**Soluções Computacionais**

**<Falcommerce>**

|  |  |
| --- | --- |
| **Nome Completo** | **RA** |
| Andrews Soares Frasson | 22.00908-6 |
| Joao Pedro Crepaldi Fukumura | 22.01243-5 |
| lucca leça aggio | 21.00867-0 |
|  |  |
|  |  |
|  |  |

**2024**

1. **Introdução**  
   A Falcommerce foi criada a partir de um cenário onde pequenos e médios empreendedores enfrentam desafios significativos para usufruir das ferramentas avançadas que o mundo da programação pode oferecer. Além disso, muitos desses empreendedores dependem de programadores que frequentemente cobram altos valores por seus serviços e manutenção. Apesar do constante crescimento do mercado de e-commerce, muitas soluções existentes são ou muito simples e limitadas, ou extremamente complexas e caras, criando uma lacuna para negócios que buscam uma solução intermediária, flexível e escalável.

Atualmente, empreendedores enfrentam altos custos de desenvolvimento e manutenção de plataformas personalizadas, falta de flexibilidade em soluções pré-prontas, dificuldades em escalar operações e complexidade na integração de diferentes ferramentas essenciais para uma operação eficiente. Pesquisas de mercado e feedback de potenciais clientes indicam uma demanda crescente por soluções personalizáveis e de fácil implementação, que ofereçam escalabilidade sem grandes investimentos em infraestrutura, integração simplificada de diversas funcionalidades, suporte técnico contínuo e flexibilidade para evolução do negócio.

Para atender essas necessidades, a Falcommerce estabeleceu objetivos técnicos e de negócio claros. No âmbito técnico, o foco está no desenvolvimento de um site web funcional que servirá como ponto de entrada e apresentação do conceito. Isso inclui a implementação de páginas essenciais, uma interface intuitiva e responsiva, e o desenvolvimento de pelo menos uma funcionalidade de exemplo que demonstre o potencial da plataforma. A arquitetura do sistema será projetada para ser escalável, permitindo fácil adição de novas funcionalidades no futuro.

Sobre os objetivos de negócio, a Falcommerce planeja realizar uma pesquisa de mercado aprofundada para identificar as necessidades mais urgentes dos pequenos e médios e-commerces. Com base nisso, será desenvolvido um catálogo conceitual de funcionalidades potenciais, categorizadas por departamentos. Um plano de desenvolvimento futuro será criado, priorizando as funcionalidades mais demandadas pelo mercado. Métricas de engajamento serão estabelecidas para avaliar o interesse do público-alvo, e uma apresentação convincente do projeto será preparada, destacando seu potencial de mercado e visão de longo prazo.

Além disso, a Falcommerce pretende desenvolver um plano para coletar feedback dos usuários, explorar parcerias acadêmicas e com empreendedores locais para validação do conceito, criar um roadmap inicial para o desenvolvimento futuro, e estabelecer um plano de pesquisa contínua para manter-se atualizada sobre as tendências do mercado. Uma estratégia inicial de marketing digital também será desenvolvida para promover o conceito e atrair potenciais usuários interessados em acompanhar o desenvolvimento do projeto.

Com essa abordagem abrangente, a Falcommerce visa não apenas criar uma solução tecnológica, mas também estabelecer uma base sólida para um empreendimento que possa evoluir e se adaptar às necessidades dinâmicas do mercado de e-commerce, oferecendo valor real para pequenos e médios empreendedores.

1. **Plano de Gerenciamento de Projetos**
   1. **Ciclo de Vida Abordado**

Para o desenvolvimento da Falcommerce, optamos por uma abordagem híbrida, combinando elementos dos métodos ágil e preditivo. Inicialmente, será utilizado um planejamento preditivo para definir os requisitos e a arquitetura do sistema, estabelecendo marcos e prazos para os entregáveis chave. Em seguida, o desenvolvimento será conduzido de forma ágil, com sprints de 2-3 semanas. Essa abordagem oferece flexibilidade com uma direção clara, entrega de valor antecipada, gerenciamento eficaz de riscos e envolvimento contínuo das partes interessadas, resultando em um produto final de alta qualidade. Optamos por uma abordagem híbrida, combinando métodos preditivos e ágeis, para a criação do site da Falcommerce devido à necessidade de equilibrar planejamento estruturado e flexibilidade. O método preditivo nos permite definir claramente os requisitos iniciais, a arquitetura do sistema e estabelecer marcos essenciais, garantindo uma visão clara e objetivos bem delineados desde o início. Embora possa haver desafios na gestão da complexidade e no equilíbrio entre documentação, a escolha se justifica por causa da natureza dinâmica do mercado de e-commerce, as expectativas das partes interessadas, e a necessidade de um escopo inicial bem definido com capacidade de expansão futura.

* 1. **Principais requisitos das principais entregas/produtos**

Estruturar a documentação de requisitos em uma tabela para facilitar a consulta. Adicionar uma categorização entre requisitos obrigatórios e opcionais, ou de alta, média e baixa prioridade

* 1. **Estrutura Analítica do Projeto – Fases e principais entregas**

A EAP deve incluir uma descrição detalhada de cada fase do projeto e suas entregas, com prazos definidos. Se possível, incluir ferramentas visuais para apresentar a EAP.

