**Projeto ByteStore**

Professor: Geovane da Costa Oliveira

Engenharia de Software: 4° Semestre

Matéria: **PROJETO INTEGRADO DE SISTEMAS DE INFORMAÇÃO PARA NEGÓCIOS (ENSOF4BM1)**

Alunos: Aquiles,Hanniel,Danilo,Guilherme – Matutino – 4B

**SUMÁRIO**

CAPA

Página 2 -------------------------------------------------------------------------------- Sumário

Página 3 -------------------------------------------------------------- Requisitos Funcionais

Página 4 -------------------------------------------------------- Requisitos Não Funcionais

Página 5 ------------------------------------------- Requisitos de Domínio/Stakeholders

Página 6 --------------------------------- Elicitação de Requisitos/Desafios do projeto

Páginas 7,8,9 ----------------------------------------------------- Sub-Desafios do projeto

Página 9 ------------------------------------------- Membros do Grupo e suas Funções

Páginas 10,11,12,13,14 ----------------------------- Desenvolvimento e Metodologia

Páginas 15 e 16 ---------------------------------------------- Model Canva e Diagramas

Página 17 ----------------------------------------- Mapa de Navegação e Prototipagem

Página 18 ------- Conclusão / Trecho do Código (Função Adicionar ao Carrinho)

Página 19 -------------------------------------------------------------------- Prints do Projeto

Página 19,20,21 ------------------------------------ Referências, BPMN e Cronograma

### 1. Requisitos Funcionais

Os requisitos funcionais descrevem o que o sistema deve fazer. Eles são as ações e comportamentos que o usuário espera.

**RF1 - Autenticação e Gestão de Perfil de Usuário**:

* + **Sistema de Login/Logout:** O sistema deve oferecer um fluxo de autenticação seguro para que usuários existentes possam acessar suas contas. Isso inclui a funcionalidade de "esqueci minha senha" com recuperação via e-mail.
  + **Cadastro de Usuários:** Novos usuários devem poder se cadastrar fornecendo nome completo, e-mail e uma senha segura.
  + **Gerenciamento de Perfil:** Uma área de perfil deve permitir ao usuário visualizar e editar informações pessoais (endereço, telefone) e gerenciar suas credenciais de login.

**RF2 - Gestão de Catálogo e Produtos**:

* + **Variedade de Produtos:** A plataforma deve exibir produtos de forma clara e intuitiva, com categorias bem definidas para diferentes públicos (Moda Masculina, Moda Feminina).
  + **Funcionalidade de Busca e Filtros:** O usuário deve ter à disposição uma barra de busca eficiente e filtros avançados para refinar resultados por preço, tamanho,categoria e palavras-chave.
  + **Detalhes do Produto:** Cada produto deve ter informações como: fotos em alta resolução, e um painel para seleção de tamanho e quantidade.
  + **Lista de Desejos:** O usuário logado deve poder adicionar produtos a uma lista de desejos para compra futura.

**RF3 - Processo de Compra e Transação**:

* + **Carrinho de Compras:** O carrinho deve ser dinâmico, permitindo adicionar, remover a quantidade de itens. Ele deve exibir o subtotal e o total geral.
  + **Múltiplas Opções de Pagamento:** O sistema deve estar integrado a gateways de pagamento confiáveis (como PagBank, Mercado Pago ou Stripe) para processar transações com **cartão de crédito**, **PIX** e **boleto bancário**.
  + **Cálculo de Frete:** O cálculo de frete deve ser preciso, utilizando a API dos Correios ou de transportadoras parceiras, baseado no CEP de destino, peso e dimensões do produto.

### 2. Requisitos Não-Funcionais

Esses requisitos definem a qualidade e os atributos do sistema, mas não o que ele faz. Eles garantem a experiência do usuário e a confiabilidade do sistema.

**RNF1 - Desempenho e Velocidade:**

* + **Tempo de Resposta:** Todas as páginas e funcionalidades críticas (carregamento de produtos, busca, checkout) devem ter um tempo de resposta inferior a **0.50 segundos** em 90% dos casos.
  + **Capacidade de Carga:** O sistema deve ser capaz de suportar um tráfego de até **100 usuários simultâneos** sem degradação do desempenho, garantindo uma experiência fluida mesmo em picos de vendas.

**RNF2 - Usabilidade e Acessibilidade:**

* + **Interface Responsiva:** O design deve se adaptar perfeitamente a qualquer dispositivo (desktop, tablet, smartphone), garantindo uma navegação consistente.
  + **Design Intuitivo:** A arquitetura de informação e a navegação devem ser lógicas, permitindo que um usuário encontre e compre um produto em um número mínimo de cliques.

**RNF3 - Segurança e Confiabilidade:**

* + **Proteção de Dados (LGPD):** A plataforma deve estar em total conformidade com a **Lei Geral de Proteção de Dados (LGPD)**, com políticas de privacidade transparentes e a coleta mínima de dados.
  + **Transações Seguras:** Todas as transações financeiras devem ser criptografadas (HTTPS) e o sistema deve seguir os padrões de segurança do setor de cartões (PCI DSS).
  + **Tolerância a Falhas:** A arquitetura do sistema deve ser robusta o suficiente para lidar com falhas de componentes, garantindo alta disponibilidade.

**RNF4 - Escalabilidade e Manutenibilidade:**

* + **Sistema Escalável:** A infraestrutura deve ser capaz de escalar horizontalmente para acomodar o aumento de usuários e dados, sem a necessidade de reestruturação completa.
  + **Código Limpo:** O código-fonte deve ser modular, bem documentado e fácil de manter, permitindo a adição de novas funcionalidades no futuro.

### 3. Requisitos de Domínio

Esses requisitos são específicos do negócio e do setor de atuação, garantindo que o sistema se encaixe no contexto real.

**RD1 - Regulamentação Fiscal:**

* + **Sistema Tributário Brasileiro:** A plataforma deve ser capaz de aplicar a legislação tributária brasileira, calculando impostos (como ICMS e IPI) sobre as vendas. A integração com sistemas de emissão de nota fiscal eletrônica (NF-e) é um requisito-chave.

**RD2 - Políticas de Logística e Envio:**

* + **Políticas de Entrega:** O sistema deve refletir as políticas de frete de cada fornecedor, exibindo prazos de entrega estimados e custos de forma transparente.
  + **Código de Rastreio:** Cada pedido deve gerar um código de rastreio que será enviado ao cliente, permitindo o acompanhamento da entrega desde a postagem até a chegada.

### 4. Stakeholders

Os stakeholders são todas as pessoas ou grupos com interesse no projeto.

* **Fornecedores (Parceiros de Negócio):** São os pequenos e médios empreendedores que utilizam a plataforma para vender seus produtos. Seus interesses incluem a visibilidade de seus produtos, a facilidade de gestão de estoque e um processo de pagamento confiável.
* **Equipe Interna:**
  + **Gerente de Projeto (Aquiles):** Responsável por garantir que o projeto seja entregue no prazo e dentro do orçamento.
  + **Desenvolvedores (Guilherme e Hanniel):** A equipe técnica que implementa as funcionalidades do site.
  + **Analista de Requisitos/Testador (Danilo):** Garante que os requisitos do cliente sejam atendidos e que o sistema funcione sem falhas.
  + **Marketing e Vendas:** Responsáveis por atrair e converter clientes.
* **Empresas de Pagamento:** Provedores de serviços financeiros que asseguram a segurança das transações.
* **Transportadoras:** Empresas parceiras que garantem a logística e a entrega dos produtos.
* **Usuário Final:** O cliente que utiliza a plataforma para comprar e interagir com a loja.

