

**Documentação de um**

**Produto de Software**

**Falcommerce**

**Nome dos Alunos:**

|  |  |
| --- | --- |
| **RA** | **Nome** |
| **21.00867-0** | **Lucca Leça Aggio** |
| **22.01243-5** | **Joao Pedro Crepaldi Fukumura** |
| **22.00908-6** | **Andrews Soares Frasson** |
|  |  |
|  |  |

**2024**

**ÍNDICE DETALHADO**

[1. Descrição/Resumo do Projeto 3](#_Toc164674371)

[2. Tecnologias Empregadas 3](#_Toc164674372)

[3. Definição de Papéis no Projeto 3](#_Toc164674373)

[4. Levantamento dos Requisitos do Sistema de Software – Product Backlog 3](#_Toc164674374)

[5. Sprint 1 4](#_Toc164674375)

[5.1. Planejamento da Sprint 1 – Sprint Backlog 4](#_Toc164674376)

[5.2. Implementação e Testes – Sprint 1 4](#_Toc164674377)

[5.3. Review da Sprint 1 4](#_Toc164674378)

[5.4. Retrospectiva da Sprint 1 4](#_Toc164674379)

[6. Sprint 2 5](#_Toc164674380)

[7. Registro da Apresentação ao Parceiro 6](#_Toc164674381)

[Referências 7](#_Toc164674382)

[Apêndice I 8](#_Toc164674383)

# Descrição/Resumo do Projeto

|  |
| --- |
| As empresas têm aumentado sua presença no mercado digital de forma acelerada nos últimos anos. Consequentemente, o volume de informações cresceu exponencialmente, e muitas empresas de pequeno e médio porte não conseguiram se adaptar a essa nova realidade, enfrentando dificuldades na gestão e valorização desses dados. A Falcommerce surge para resolver essa necessidade com a seguinte premissa: facilitar a visualização de dados, melhorar a integração entre setores, organizar informações e fornecer ferramentas personalizadas conforme a necessidade do cliente por meio de softwares e soluções digitais.  Oferecendo um catálogo de ferramentas flexíveis, onde os clientes podem escolher entre diversas opções de funcionalidades, organizadas em planos que se adaptam à necessidade de cada negócio. Além disso, oferecemos a possibilidade de desenvolver ferramentas personalizadas, proporcionando uma solução sob medida para os desafios específicos de cada cliente. Essa abordagem não apenas entrega funcionalidades prontas, mas também assume toda a parte técnica, permitindo que os clientes foquem exclusivamente no crescimento de seu negócio.  Ao contrário de outras soluções, a Falcommerce opera como um software house especializado em desenvolver e fornecer ferramentas digitais personalizadas a partir de planos feitos sobre a premissa de demandas. Não apenas oferecendo funcionalidades prontas, mas também assumindo toda a parte técnica, permitindo que os clientes se concentrem exclusivamente no crescimento de seu negócio. A partir de planos flexíveis, que se adaptam às necessidades específicas, o cliente recebe um software que, conforme suas exigências, pode fornecer mais ou menos funcionalidades, padronizadas ou personalizadas, para enfrentar as adversidades do mundo corporativo.  Nosso objetivo é capacitar pequenas e médias empresas a gerenciar e otimizar suas operações digitais de maneira eficaz, aumentando sua competitividade no mercado e facilitando seu crescimento sustentável. Com uma abordagem focada na simplicidade, eficiência e personalização, a Falcommerce se posiciona como a solução ideal para empresas que desejam maximizar seu potencial no mercado digital sem se sobrecarregar com a complexidade do mercado de tecnologia. |

# Tecnologias Empregadas

## 2.1- Ferramentas de Prototipação e Design Visual

**Figma**:

**Motivo**: O Figma foi escolhido para a prototipação de telas e design de interface. Suas funcionalidades de colaboração em tempo real são essenciais para o desenvolvimento do layout do site, permitindo ajustes rápidos e interações dinâmicas. Além disso, os plugins facilitam a criação de componentes reutilizáveis e layouts responsivos.

