Casos de Uso

Diagrama de contexto

Diagrama de sequência

Diagrama de atividade

2. DESENVOLVIMENTO

2.1 Metodologia de Desenvolvimento

2.1.1 Metodologia Ágil (Scrum)

2.2 Prototipação

2.2.1 Ferramentas Utilizadas

2.2.2 Descrição dos Protótipos

2.2.3 Validação dos Protótipo

2.3 Codificação

2.3.1 Ambiente de Desenvolvimento

2.3.2 Procedimentos de Desenvolvimento

2.4 Implementação

2.4.1 Controle de versionamento (Github)

2.4.2 Hospedagem (Vercel/ Netlify)

3. TESTE

3.1 Plano de Testes

3.1.1 Escopo de teste.

3.2 Manutenção

3.2.1 Plano de Manutenção

3.2.2 Suporte

4. EVOLUÇÃO

4.1 Histórico de Versões

4.2 Plano de Evolução

1. ESPECIFICAÇÃO

1.1 Introdução

Este documento descreve a especificação do projeto de uma aplicação web, para a loja de calçados “Casa Sônia”. A loja oferece uma variedade de calçados infantis, masculinos, femininos e esportivos.

1.2 Objetivo

O objetivo deste projeto é atualizar o site da empresa, proporcionando a atualização da imagem da empresa no digital, que atualmente pode ser visto no seguinte endereço eletrônico: <https://casasonia.com.br>, a proposta é desenvolver uma aplicação web no estilo “apresentação da empresa”, com layout mais atual que apresenta a loja e os produtos da loja de calçados de forma atraente e funcional, facilitando a navegação dos usuários e promovendo as vendas.

1.3 Escopo

O escopo do projeto inclui o desenvolvimento com as seguintes limitações:

- Cabeçalho com navbar
- Página inicial com destaques e alguns produtos usando um script js, para transcorrer as imagens .
- Formulário de newsletter para cadastro de usuário para promoções.
- Dark mode.
- O software terá 3 páginas-(Home/ Faq/ Produtos).
- Cards de produtos na página /produtos.
- Área de contato e informações da loja.
- Seção de Marcas e História da empresa.
- Drop Down na página /faq

- Desenvolvimento de Api em **NodeJs**- v18.17.0, juntamente com as bibliotecas **supabase-js** versão: 2.46.1, **Express-Js** versão: 4.21.1, **Dotenv** versão: 16.4.5; para requisições HTTP, com rotas GET e POST.
- Database as a Service PostgreSQL (DBaaS), **Supabase**, para armazenamento de dados.

1.4 Descrição do Software

1.4.1 Visão Geral

O software será desenvolvido em sua 1º versão utilizando tecnologias web modernas, como HTML5, CSS3, JavaScript.

1.4.2 Público-Alvo

O público-alvo da empresa são consumidores de todas as idades que procuram calçados de qualidade para diferentes ocasiões, incluindo pais em busca de calçados infantis, homens e mulheres em busca de calçados casuais e formais, e atletas em busca de calçados esportivos.

1.5 Requisitos

1.5.1 Levantamento de requisitos

O levantamento de requisitos foi feito através de uma breve reunião com o proprietário da loja, e questionários com clientes, para podermos identificar as principais necessidades e expectativas. Foi analisado que os clientes buscam uma página simples , visualmente atraente e fácil de navegar, com informação clara sobre a loja e os produtos.

1.5.2 Requisitos funcionais

| <i>Componentes</i> | <i>Descrição</i> |
|---------------------------|--|
| Body | <ul style="list-style-type: none"> • Botão ‘Scroll to Top’ |
| Header | <ul style="list-style-type: none"> • Links de acesso para a sessão desejada e outras páginas. (Desktop) • Botão hambúrguer que abre as opções de acessos (Mobile) |