### 5. Elicitação de Requisitos

A elicitação é o processo de coletar e entender os requisitos.

* **Brainstorming:** Sessões com a equipe interna para gerar e refinar ideias para funcionalidades e soluções técnicas.
* **Análise de Mercado:** Estudo de concorrentes e plataformas de e-commerce de sucesso para identificar as melhores práticas e funcionalidades essenciais.
* **Questionários e Entrevistas:** Aplicação de questionários e entrevistas com os principais stakeholders (gestor do projeto e potenciais empreendedores) para entender suas necessidades, desafios e expectativas.
* **Testes de Usabilidade e Feedback de Usuários:** Coleta de feedback de usuários finais por meio de protótipos e testes para garantir que a interface seja intuitiva.

### 6. Desafio do Projeto

O desafio é a meta principal a ser alcançada pelo projeto.

* **Desafio Principal:** O objetivo central é criar uma plataforma de e-commerce que não apenas sirva como um canal de vendas online, mas também como uma ferramenta acessível e completa para que **pequenos e médios empreendedores de roupas** possam digitalizar seus negócios, alcançando um público-alvo maior e escalando suas operações com eficiência.

### Sub-desafios do Projeto

O projeto ByteStore enfrenta alguns desafios técnicos e de negócio para cumprir seu objetivo de ser uma plataforma de e-commerce completa para pequenos e médios empreendedores.

* **Usabilidade e Acessibilidade**
  + **Objetivo:** Criar uma experiência de usuário (UX) intuitiva e uma interface de usuário (UI) atraente e fácil de navegar para o cliente
  + **Detalhes:**
    - **Interface Amigável:** A plataforma deve ser desenvolvida com um **design responsivo**, garantindo que a navegação e a compra sejam fluidas em qualquer dispositivo, seja um computador desktop, tablet ou smartphone.
    - **Fluxo de Compra Otimizado:** O processo de checkout precisa ser simplificado, exigindo o mínimo de cliques e informações do usuário para reduzir o abandono de carrinho.
* **Segurança e Confiabilidade**
  + **Objetivo:** Proteger os dados sensíveis dos usuários e garantir a integridade das transações financeiras.
  + **Detalhes:**
    - **Proteção de Dados Pessoais:** Implementar práticas de segurança em conformidade com a **Lei Geral de Proteção de Dados (LGPD)**, com criptografia de dados, políticas de privacidade claras e consentimento explícito do usuário.
    - **Transações Financeiras Seguras:** Utilizar o protocolo HTTPS para toda a comunicação e integrar o sistema a gateways de pagamento certificados (PCI DSS), garantindo que as informações de cartão de crédito e outras formas de pagamento sejam processadas de forma segura.
    - **Sistema Robusto:** Desenvolver uma arquitetura que inclua backups regulares do banco de dados e mecanismos de recuperação de desastres para garantir a **alta disponibilidade** da plataforma.
* **Escalabilidade**
  + **Objetivo:** Construir uma arquitetura flexível que suporte o crescimento do número de usuários, produtos e transações sem a necessidade de uma reestruturação completa.
  + **Detalhes:**
    - **Arquitetura Modular:** O código e a infraestrutura devem ser projetados de forma modular, facilitando a adição de novas funcionalidades e a otimização de componentes sem afetar o sistema como um todo.
    - **Otimização do Banco de Dados:** Estruturar o banco de dados de forma eficiente para lidar com um volume crescente de dados de produtos, usuários e pedidos, mantendo a velocidade de busca e a integridade das informações.

### Sub-desafios do Projeto

O projeto ByteStore enfrenta alguns desafios técnicos e de negócio para cumprir seu objetivo de ser uma plataforma de e-commerce completa para pequenos e médios empreendedores.

* **Usabilidade e Acessibilidade**
  + **Objetivo:** Criar uma experiência de usuário (UX) intuitiva e uma interface de usuário (UI) atraente e fácil de navegar para todos os públicos.
  + **Detalhes:**
    - **Interface Amigável:** A plataforma deve ser desenvolvida com um **design responsivo**, garantindo que a navegação e a compra sejam fluidas em qualquer dispositivo, seja um computador desktop, tablet ou smartphone.
    - **Fluxo de Compra Otimizado:** O processo de checkout precisa ser simplificado, exigindo o mínimo de cliques e informações do usuário para reduzir o abandono de carrinho.
    - **Acessibilidade (WCAG):** A interface deve seguir as diretrizes de acessibilidade web (WCAG) para garantir que usuários com deficiências visuais ou motoras possam navegar e interagir com o site por meio de leitores de tela e navegação por teclado.
* **Segurança e Confiabilidade**
  + **Objetivo:** Proteger os dados sensíveis dos usuários e garantir a integridade das transações financeiras.
  + **Detalhes:**
    - **Proteção de Dados Pessoais:** Implementar práticas de segurança em conformidade com a **Lei Geral de Proteção de Dados (LGPD)**, com criptografia de dados, políticas de privacidade claras e consentimento explícito do usuário.
    - **Transações Financeiras Seguras:** Utilizar o protocolo HTTPS para toda a comunicação e integrar o sistema a gateways de pagamento certificados (PCI DSS), garantindo que as informações de cartão de crédito e outras formas de pagamento sejam processadas de forma segura.
    - **Sistema Robusto:** Desenvolver uma arquitetura que inclua backups regulares do banco de dados e mecanismos de recuperação de desastres para garantir a **alta disponibilidade** da plataforma.
* **Escalabilidade**
  + **Objetivo:** Construir uma arquitetura flexível que suporte o crescimento do número de usuários, produtos e transações sem a necessidade de uma reestruturação completa.
  + **Detalhes:**
    - **Arquitetura Modular:** O código e a infraestrutura devem ser projetados de forma modular, facilitando a adição de novas funcionalidades e a otimização de componentes sem afetar o sistema como um todo, necessitando de poucas alterações.
    - **Otimização do Banco de Dados:** Estruturar o banco de dados de forma eficiente para lidar com um volume crescente de dados de produtos, usuários e pedidos, mantendo a velocidade de busca e a integridade das informações.

**MEMBROS DO GRUPO** e suas respectivas **FUNÇÕES:**

* Aquiles Leandro Alves Rocha: **Gerente de Projeto / GP**

Responsável por planejar, executar e monitorar o projeto. Ele define prazos e atua como a principal ponte de comunicação entre a equipe e as partes interessadas. É o líder que garante que o projeto seja entregue no prazo e dentro do escopo.

* Danilo Pereira Braga: **Analista de Requisitos / Testador**

Como Analista de Requisitos, ele pode ser responsável por coletar e documentar as necessidades dos clientes e do projeto. Como Testador, ele garante a qualidade do produto final, identificando bugs e validando se o sistema atende aos requisitos definidos. Ele trabalha em estreita colaboração com os desenvolvedores para garantir que o produto esteja livre de falhas antes do lançamento.

* Hanniel Santos de Alencar e Guilherme Santos Chedid: **Desenvolvedores / DEVS**

Responsáveis por escrever o código, construir as funcionalidades do sistema e integrar as diferentes partes do projeto. Eles trabalham em conjunto para resolver problemas técnicos, implementar as funcionalidades planejadas e garantir a qualidade e o bom funcionamento do software.