## 2.2- Linguagens de Programação

Next.js (Frontend):

Motivo: O Next.js é um framework JavaScript amplamente utilizado para a construção de interfaces de usuário dinâmicas e responsivas. Ele facilita o desenvolvimento frontend com recursos como renderização híbrida (tanto lado do cliente como do servidor) , o que otimiza a performance do site ao carregar as páginas mais rapidamente,. A possibilidade de criar componentes reutilizáveis torna o desenvolvimento frontend mais ágil e organizado. Outra vantagem é sua facilidade de integração com APIs e microserviços, que simplifica a conexão com dados externos e permite a criação de sites rápidos e otimizados para a web.

.

JavaScript/Node.js (Backend):

Motivo: O Node.js foi escolhido para o backend devido a maior experiência do desenvolvedor e capacidade da linguagem de lidar com múltiplas conexões simultâneas de maneira eficiente. Além disso, o Node.js utiliza JavaScript, facilitando a manutenção e o desenvolvimento por já utilizarmos no front-end.

**2.3- SGBD (sistema gerenciador de Banco de Dados)**

MySQL na AWS:

Motivo**:** Optamos pelo MySQL na AWS por ser uma solução em nuvem gratuita que oferece diversas funcionalidades essenciais para o nosso projeto. A AWS proporciona escalabilidade, alta disponibilidade e segurança, facilitando o gerenciamento do banco de dados e permitindo um maior foco no desenvolvimento da aplicação.

**2.4- Ferramenta de gerenciamento de banco de dados (cliente de banco de dados)**

DBeaver:

Motivo: O cliente de banco de dados facilita o trabalho com o SGBD ao fornecer uma interface amigável e ferramentas adicionais para o gerenciamento do banco de dados, tornando o processo de administração, desenvolvimento, análise de dados e testes mais acessíveis e eficiente.

**2.5- Ferramentas de Desenvolvimento e Gerenciamento**

Git e GitHub:

Motivo: O Git é uma ferramenta de controle de versão distribuído que permite colaboração eficiente entre desenvolvedores. O GitHub será utilizado como a plataforma para hospedar o repositório do projeto, facilitando a visualização de mudanças no código, a gestão de branches e a integração contínua, garantindo um desenvolvimento organizado e seguro.

Jira:

Motivo: O Jira é uma ferramenta de gerenciamento de projetos que será utilizada para o controle do backlog, sprints e tarefas. Com ele, a equipe pode acompanhar o progresso do projeto, priorizar tarefas e garantir que todas as funcionalidades sejam entregues conforme o planejado.

2.4. Testes e Qualidade

Jest (Testes Unitários):

Justificativa: O Jest é uma ferramenta de testes em JavaScript que será usada para garantir a qualidade do código. Ele permite a execução de testes unitários e de integração de forma simples e eficiente, garantindo que cada funcionalidade desenvolvida esteja funcionando corretamente antes de ser integrada ao sistema.

Cypress (Testes End-to-End):

Motivo: O Cypress é uma ferramenta de testes end-to-end que será utilizada para simular o comportamento do usuário e verificar se o sistema funciona como esperado em ambiente real. Isso assegura que todas as funcionalidades do Falcommerce estejam viáveis e livres de bugs. Ele tem um papel de simulador de cenário no mundo real, onde ele vai testar a aplicação como se fosse um usuário final.

# Definição de Papéis no Projeto

**Product Owner - Andrews Soares Frasson**

O Product Owner é o responsável por decidir o que é mais importante para o projeto. Ele vai garantir que as prioridades estejam bem definidas e que a equipe esteja sempre focada no que tem mais valor para o cliente. Também será responsável por manter o **backlog**, que é basicamente uma lista de tudo que precisa ser feito, organizada por ordem de prioridade. Além disso, vai garantir que todos entendam claramente o que precisa ser entregue a cada **sprint** (que são ciclos de desenvolvimento de algumas semanas).