| | |
|------------------------------|---|
| | <ul style="list-style-type: none"> • Botão ‘Dark Mode’, para poder deixar o layout da página com outra estilização. |
| Sessão Banner | <ul style="list-style-type: none"> • Slide com imagens e slogan da marca (Interação automático). |
| Sessão Marcas | <ul style="list-style-type: none"> • Slide com opção de ‘próximo’ e ‘anterior’ (Interação automática e manual), (Mobile). |
| Sessão Vitrine | <ul style="list-style-type: none"> • Slide com opção de ‘próximo’ e ‘anterior’ (Interação automática e manual). |
| Sessão Lojas | <ul style="list-style-type: none"> • Link no endereço para ser direcionado para o ‘google maps’. |
| Sessão Contato | <ul style="list-style-type: none"> • Links de redes sociais e contato (Whatsapp, Facebook, Instagram). • Newsletter (Usuário inserir nome e e-mail). • Acesso ‘Dúvidas frequentes’, leva para outra página. • Acesso ‘Política de privacidade ’, abre um documento PDF no navegador. • Acesso ‘termo de uso’, abre um documento PDF no navegador. |
| Páginas - Dúvidas Frequentes | <ul style="list-style-type: none"> • Dropdown (Abre texto oculto) |

1.5.3 Requisitos não funcionais:

| Código: | Tipo: | Descrição: |
|---------|-------------|--|
| RNF-01 | Usabilidade | A página deve ser fácil de navegar, com menus claros e de fácil acesso para diferentes seções. |
| RNF-02 | Desempenho | O tempo de carregamento da página não deve exceder 3 segundos em uma conexão média. |

| | | |
|--------|----------------|---|
| RNF-03 | Acessibilidade | Incluir descrições alternativas (alt text) nas imagens para acessibilidade. |
| RNF-04 | Acessibilidade | Contraste de cores adequadas para facilitar a leitura. |

1.6 Modelagem do Software

Diagramas UML

UML (Unified Modeling Language) é uma linguagem padrão usada para especificar, visualizar, construir e documentar os artefatos de sistemas de software. É uma ferramenta amplamente utilizada no desenvolvimento de sistemas orientados a objetos. A UML permite que os desenvolvedores e stakeholders compreendam a estrutura e o comportamento do sistema de forma clara e organizada, utilizando diagramas que representam diferentes aspectos do software.

Os principais tipos de diagramas UML que foram usados no desenvolvimento deste projeto incluem:

Diagrama de Casos de Uso

Os casos de uso descrevem as interações típicas dos usuários com o sistema. Exemplos incluem:

- O usuário navega pela navbar podendo acessar outras páginas da aplicação.
- Usuário visualiza detalhes de um produto no componente de “cards”, acionando o botão comprar que o redireciona para o WhatsApp.
- O usuário pode acessar as redes sociais da empresa.

Este diagrama mostra os diferentes casos de uso e os atores envolvidos:

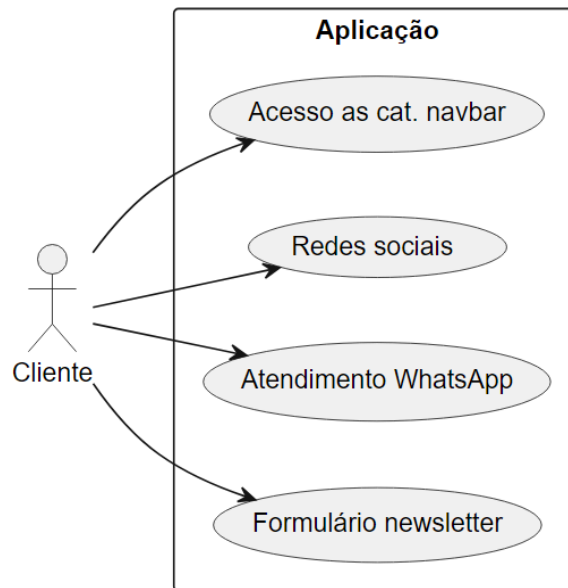


Diagrama de Contexto: Inscrição Newsletter

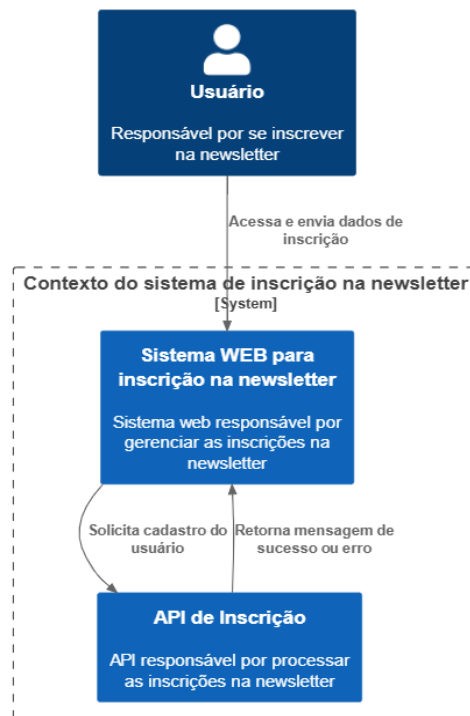


Diagrama de Sequência: Inscrição Newsletter

O diagrama de sequência ilustra a interação entre os objetos ao longo do tempo para um caso de uso específico.

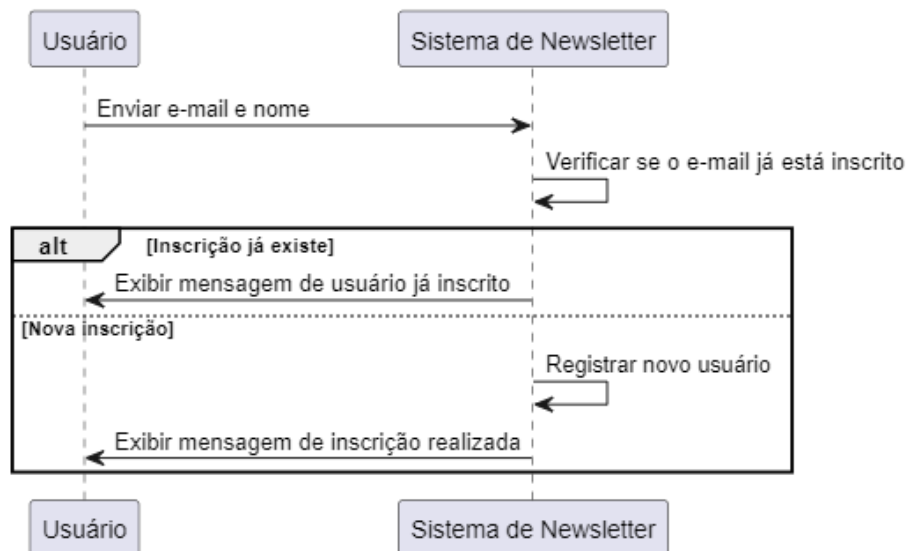
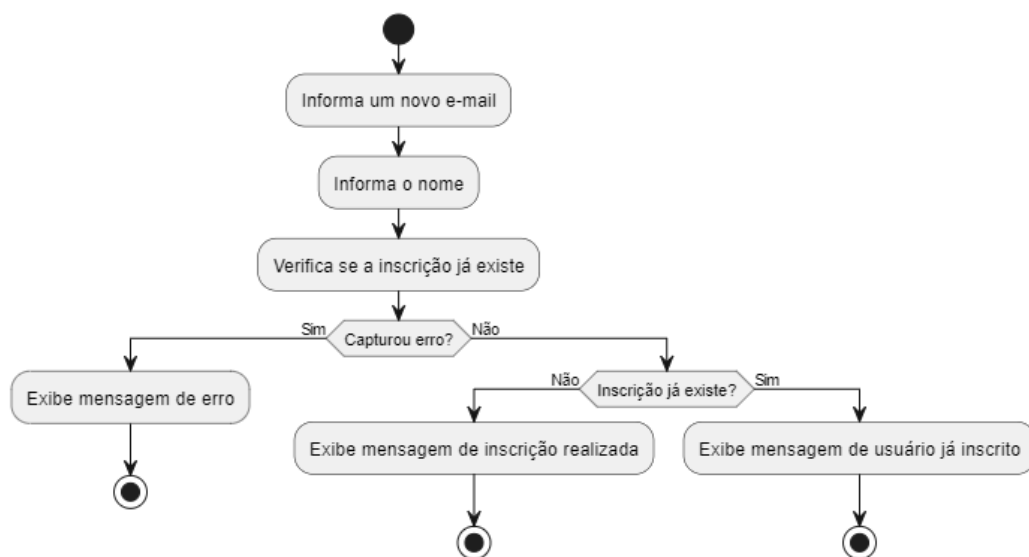


Diagrama de Atividade: Inscrição Newsletter

O diagrama de atividades representa o fluxo de atividades dentro de um processo ou caso de uso.