### 7. Desenvolvimento e Metodologia

**Desenvolvimento**

O desenvolvimento do trabalho ocorreu em etapas, de forma a aplicar os instrumentos metodológicos definidos, combinando a análise do negócio com a implementação técnica da solução.Inicialmente, a equipe mergulhou na Análise do Negócio, a fase crucial de entendimento dos requisitos. Nesse momento, o foco foi mapear o fluxo de trabalho do e-commerce, definir as funcionalidades essenciais como a navegação de produtos, o sistema de carrinho de compras e o gerenciamento de pedidos. Compreender o público-alvo e suas necessidades foi fundamental para moldar a experiência de usuário.

Em seguida, a fase de Planejamento e Design traduziu as necessidades do negócio em um plano técnico. Foram criados os wireframes e mockups da interface (UI) para garantir um design intuitivo. Simultaneamente, a arquitetura do sistema foi desenhada, incluindo a modelagem do banco de dados e a definição da lógica do backend em Node.js. A escolha dessas tecnologias foi estratégica para assegurar um desempenho rápido, escalabilidade e uma base sólida para o desenvolvimento futuro.

A fase de Implementação Técnica foi onde a solução ganhou vida. O frontend foi construído com HTML, CSS e JavaScript, seguindo o design planejado e as boas práticas de código. O backend em Node.js foi desenvolvido para gerenciar o fluxo de dados, conectando a interface do usuário ao banco de dados e garantindo a segurança das operações. A colaboração da equipe foi otimizada com o uso do Visual Studio Code, que permitiu um ambiente de trabalho unificado e eficiente.

Finalmente, a etapa de Testes e Otimização garantiu a qualidade do produto final. Cada funcionalidade foi exaustivamente testada para identificar e corrigir bugs, falhas de usabilidade e gargalos de desempenho. A aplicação foi submetida a testes de responsividade em diferentes dispositivos para assegurar que a experiência de compra fosse perfeita tanto no desktop quanto no mobile. Esse processo iterativo permitiu que o projeto não fosse apenas funcional, mas também robusto, seguro e preparado para crescer.

### 7.1. Diagnóstico e Coleta de Requisitos

A primeira fase incluiu um diagnóstico aprofundado para compreender a situação atual da empresa. Realizamos o mapeamento das práticas de controle de estoque e vendas, identificando falhas críticas como a falta de precisão na contagem de produtos, atrasos no atendimento ao cliente e a ausência de um registro digital dos processos.

Para complementar, foram realizadas entrevistas com o gestor e questionários com a equipe de vendas, o que permitiu identificar as principais dificuldades e a baixa familiaridade com ferramentas digitais. A análise de documentos revelou que as planilhas físicas usadas no controle de estoque apresentavam erros recorrentes e falta de padronização, reforçando a necessidade de uma solução digital.

### 7.2. Fundamentação Teórica e Proposta da Solução

A revisão bibliográfica foi crucial para a base do projeto, com a consulta a livros e artigos científicos sobre gestão de TI, comércio eletrônico e automação de processos. Esse referencial teórico serviu de alicerce para a elaboração da proposta de solução, que focou na implementação de um sistema de gestão integrado a uma plataforma digital. Os requisitos de baixo custo, facilidade de uso e escalabilidade futura foram a prioridade no planejamento.

### 7.3. Tecnologias utilizadas

* **HTML5 -> Estrutura** O **HTML5** foi a base para a criação da **estrutura e do conteúdo** de todas as páginas do site, como a página inicial (home.html), a página de produtos feminina (feminina.html) e a página do carrinho (carrinho.html), por exemplo. Ele define os elementos como cabeçalhos, parágrafos, imagens e botões, organizando a informação de maneira semântica e acessível.
* **CSS3 -> Design Responsivo** O **CSS3** foi usado para estilizar a plataforma e criar um **design responsivo e visualmente agradável**. Com ele, foram definidos o layout, as cores, as fontes e os espaçamentos. O CSS3 garantiu que a loja online se adaptasse a diferentes tamanhos de tela (computador, tablet e celular), proporcionando uma experiência de usuário consistente e intuitiva em todos os dispositivos.
* **JavaScript -> Interatividade** O **JavaScript** foi o motor do site, responsável por toda a **interatividade e a lógica de negócio**. Ele permitiu a criação de funcionalidades dinâmicas, como o sistema de filtros para produtos, a adição de itens ao carrinho de compras e o cálculo do total da compra. Além disso, o JavaScript gerencia a comunicação com o banco de dados e as operações de login e cadastro.
* **Node.js -> Backend** O **Node.js** foi utilizado para o **backend** do projeto, permitindo que a lógica de programação do JavaScript fosse executada no servidor. Essa tecnologia é ideal para gerenciar as rotas da aplicação, processar as requisições do usuário e realizar a comunicação com o banco de dados, garantindo um desempenho eficiente para toda a loja virtual.
* **VSCode -> Editor de Código** O **VSCode** (Visual Studio Code) foi a principal ferramenta de trabalho, funcionando como o **editor de código** do projeto. Sua interface e seus recursos, como extensões, depurador integrado e o sistema de controle de versão Git, otimizaram o desenvolvimento. Ele permitiu que a equipe trabalhasse de forma mais eficiente e colaborativa, facilitando a escrita e a organização do código.

**Metodologias**

A presente pesquisa caracteriza-se como um estudo de caso de caráter aplicado, com abordagem qualitativa e descritiva. Seu objetivo é analisar os problemas enfrentados na gestão manual de processos e propor uma solução em Tecnologia da Informação (TI) que seja viável e acessível.

### 7.4. Tipo de Pesquisa

* Natureza: aplicada, por buscar resolver um problema real.
* Abordagem: qualitativa, priorizando a interpretação e análise dos dados.
* Objetivos: descritiva, pois descreve a situação e propõe melhorias.
* Procedimento técnico: estudo de caso, voltado à realidade de uma pequena organização.

### 7.5. Fontes de Dados

* Primárias: entrevistas e questionários com o gestor e colaboradores.
* Secundárias: revisão bibliográfica (livros, artigos, documentos, legislações) e análise de relatórios/planilhas utilizadas nos processos manuais.

### 7.6. Instrumentos de Coleta de Dados

* Questionários estruturados para identificar falhas percebidas pelos usuários.
* Entrevistas semiestruturadas com o gestor para compreender limitações e necessidades.
* Análise documental das planilhas físicas para mapear inconsistências.
* Revisão bibliográfica para sustentar a solução proposta.

### 7.7. Procedimentos de Coleta e Análise

* Etapa 1: observação direta dos processos internos.
* Etapa 2: aplicação de entrevistas e questionários.
* Etapa 3: levantamento de referencial teórico.
* Análise: realizada por meio de análise de conteúdo, cruzando dados empíricos com fundamentos teóricos para propor uma solução adequada.

