Projeto Integrador 2º Semestre - DSM

**Disciplinas:**

Banco de Dados

Desenvolvimento Web II

Engenharia de Software II

**Professores:**

Bruno

Nilton

Orlando

**Grupo(2) / Datamind:**

**Sistema:**

|  |  |
| --- | --- |
| Integrante | **Papel Principal** |
| Gabriel Victorino | PO, Desenvolvedor Backend |
| Gustavo Francisco Habermann | Desenvolvedor Frontend |
| João Vitor de Camargo |  |
| Rafael Botezelli |  |
|  |  |

**Fatec Araras**

**2024**

FICHA DE CONTROLE - PROJETO INTERDISCIPLINAR

DISCIPLINA CHAVE: Engenharia de Software II - PI II

PROFESSOR: Bruno Henrique de Paula Ferreira

GRUPO: Nome do grupo SEMESTRE: 2/2024

TÍTULO DO PROJETO: título

DATA DA APRESENTAÇÃO: 03/12/2024

NOTA:

INTEGRANTES DO GRUPO: Nome grupo

|  |  |
| --- | --- |
| Nome | Nota Individual |
| Gabriel Victorino |  |
| Gustavo Francisco Habermann |  |
| João Vitor de Camargo |  |
| Rafael Botezelli |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

Araras, 03 de dezembro de 2024

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Professor Bruno Henrique de Paula Ferreira

**Sumário**

[1. Apresentação da Empresa 5](#_Toc176685201)

[1.1 Missão 5](#_Toc176685202)

[1.4 Link Repositório 6](#_Toc176685203)

[2. Escopo do sistema 7](#_Toc176685204)

[2.1 Objetivos do projeto 7](#_Toc176685205)

[2.2 Técnica de levantamento de requisitos 7](#_Toc176685206)

[2.3 Requisitos funcionais 7](#_Toc176685207)

[2.4 Requisitos não funcionais 7](#_Toc176685208)

[2.5 Comparativo entre sistemas 8](#_Toc176685209)

[Descrever resumidamente e apresentar uma tabela de comparação do sistema desenvolvido versus 1 ou 2 outros sistemas de mercado. 8](#_Toc176685210)

[2.6 Cronograma 8](#_Toc176685211)

[3. Documentação do Sistema 9](#_Toc176685212)

[3.1 Metodologia de Desenvolvimento 9](#_Toc176685213)

[3.2 Diagramas UML 9](#_Toc176685214)

[Diagrama de caso de uso (1º) 9](#_Toc176685215)

[Diagrama de classes (2º) 9](#_Toc176685216)

[Diagrama de sequência (2º) 9](#_Toc176685217)

[3.3 Modelo Conceitual (1º) 10](#_Toc176685218)

[3.4 Modelo lógico (1º) 10](#_Toc176685219)

[3.5 Modelo Físico (2º) 10](#_Toc176685220)

[3.6 Recursos e ferramentas (1º e 2º) 10](#_Toc176685221)

[3.7 Etapas / Sprints realizados (1º e 2º) 10](#_Toc176685222)

[3.8 Interface do usuário 10](#_Toc176685223)

[4. Testes e Qualidade 11](#_Toc176685224)

[4.1 Estratégia de Testes: Descrever a estratégia de testes adotada 11](#_Toc176685225)

[4.2 Resultados dos Testes: Apresentar os resultados dos testes realizados 11](#_Toc176685226)

[4.3 Garantia da Qualidade: Descrever as práticas adotadas para garantia da qualidade 11](#_Toc176685227)

[4.4 Requisitos mínimos de hardware e software para o sistema 11](#_Toc176685228)

[4.5 Contrato para desenvolvimento de software 11](#_Toc176685229)

[5. Considerações Finais 11](#_Toc176685230)

[6 Referências 11](#_Toc176685231)

[Anexo I - Diário de bordo 11](#_Toc176685232)

[Anexo II – Cronograma efetivo 11](#_Toc176685233)

[Anexo III – Evidências 11](#_Toc176685234)

# Apresentação da Empresa

A empresa Datamind, com focos em pequenos negócios, busca juntamente a empresa [INSERIR NOME OUTRA EMPRESA], a satisfação de seus clientes e demonstrar como tecnologias criam um grande impacto na vida de todos do cotidiano. Abaixo estão apresentados os logotipos de ambas:

**Logotipo

Descrição gerada automaticamente com confiança baixaLogotipo Datamind**

**Logotipo** **OUTRA EMPRESA**

Logotipo Empresa + Logo Cliente

## 1.1 Missão

A empresa busca desenvolver soluções tecnológicas acessíveis e intuitivas para pequenos negócios, potencializando a organização, a gestão e experiência do pequeno empresário, aplicando conceitos de software apreendidos durante a execução do atual projeto.

1.2 Visão

Ser a principal escolha de software de gestão para microempreendedores, oferecendo ferramentas simples e eficientes que ajudam a automatizar suas operações e a crescer de forma sustentável, buscando em primeiro lugar, a satisfação do atual cliente deste projeto e a expansão do contato e influência.