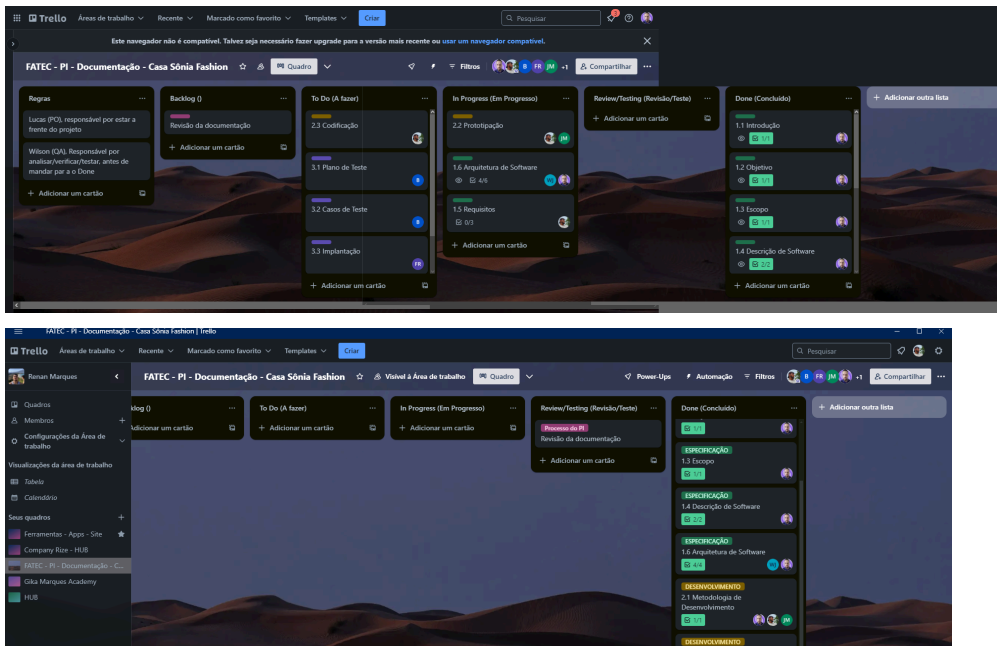


2. DESENVOLVIMENTO

2.1 Metodologia de Desenvolvimento

2.1.1 Metodologia Ágil (Scrum)

Para aplicação da metodologia ágil, foi utilizado o [Trello](#), solicite acesso para visualizar o quadro de gerenciamento do projeto, para o controle do desenvolvimento da documentação e do software, e as tarefas foram distribuídas entre os membros colaboradores.



2.2 Prototipação

2.2.1 Ferramentas Utilizadas

Foi utilizada duas ferramentas muito importantes para a criação dos design do software, o **Figma** e o **Canvas**.

2.2.2 Descrição dos Protótipos

Começamos com uma análise aprofundada do branding para definir o logotipo, paleta de cores e tipografia ideal. Com essa base, avançamos para o design da versão desktop, seguido pela adaptação para mobile. Durante todo o processo, selecionamos imagens de alta qualidade para enriquecer a prototipagem e garantir uma apresentação visual impactante.

2.2.3 Validação dos Protótipo

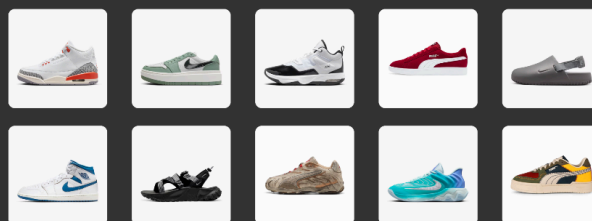
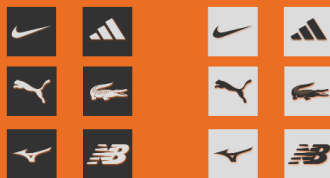
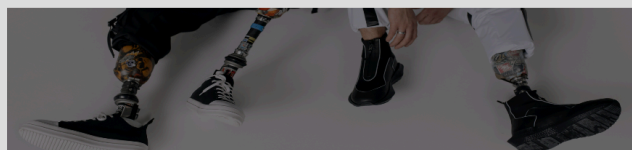
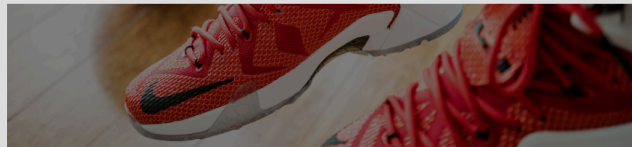
Branding:



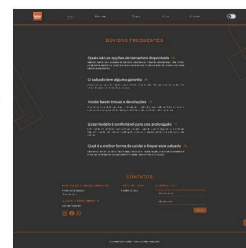
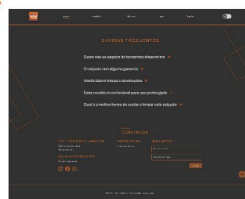
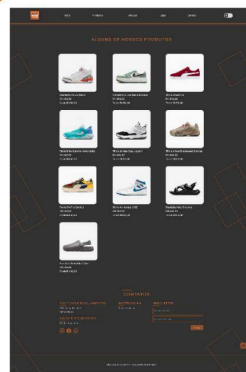
Imagens:

Imagens

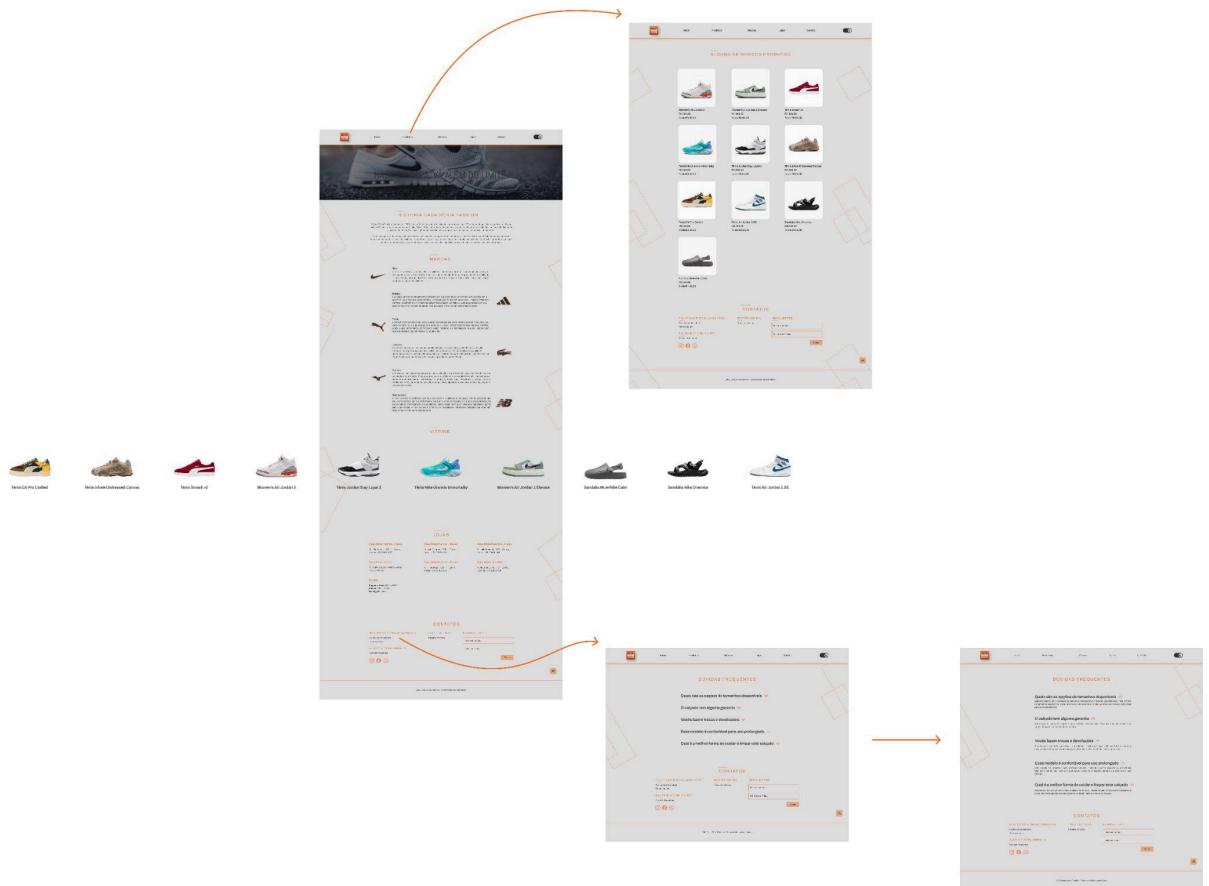
Salvar em SVG



Version [desk-top] [dark]:



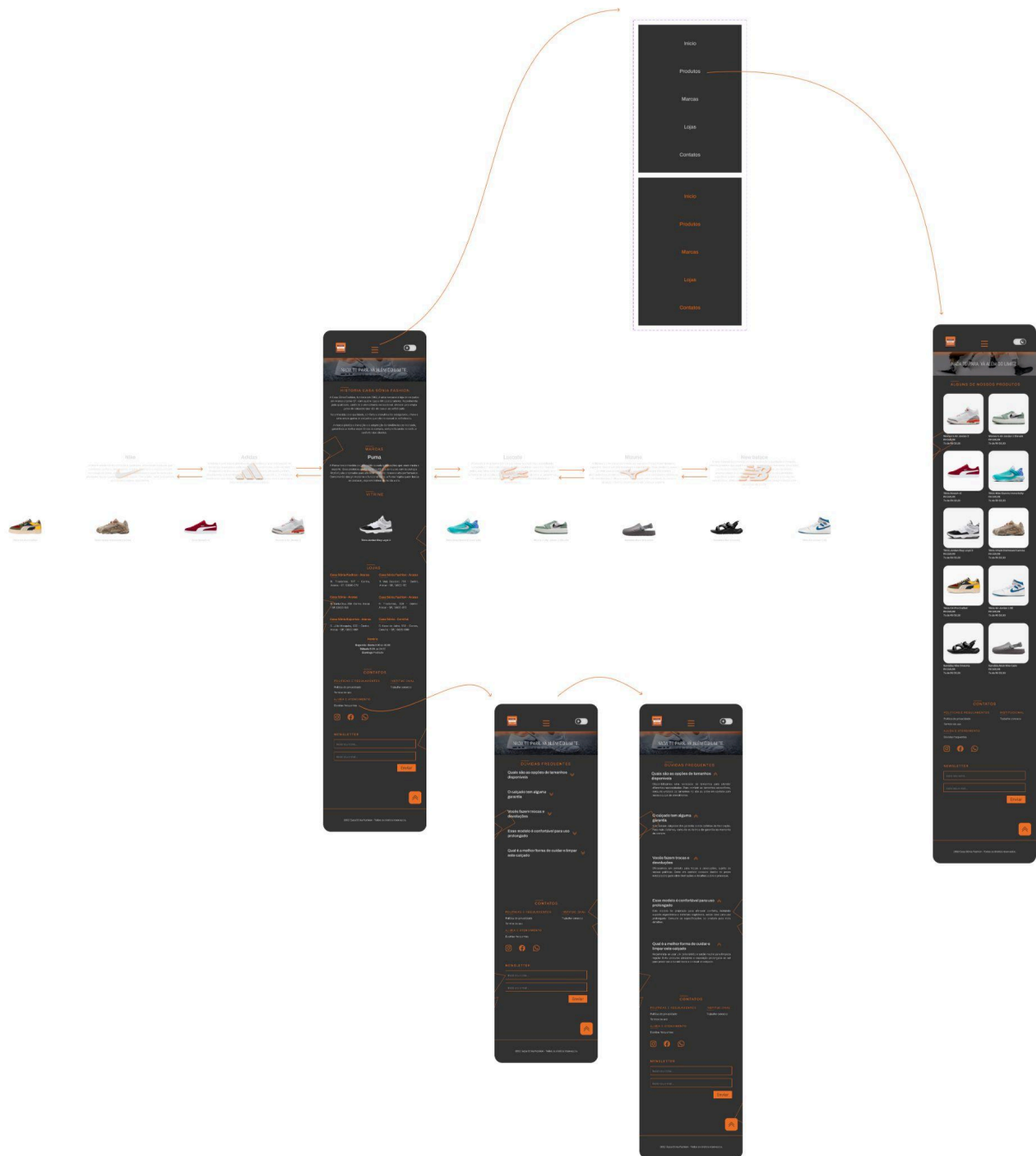
Version [desk-top] [light]:



Version [mobile] [light]:



Version [mobile] [dark]:



Nesta etapa, com o layout pronto, e validado pelo cliente, já é definido e pode seguir para a fase de desenvolvimento do software.