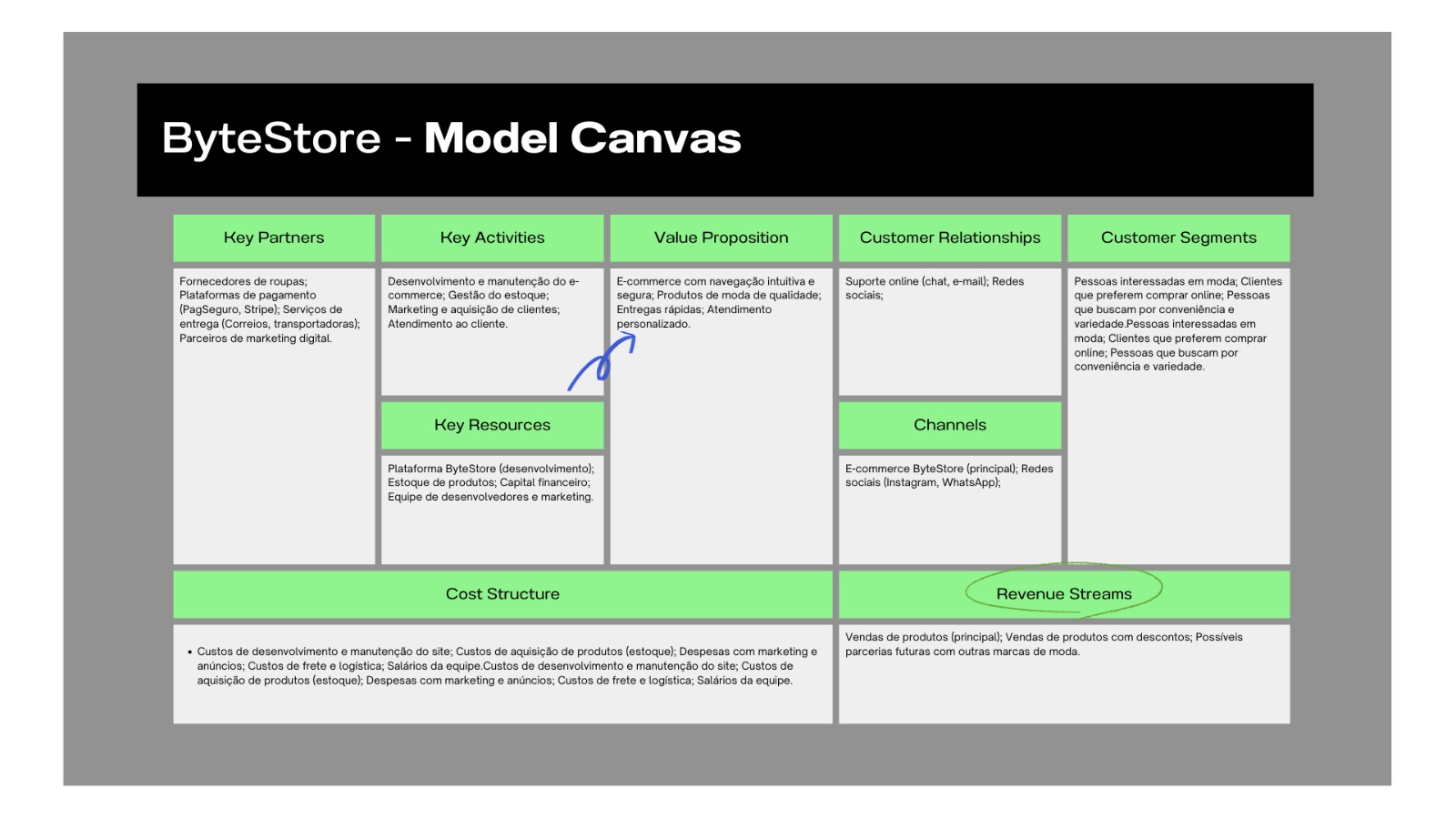
**7.8.Desenvolvimento e Implementação Técnica**

Com a proposta validada, partimos para o desenvolvimento da solução, utilizando as seguintes tecnologias:

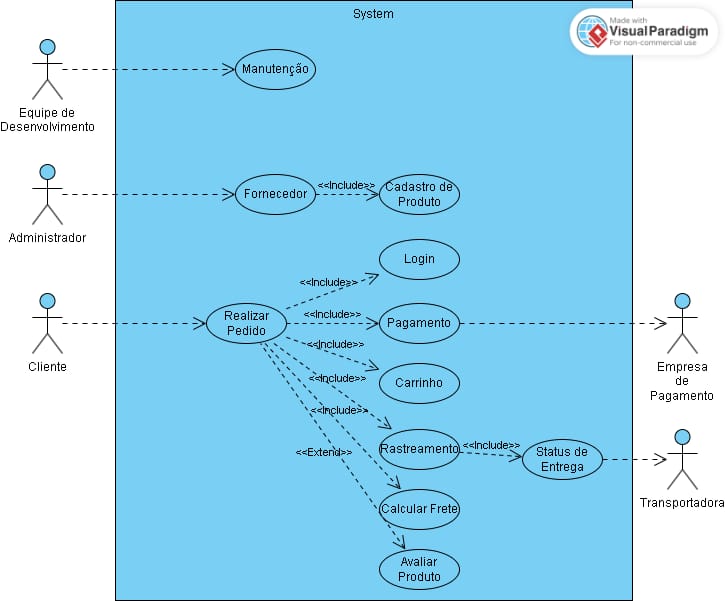
* Editor de Código: O **Visual Studio Code (VS Code)** é um editor de código-fonte gratuito e de código aberto, **criado pela Microsoft**. Ele foi lançado em 2015, marcando uma grande mudança na estratégia da empresa em direção a ferramentas de desenvolvimento mais leves e multi-plataforma. Antes dele, a Microsoft era conhecida principalmente por seu ambiente de desenvolvimento integrado (IDE) pesado, o Visual Studio, que era usado principalmente para o Windows. O VS Code foi projetado para ser uma alternativa rápida e flexível, capaz de competir com editores populares da época, como Sublime Text e Atom, e se tornou rapidamente o padrão da indústria.
* **Para que é utilizado?** O VS Code é utilizado para escrever, depurar e testar código. Ele oferece funcionalidades como o IntelliSense (preenchimento automático inteligente), depuradores integrados e suporte nativo para Git, facilitando o fluxo de trabalho do desenvolvedor.
* **Por que foi a escolha do projeto?** A decisão de usá-lo foi baseada na sua vasta biblioteca de extensões e na facilidade de colaboração, que permitiu uma organização impecável do código e um ambiente de trabalho unificado para todos os membros da equipe.
* **Frontend (Interface do Usuário):**
  + **HTML** (HyperText Markup Language): Criada por Tim Berners-Lee em 1991, enquanto trabalhava no CERN (Organização Europeia para a Pesquisa Nuclear), a linguagem foi a pedra fundamental para a criação da World Wide Web. Berners-Lee idealizou o HTML para permitir que pesquisadores de diferentes instituições pudessem compartilhar e vincular documentos de forma fácil e padronizada. Seu objetivo era criar um sistema de hipertexto global para a troca de informações.
  + **Para que é utilizado?** O **HTML (HyperText Markup Language)** é utilizado para dar estrutura ao conteúdo de uma página web. Ele atua como o esqueleto do site, definindo a hierarquia e o posicionamento de todos os elementos, como textos, imagens, botões e formulários.
  + **Por que foi a escolha do projeto?** Como a linguagem padrão da web, o HTML foi fundamental para construir a base do e-commerce. Ele garantiu que a estrutura do site fosse sólida e semântica, o que é crucial para a acessibilidade e para o bom funcionamento da loja.
  + **CSS** (Cascading Style Sheets): Desenvolvido em 1996, com Håkon Wium Lie e Bert Bos como principais autores no âmbito do W3C (World Wide Web Consortium), o CSS foi criado para resolver um problema crescente na web primitiva. Naquela época, a formatação e o estilo de uma página eram definidos diretamente dentro do HTML, o que tornava o código repetitivo, difícil de manter e visualmente limitado. O CSS nasceu para separar a "estrutura" da página (HTML) da sua "apresentação" visual, permitindo um controle muito maior sobre o design.
  + **Para que é utilizado?** O **CSS (Cascading Style Sheets)** é a linguagem de estilização que transforma a estrutura do HTML em um design visualmente atraente. Ele é utilizado para controlar cores, fontes, layouts, espaçamento e a responsividade da página.
  + **Por que foi a escolha do projeto?** O CSS foi essencial para criar uma interface de usuário agradável e intuitiva. Ele permitiu que o design do site fosse responsivo, adaptando-se a diferentes tamanhos de tela (desktop, tablet e celular), garantindo uma experiência de compra consistente para todos os usuários.
  + **JavaScript (JS):** Criado por **Brendan Eich** em 1995 na **Netscape Communications**, o JavaScript foi desenvolvido em apenas 10 dias com a missão de trazer interatividade para o navegador Netscape Navigator. Antes do JavaScript, as páginas da web eram estáticas. A linguagem, originalmente chamada de "Mocha", permitiu que desenvolvedores adicionassem comportamentos dinâmicos, como a validação de formulários e animações, diretamente no navegador do usuário, revolucionando a experiência de navegação.
  + **Para que é utilizado?** O **JavaScript (JS)** é uma linguagem de programação utilizada para adicionar interatividade e dinamismo a uma página web. Ele permite a criação de funcionalidades avançadas, como validação de dados, manipulação de elementos da interface e comunicação com o servidor.
  + **Por que foi a escolha do projeto?** A escolha do **JavaScript** foi crucial para a lógica de negócio do projeto. Ele é o responsável por gerenciar as operações do carrinho de compras, processar os dados dos formulários e interagir com o banco de dados, tornando o site uma plataforma funcional e dinâmica.
* Banco de Dados: **Node.js (Backend):** O Node.js é um ambiente de execução de JavaScript no lado do servidor, **criado por Ryan Dahl** em 2009. Dahl desenvolveu-o por frustração com as limitações dos servidores web tradicionais em lidar com múltiplas conexões simultâneas de forma eficiente. Ele utilizou o motor V8 do Google Chrome para permitir que o JavaScript fosse executado fora do navegador, dando vida a uma nova era de aplicativos de backend que são incrivelmente rápidos e escaláveis.
* **Para que é utilizado?** O **Node.js** é um ambiente de execução que permite que o JavaScript seja usado no lado do servidor (backend). Ele é utilizado para construir APIs, servidores web e aplicações de tempo real, lidando com a lógica de negócio e a comunicação com o banco de dados.
* **Por que foi a escolha do projeto?** Utilizar o **Node.js** permitiu que o projeto tivesse uma linguagem unificada (**JavaScript**) tanto no frontend quanto no backend. Essa consistência simplificou o desenvolvimento, o que facilitou a manutenção do código e permitiu uma implementação mais rápida e eficiente de novas funcionalidades.