* 1. **Marcos**

Adicionar uma descrição clara de como cada marco será avaliado e os critérios de sucesso para a entrega da fase.

* 1. **Partes interessadas do projeto**

Incluir um gráfico RACI (Responsibility, Accountability, Consulted, Informed) para descrever melhor as funções e responsabilidades de cada parte interessada, garantindo clareza nos papéis e na comunicação.

* 1. **Restrições**

Relacione as restrições do projeto, ou seja, limitação aplicável ao projeto, a qual afetará seu desempenho. Limitações reais: orçamento, recursos, tempo de alocação, ... Ex.: Entrega no dia 30/11/2024.

* 1. **Premissas**

Relacione as premissas do projeto, ou seja, fatores considerados verdadeiros sem prova para fins de planejamento. Ex.: Disponibilidade de 50% do tempo do cliente durante os testes.

* 1. **Formas de Comunicação**

Identifique a forma de comunicação e periodicidade com as partes interessada. Pode-se utilizar o modelo de matriz utilizado na aula de Gestão de Projetos.

* 1. **Riscos**

Identifique os riscos do projeto, categoria e ações. Pode-se utilizar o modelo de matriz de risco utilizado na aula de Gestão de Projetos.

* 1. **Cronograma e etapas do projeto.**

Incorporar cronograma com fases detalhadas e prazos realistas. Adicione também recursos alocados para cada fase, destacando dependências entre as tarefas. O cronograma deve estar de acordo com o modelo de processo escolhido.

|  |
| --- |
| ***Observações***  ***OBS 1: Caso o item ciclo de vida abordado seja desenvolvimento ágil realize o preenchimento da documentação de software a partir do item 5 e remova os itens de 3 a 8 (AZUL NO TEXTO).***  ***OBS 2: Caso o item ciclo de vida abordado seja cascata realize o preenchimento da documentação de software a partir do item 3 e remova o item 9 A 11 (VERDE NO TEXTO).***  **O item 10 deve ser elaborado para qualquer ciclo de vida.** |

1. **Requisitos do Sistema**
   1. **Extração de Requisitos**

Especifique como os requisitos foram coletados, caso tenha sido entrevista ou questionário, inclua as evidências no apêndice.

* 1. **Requisitos Funcionais**

Descreva as funcionalidades que o software deve ter, de acordo, com a extração de requisitos.

* 1. **Requisitos Não Funcionais**

Descreva os requisitos não funcionais, com as características de qualidade (performance, usabilidade, segurança, etc.).

1. **Modelagem do Sistema**
   1. **Diagrama de Casos de Uso**

Represente dos requisitos funcionais, por meio do diagrama de caso de uso e se necessário as especificações detalhadas de cada ator e caso de uso.

* 1. **Diagrama de Classes**

Elabore a estrutura de solução, por meio do diagrama de classes.

* 1. **Outros Diagramas UML**

Elabore diagramas complementares se necessário para uma melhor modelagem, e proposta de solução. Exemplos de outros diagramas: sequência, atividades, componentes, estados etc.

1. **Tecnologias e Ferramentas Utilizadas para Desenvolvimento**

Descreva a arquitetura, tecnologias utilizadas para a implementação, banco de dados e ambiente de implantação.

1. **Design de Interface**

Descreva as características de experiência do usuário e como será essa interface do usuário, justificando cores, usabilidade, acessibilidade etc.

Elabore o protótipo das telas.

1. **Desenvolvimento e Implementação**

Indique o link do repositório com o código implementado.

1. **Testes**

Detalhe o plano de testes. Defina os tipos de testes (unitários, integração, aceitação, carga) e as ferramentas utilizadas. Especificar métricas de qualidade para a validação dos testes e indique o repositório com os testes

1. **Levantamento dos Requisitos do Sistema de Software – Product Backlog**

Crie um Product Backlog, de acordo com a extração de requisitos, caso tenha sido entrevista ou questionário, inclua as evidências no apêndice.

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.

1. **Tecnologias e Ferramentas Utilizadas para Desenvolvimento**

Descreva a arquitetura, tecnologias utilizadas para a implementação, banco de dados e ambiente de implantação.

1. **Sprint X**

Apresente o planejamento, desenvolvimento, testes, entrega e retrospectiva da Sprint 1. Lembre-se que você pode utilizar modelos da UML para detalhar a sprint, como: diagrama de classes, sequência, implantação etc.

Este item deve se repetir com todos os itens para as demais Sprints que serão desenvolvidas no projeto.

* + 1. **Planejamento da Sprint 1 – Sprint Backlog**

Apresente 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.

* + 1. **Design de Interface**

Descreva as características de experiência do usuário e como será essa interface do usuário, justificando cores, usabilidade, acessibilidade etc.

Elabore o protótipo das telas.

* + 1. **Implementação e Testes – Sprint 1**

Indique o link do repositório com o código e os testes implementados na Sprint 1.

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

* + 1. **Review da Sprint 1**

Descreva 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 descreva o parecer do parceiro testando essa Sprint.

* + 1. **Retrospectiva da Sprint 1**

Descreva 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.

1. **Conclusão**

Adicione uma análise mais completa dos resultados alcançados, comparando com os objetivos estabelecidos inicialmente. Incluir uma seção de 'Próximos Passos', indicando o que será feito após a conclusão do projeto.