1.3 Valores

Os valores da empresa Datamind, contemplam:

* **Inovação**: Buscamos sempre soluções criativas e inovadoras.
* **Qualidade**: Oferecemos produtos que agregam valor e confiança.
* **Acessibilidade**: Tornamos a tecnologia acessível para micro e pequenos empreendedores

## 1.4 Link Repositório

Abaixo segue o link do repositório principal:

<https://github.com/GabrielVictorino8266/pi_2_semestre>

# Escopo do sistema

O escopo de um sistema refere-se ao conjunto de funcionalidades, características e requisitos que definem o que o sistema irá realizar e como ele irá operar para atender às necessidades dos usuários e stakeholders. Ele delimita os limites do projeto, determinando o que está dentro e o que está fora do âmbito do sistema.

Necessário apresentar também o nome do sistema desenvolvido.

## Objetivos do projeto

Lista dos resultados específicos que o projeto visa alcançar.

## 2.2 Técnica de levantamento de requisitos

Descrever qual(is) técnicas de levantamento foram utilizadas e detalhar o processo.

## Requisitos funcionais

Requisitos funcionais, são declarações dos serviços que o sistema deve fornecer, do modo como o sistema deve reagir a determinadas entradas e de como deve se comportar em determinadas situações. (Sommerville)

**Exemplo de descrição:**

|  |  |  |
| --- | --- | --- |
| **RF01** | Cadastro de usuário | Essencial |
| O sistema deve ter uma tela para cadastro de novos usuários. | | |

|  |  |  |
| --- | --- | --- |
| **RF02** | Login de Usuário | Importante |
| O sistema deve ter uma tela com campos de usuário de acesso e senha. | | |

|  |  |  |
| --- | --- | --- |
| **RF03** | Requisito x | Desejável |
| descrição | | |

## 2.4 Requisitos não funcionais

Requisitos não funcionais são restrições sobre os serviços ou funções oferecidas pelo sistema. Eles incluem restrições de tempo, restrições sobre o processo de desenvolvimento e restrições impostas por padrões. Os requisitos não funcionais se aplicam, frequentemente, ao sistema como um todo, em vez de às características individuais ou aos serviços. (Sommerville)

**Exemplo de descrição**

Tabela

Descrição gerada automaticamente

## 2.5 Comparativo entre sistemas

## Descrever resumidamente e apresentar uma tabela de comparação do sistema desenvolvido versus 1 ou 2 outros sistemas de mercado.

**Tabela 1 – Comparativo das funcionalidades da aplicação**

|  |  |  |  |
| --- | --- | --- | --- |
| Funcionalidades | Padaria Facil | Padoca App | Meu Sistema |
| Gerenciar Encomendas | **X** | **X** | **X** |
| Cadastro de Fornecedores | **X** | **-** | **X** |
| Gerenciar entregas | **-** | **-** | **X** |
| Gerenciar Promoções | **-** | **-** | **X** |
| Compartilhar solicitações nas redes sociais | **-** | **-** | **X** |
| Ser multiplataforma | **X** | **-** | **X** |

## 2.6 Cronograma

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Tarefas** | **22/08** | **29/08 a 05/09** | **12/09 a 26/09** | **01/10 a**  **22/10** | **29/10 a 12/11** | **13/11 a 22/11** | **03/12** |
| Definição Grupos | X |  |  |  |  |  |  |
| Criação Empresa |  | X |  |  |  |  |  |
| Escopo Sistema |  |  | X |  |  |  |  |
| Requisitos |  |  | X | X | X |  |  |
| Diagramas |  |  | X | X | X |  |  |
| Protótipo |  |  |  |  | X | X |  |
| Documentação |  | X | X | X | X | X |  |
| Entrega |  |  |  |  |  | X |  |
| Apresentação |  |  |  |  |  |  | X |

# Documentação do Sistema

Neste capítulo, são apresentados os documentos técnicos que descrevem os aspectos fundamentais do sistema desenvolvido, fornecendo uma base sólida para compreensão e manutenção futura. A documentação é uma parte essencial do processo de desenvolvimento de software, pois oferece um registro detalhado das decisões tomadas e das características do sistema.

## Metodologia de Desenvolvimento

Conjunto de processos, práticas e diretrizes que guiam o ciclo de vida do desenvolvimento de software, desde a concepção até a entrega e manutenção do produto final. Ela define como o trabalho é organizado, como as decisões são tomadas e como as atividades são realizadas ao longo do projeto. Uma metodologia pode ser ágil, como o Scrum ou o Kanban, ou tradicional, como o modelo em cascata.

## 3.2 Diagramas UML

## Diagrama de caso de uso (1º)

## Diagrama de classes (2º)

## Diagrama de sequência (2º)

## Modelo Conceitual (1º)

## Modelo lógico (1º)

## Modelo Físico (2º)

## Recursos e ferramentas (1º e 2º)

## 3.7 Etapas / Sprints realizados (1º e 2º)

## 3.8 Interface do usuário

# Testes e Qualidade

## 4.1 Estratégia de Testes: Descrever a estratégia de testes adotada

## 4.2 Resultados dos Testes: Apresentar os resultados dos testes realizados

## 4.3 Garantia da Qualidade: Descrever as práticas adotadas para garantia da qualidade

## 4.4 Requisitos mínimos de hardware e software para o sistema

## 4.5 Contrato para desenvolvimento de software

# Considerações Finais

Conclusão: Concluir o trabalho e destacar aprendizados

Contribuições Individuais: Descrever as contribuições individuais de cada membro da equipe

# 6 Referências

# Anexo I - Diário de bordo

**Físico:** Feito pelo grupo.

**Digital:** individual / Links

# Anexo II – Cronograma efetivo

# Anexo III – Evidências

Link Live, participantes externos, prints.