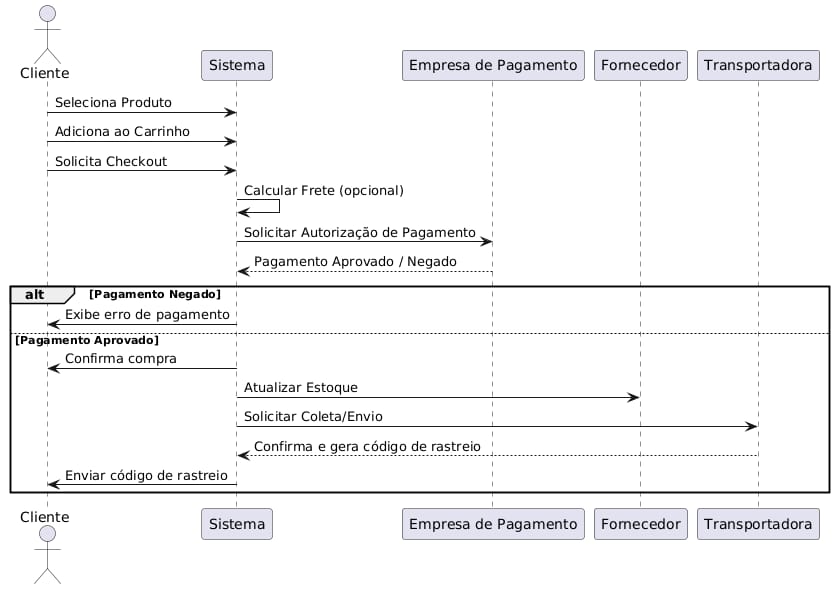
Essa abordagem técnica permitiu a criação de uma solução funcional e robusta, que não apenas resolveu as falhas identificadas, mas também preparou o negócio para futuras expansões no mercado digital.