**Scrum Master - João Pedro Crepaldi Fukumura**

O Scrum Master vai garantir que as práticas ágeis ocorram corretamente. Também vai organizar as seguintes reuniões:

**Planejamento das Sprints**: Onde a vai ser definido o que vai acontecer em uma sprint específica.

**Reuniões diárias**: Reuniões só pra cada um dizer como está o progresso do produto.

**Retrospectiva**: Verifica o que foi feita na sprint que já aconteceu e faz melhorias para as próximas.

**Desenvolvedor – Lucca Leça Aggio**

O desenvolvedor será encarregado de fazer o código funcionar. Ele vai implementar as funcionalidades que foram definidas no backlog, sempre fazendo testes e corrigindo qualquer bug que aparecer. A ideia é garantir que o produto final esteja funcionando e atenda às expectativas do cliente. Além disso, ele vai participar das reuniões e ajudar a resolver qualquer problema técnico que surgir, colaborando com a equipe para que tudo seja entregue dentro do prazo e com qualidade.

# Levantamento dos Requisitos do Sistema de Software – Product Backlog

Este capítulo tem como objetivo apresentar o levantamento dos requisitos do Sistema de Software e a especificação de requisitos.

Deve-se criar um Product Backlog, de acordo com as reuniões realizadas com o parceiro. O Product Backlog deve conter:

* Cartões de histórias que representam os requisitos do software.
* Prioridade de cada história.
* Estimativa de esforço.
* Sprint em que será implementada.

Esse Product Backlog deve ser criado em alguma ferramenta, como: Azure, Trello ou Jira.

# Sprint 1

Este capítulo tem como objetivo apresentar o planejamento, desenvolvimento, testes, entrega e retrospectiva da Sprint 1.

**Esse capítulo deve se repetir com todos os itens para as demais Sprints que serão desenvolvidas no projeto. Exemplo: Capítulo 4. Sprint 2 e assim por diante.**

## Planejamento da Sprint 1 – Sprint Backlog

Neste item o objetivo é apresentar um detalhamento de como será implementada a Sprint, definindo as regras de negócio, proposta de arquitetura, solução técnica e implementação. Criar um quadro de tarefas detalhado de cada Sprint. Esse quadro de tarefas deve ser criado em alguma ferramenta, como: Azure, Trello ou Jira.

## Implementação e Testes – Sprint 1

Neste item indicar o link do repositório com o código e os testes implementados na Sprint 1.

Para atender as disciplinas “CIC203 - Algoritmos, Estrutura de Dados e Programação” e “SIN202 - Estrutura de Dados”, os alunos, OBRIGATORIAMENTE, devem implementar/codificar em seus projetos no mínimo um algoritmo ou uma estrutura de dados dentre as/os estudadas/os no semestre.

Os testes a serem realizados são de BDD (*Behavior Driven Development*) e TDD (*Test Driven Development*).

Explicitar neste documento os testes de BDD para os principais cartões de história desta Sprint. Os testes de TDD devem estar no repositório e ter evidências de que foram implementados e utilizados para testes.

## Review da Sprint 1

Descrever o evento de Review da Sprint. Realizar um breve relato dos resultados desta Sprint, o objetivo foi cumprido? E o que será realizado na próxima Sprint. Se possível descrever o parecer do parceiro testando essa Sprint.

## Retrospectiva da Sprint 1

Descrever o evento de Retrospectiva da Sprint. Realizar um breve relato de como foi a Sprint em relação a indivíduos, interações, pontos fontes e fracos e o que deve ser mantido e o que deve ser melhorado para a próxima Sprint.

# Sprint 2

Repetir para a Sprint 2.

# Registro da Apresentação ao Parceiro

Neste item devem ser apresentados os registros firmados com os parceiros do projeto.

Referências

Neste item incluir as referências bibliográficas que forem utilizadas na documentação, seguindo o padrão ABNT.

Apêndice I

Neste item deve ser anexado registro das reuniões com o parceiro, além de fotos, documentos de apoio e qualquer material relevante ou que comprove a interação com o parceiro.