2.3 Codificação

2.3.1 Ambiente de Desenvolvimento

Para o desenvolvimento do software será utilizada a IDE [Visual Studio Code](#), em sistema operacional Windows 10 / Windows 11 .

2.3.2 Procedimentos de Desenvolvimento

Os Procedimentos do desenvolvimento do software será em etapas, sendo elas codificação do layout, funcionalidades requisitadas, consumo da API, teste de responsividade e teste de cadastro com a API, o procedimento para o desenvolvimento da API em [NodeJs-v18.17.0](#), serão: codificação de 3 rotas a rota GET: /, rota POST: /cadastrar e rota GET: /cadastrados, e teste de requisições HTTP.

2.4 Implementação

2.4.1 Controle de versionamento (GitHub)

O versionamento da documentação, bem como do código fonte do software, será feito em repositório do [GitHub](#), e será dividido em pastas, uma para documentação e outra para o software, após concluído a primeira versão terá um commit com o nome da versão concluída.

2.4.2 Hospedagem (Vercel-Netlify)

A hospedagem do código fonte do software da API, será feito nos servidores da [Vercel](#) e o FrontEnd, será hospedado na [Netlify](#).

3. TESTE

3.1 Plano de teste

O objetivo deste plano de teste é garantir que a página da aplicação web da loja “Casa Sônia” seja apresentada de maneira eficiente, e funcional. Isso inclui o navegar entre as páginas, a função do carrousel de produtos, a inscrição no newsletter, o encaminhamento para o WhatsApp e eficiência da venda e atendimento.

3.1.1 O escopo de teste inclui

- Navegação do web site (navbar e links).
- Funcionamento do carrousel de produtos
- Funcionalidade do formulário newsletter
- Teste de usabilidade de página em diferentes tipos de dispositivos.

- O redirecionar do botão para compra via WhatsApp
- Usabilidade do dark mode
- Verificação das páginas de categorias
- Teste de API, de requisições HTTP, utilizando a extensão do “VSCode” chamada: “**Rest Client**”.
- Teste contra ataque “SQL Injection”, no formulário de cadastro NEWSLETTER.

3.2 Manutenção

Esse plano de manutenção e suporte visa garantir que a aplicação web se mantenha funcional, segura e sempre pronta para oferecer uma experiência positiva aos usuários, incentivando as vendas e a fidelização.

3.2.1 Plano de Manutenção

Manutenção Preventiva Frequência: Mensal

- Verificação de atualizações de segurança nas bibliotecas utilizadas.
- Análise de desempenho da aplicação para identificar possíveis melhorias.
- Backup do banco de dados.

Manutenção Corretiva Frequência: Sob demanda (identificação do problema).

- Correção de bugs e erros reportados pelos usuários, como links quebrados ou formulários com problemas.
- Ajustes de layout, como problemas com a navbar, dark mode, ou problemas de exibição no carrossel de produtos.
- Teste de compatibilidade com novos navegadores e versões de navegadores existentes.

3.4.2 Plano de Suporte

- Canais de Suporte: Chat direto via WhatsApp.
- Disponibilidade: Horário comercial.
- Equipe responsável, equipe de desenvolvimento e manutenção do site.

4. EVOLUÇÃO

4.1 Histórico de Versões

Este documento será atualizado conforme o desenvolvimento do projeto avança. Abaixo está o histórico de versões:

Versão 1.0: Documento inicial criado.

4.2 Plano de Evolução

O plano de evolução inclui futuras melhorias e funcionalidades adicionais, como:

- Formulário de busca de produtos.
- Implementação de um sistema de e-commerce.
- Sugestão de compras de acordo com produtos já adquiridos pelo cliente.
- Adição de novas categorias de produtos conforme a demanda.

A versão inicial visou cumprir o objetivo de remodelar a imagem digital da empresa em um primeiro momento, para atender esta necessidade.