### 8. Model Canva

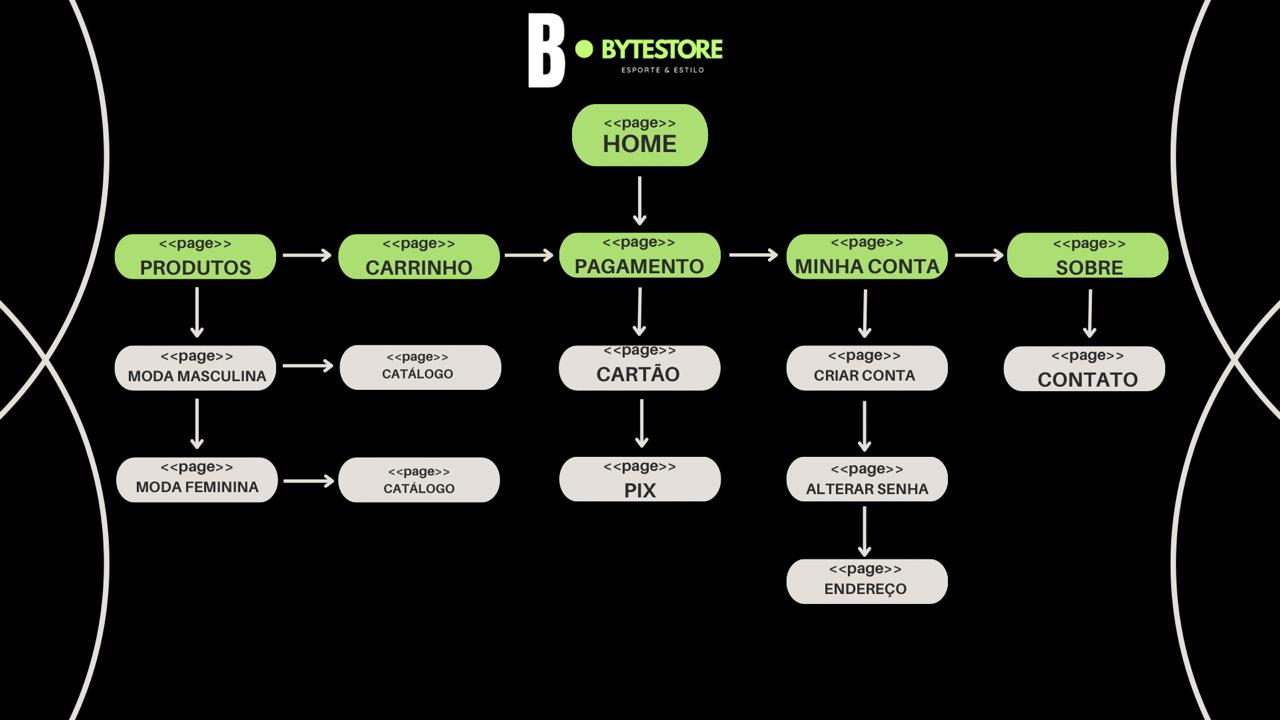


### 9. Diagramas

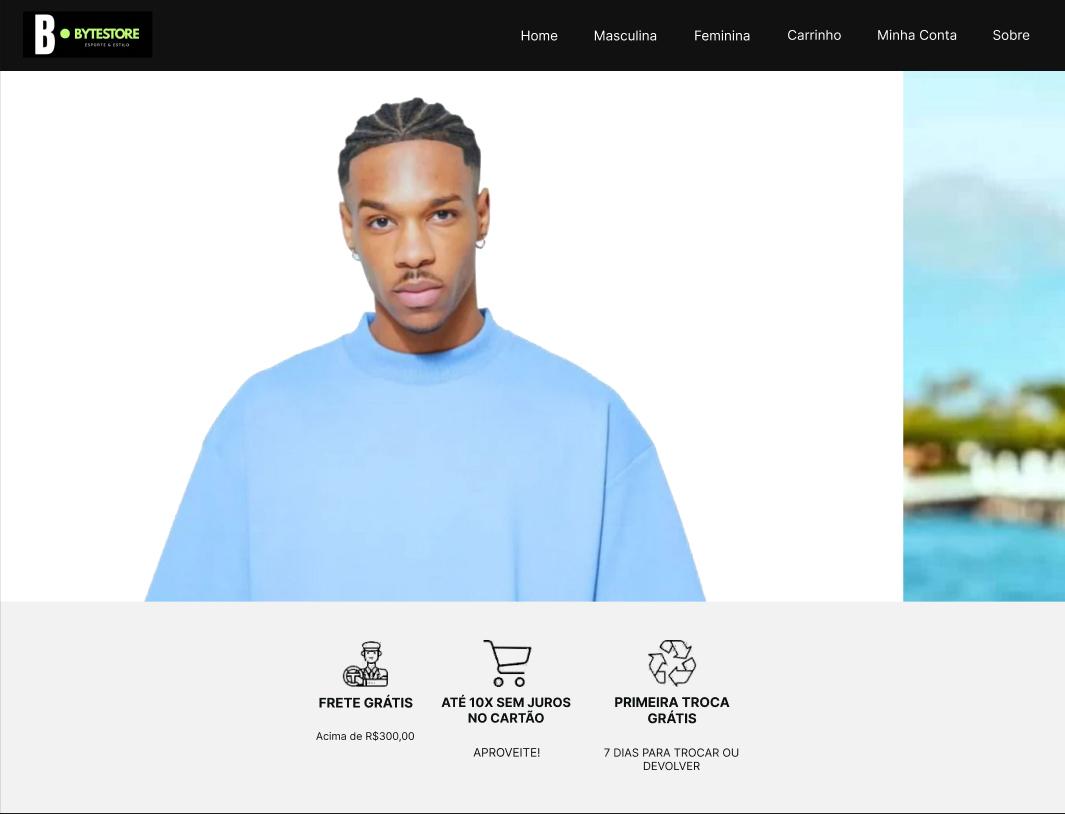




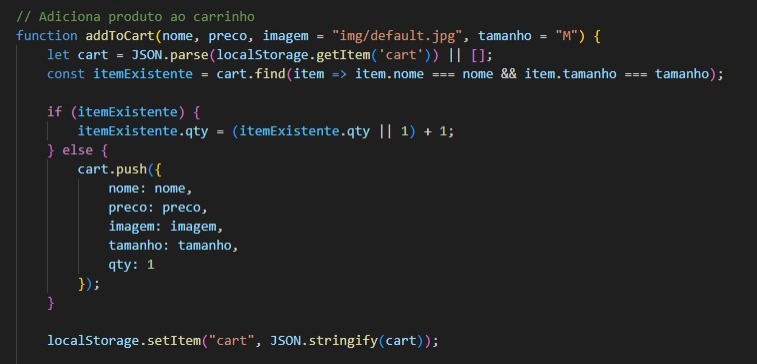
### 10.Mapa de Navegação



### 11. Prototipagem



### 12. Trecho do Código (função adicionar ao carrinho)



### Explicação:

* **Busca no carrinho** se o item já foi adicionado.
* Se já existe, **aumenta a quantidade**.
* Adiciona itens no carrinho.
* Se não existe, **cria um novo registro** com nome, preço, imagem, tamanho e quantidade.
* Usa localStorage para manter os dados salvos.

### 13. Conclusão

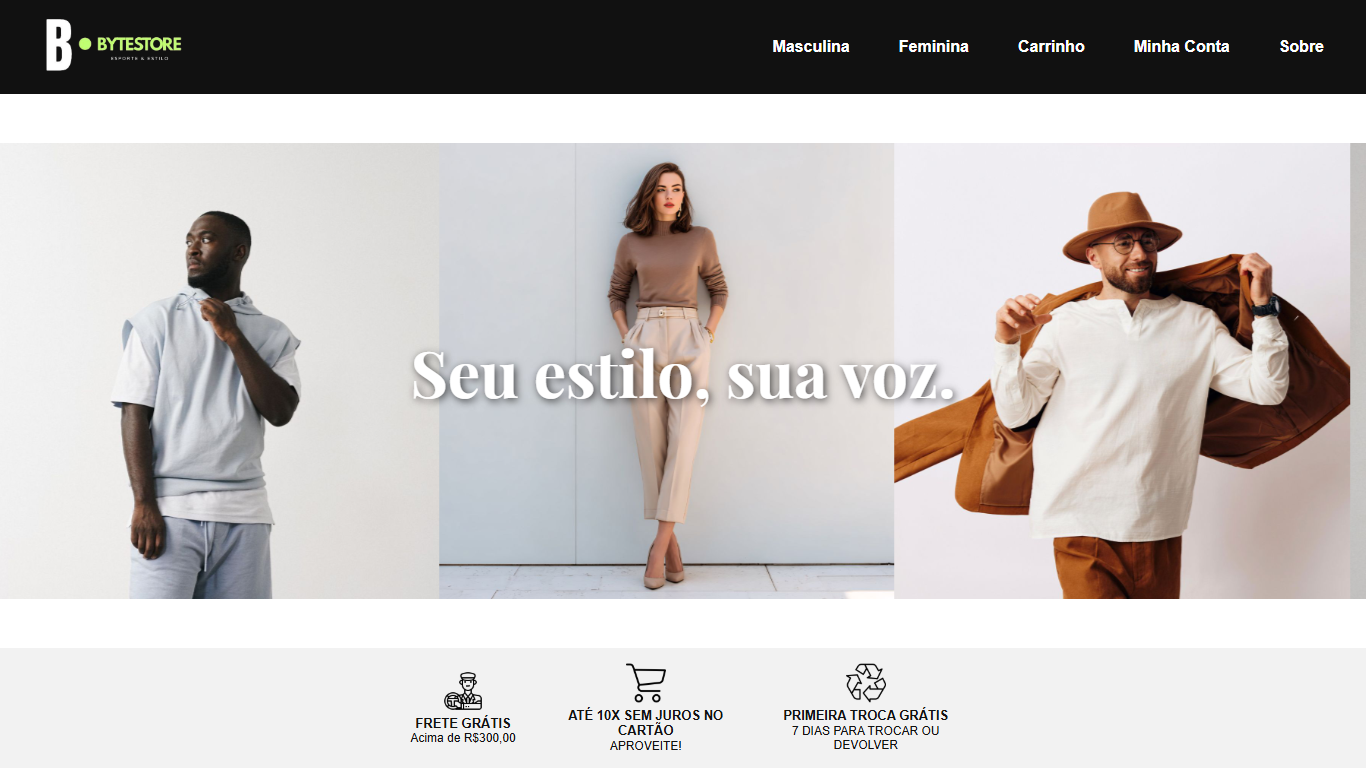
O projeto ByteStore chegou ao fim e o resultado é uma prova da nossa determinação. Embora tenhamos enfrentado um grande desafio com a redução do nosso time, conseguimos nos reorganizar e manter o foco nos objetivos principais.

O sucesso na entrega do projeto, com quase todas as funcionalidades implementadas, valida o nosso planejamento e a nossa capacidade de superar imprevistos. A experiência nos ensinou o valor da flexibilidade e da colaboração, habilidades que foram essenciais para garantir que o ByteStore se tornasse uma realidade funcional e pronta para atender a sua proposta de valor.

### 14. Prints Do Projeto

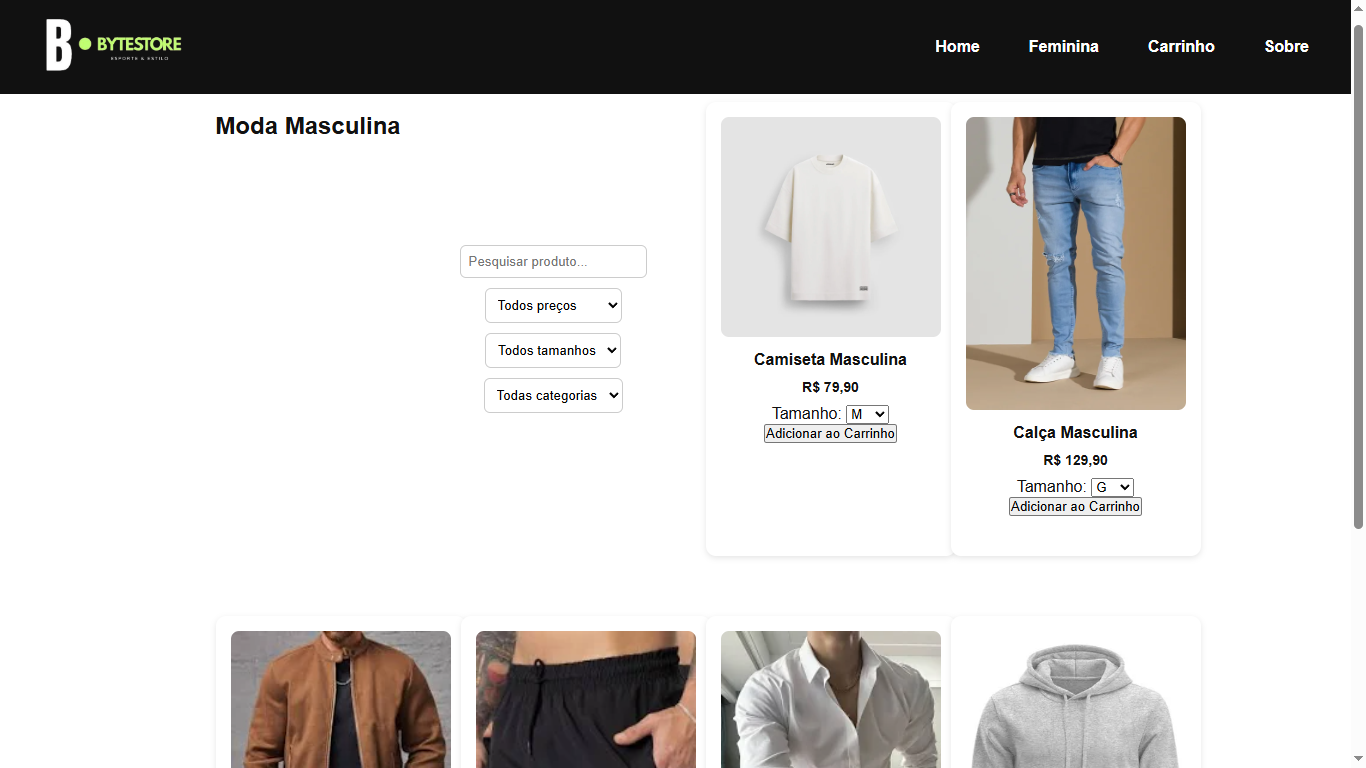
Página “Home”

Página inicial do site, com destaque visual para a loja. Possui banners rotativos (slideshow) para chamar atenção de promoções e categorias, além do menu de navegação que direciona para as demais seções.



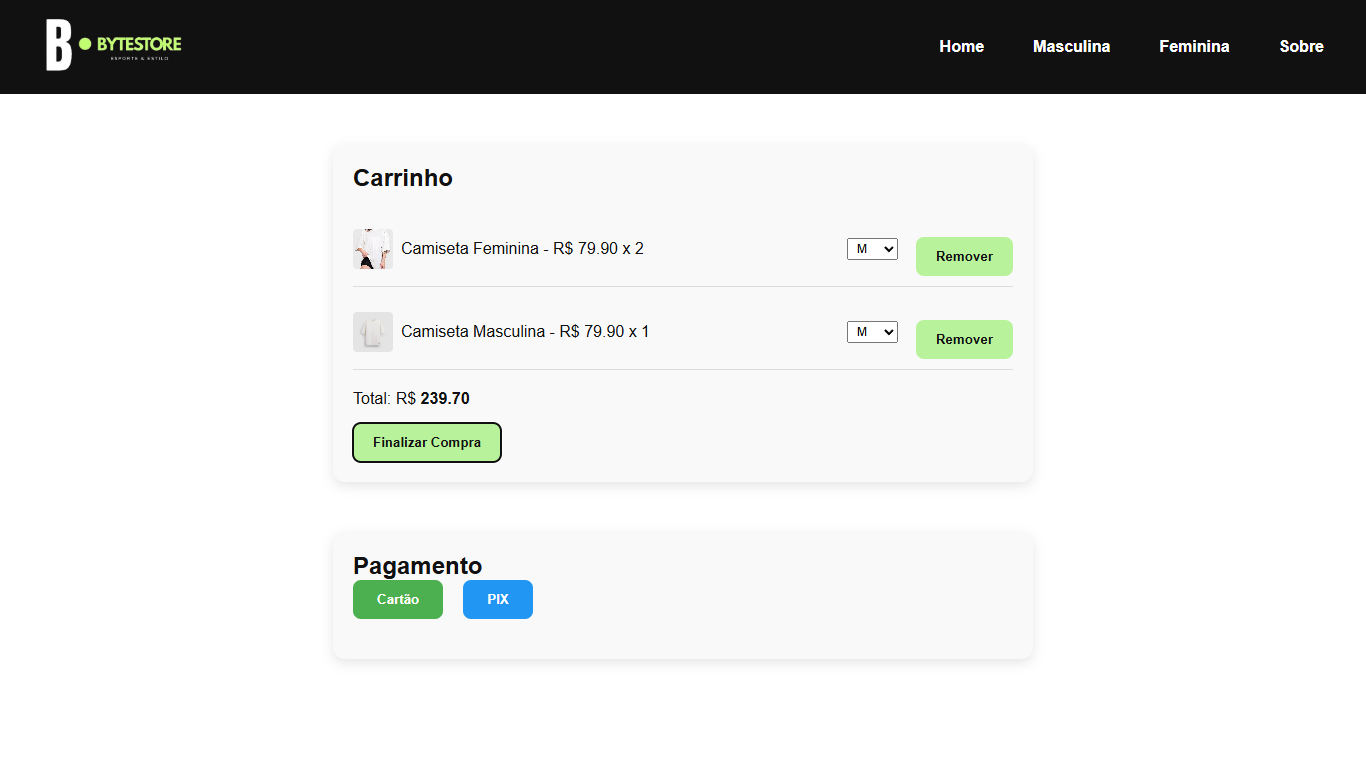
Página “Masculina”

Página dedicada aos produtos masculinos, organizada em cartões com imagem, preço, tamanho e botão para adicionar ao carrinho. Conta também com filtros de pesquisa por preço, tamanho e categoria, facilitando a busca.

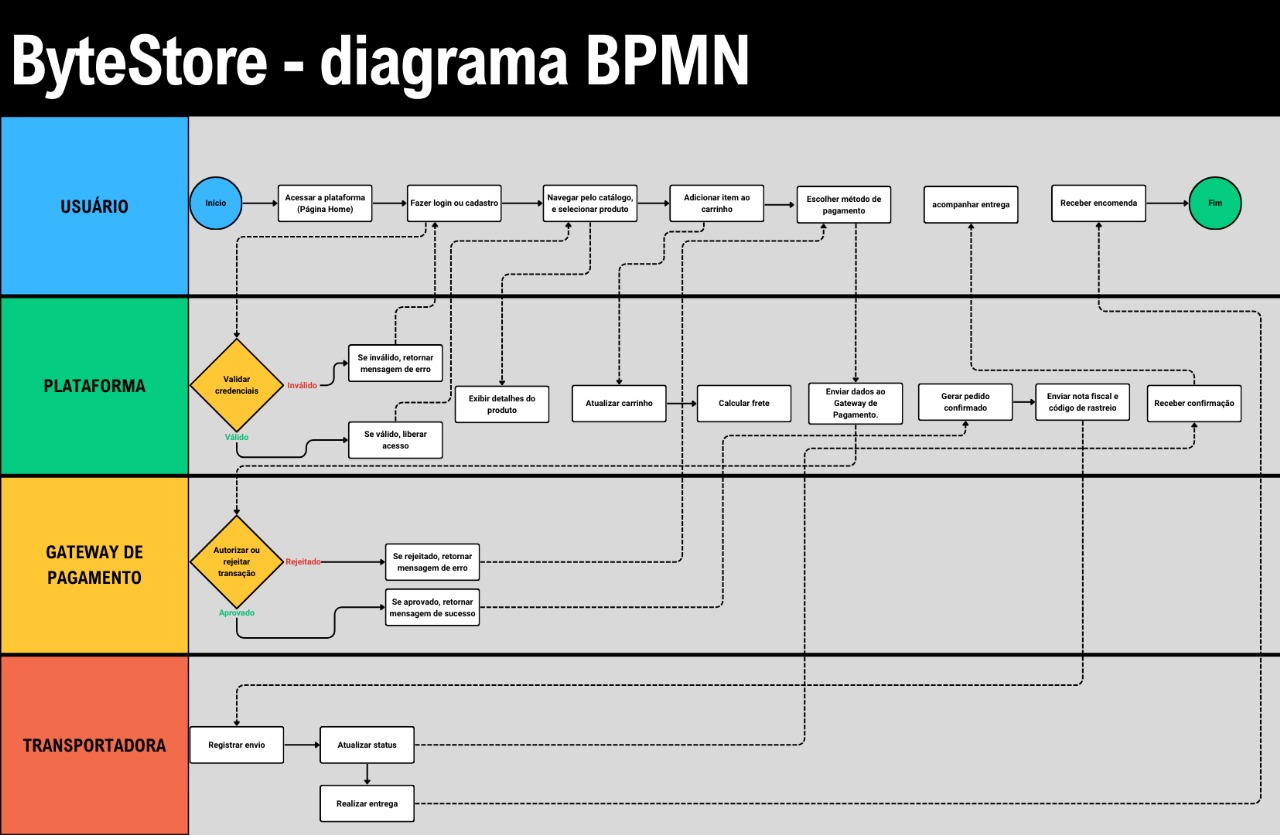


Página “Carrinho”

Página onde o cliente visualiza os itens selecionados, podendo alterar quantidade, escolher tamanho e remover produtos. Mostra também o valor total e permite prosseguir para as opções de pagamento (cartão ou PIX).



### 15. BPMN



### 16.Cronograma



### 17.Referências

### Comércio Eletrônico

Referência: AMORIM, Krislaine Silva. E-COMMERCE: UM ESTUDO BIBLIOGRÁFICO SOBRE A IMPORTÂNCIA DAS VENDAS ONLINE PARA OS MICROEMPREENDEDORES. Revista FT, v. 27, n. 128, nov. 2023.

* Link de acesso: Disponível em: <https://revistaft.com.br/e-commerce-um-estudo-bibliografico-sobre-a-importancia-das-vendas-online-para-os-microempreendedores/>.

### Sistemas de Informação Gerenciais

Referência: RODRIGUES, F. G. M.; MEIRINO, J. C.; PAULA, K. G. Sistema de informações gerenciais como ferramenta para a gestão de Recursos Humanos. RH Visão Sustentável, v. 5, n. 1, p. 119-140, 2018.

* Link de acesso: Disponível em: <https://revistas.cesgranrio.org.br/index.php/rh_visaosustentavel/article/view/3262>.

### Gestão da Inovação

**REFERÊNCIA 1:**

ALVES, B. N. et al. A gestão da inovação como prática:             Contribuições do conceito de gestão ordinária. REGEPE –         Revista de Empreendedorismo e Gestão de Pequenas                Empresas, v. 10, n. 1, p. 1862, jan./abr. 2021.

* + Link de acesso: Disponível em: <https://dialnet.unirioja.es/descarga/articulo/7929435.pdf>.

**REFERÊNCIA 2:**

PEREIRA, José Ismael et al. MODELOS DE GESTÃO DE INOVAÇÃO: CRITÉRIOS E CARACTERÍSTICAS UTILIZADOS EM REVISÕES DE LITERATURA. SIMPOI 2021, [S. l.], 2021.

* + Link de acesso: Disponível em: <https://anpad.com.br/uploads/articles/112/approved/c3992e9a68c5ae12bd18488bc579b30d.pdf>.

**PMBOK:**

* [https://www.pmi.org/standards/pmbok](https://pmkb.com.br/sig/padroes-frameworks/pmbok-pmi/)