



# Projeto A3

## Gerenciador de Tarefas

### Documentação de Desenvolvimento de Software



Disciplina

UC – Gestão e Qualidade de Software

Prof. Calvetti



Aluno

Bruno Rodrigues Reis

8222243147



### Objetivos do Projeto

- ✓ Demonstrar aplicação prática de conceitos de engenharia de software
- ✓ Elaborar documentação completa de desenvolvimento
- ✓ Implementar sistema funcional com testes abrangentes
- ✓ Aplicar gestão de configuração e controle de qualidade



# Metodologia e Planejamento

Abordagem sistemática baseada em engenharia de software



## Abordagem Metodológica

- Processo iterativo e incremental
- Foco na qualidade e documentação



## Cronograma Geral

Duração Total:	4 semanas
Esforço Estimado:	60 horas
Equipe:	Multidisciplinar

## Fases do Projeto



### Análise & Design

Semana 1

Requisitos, arquitetura e especificações



### Implementação

Semanas 2-3

Desenvolvimento backend e frontend



### Testes & Docs

Semana 4

Validação e documentação final

## Recursos Necessários



### Humanos

Desenvolvedores, Analistas, Testadores



### Tecnológicos

Figma, Git, Frameworks de Teste



### Infraestrutura

Ambientes de Dev, Teste e Produção



# Tecnologias Escolhidas

Stack tecnológico otimizado para praticidade e eficiência



## Flask Framework

Microframework Python para desenvolvimento web

Escolha Principal

### Backend



#### Python + Flask

Framework web minimalista



#### SQLite

Banco de dados leve



### Frontend



#### HTML5 + CSS3

Estrutura e estilização



#### JavaScript

Interatividade e APIs



### Justificativa da Escolha do Flask



#### Praticidade

Microframework leve e de fácil aprendizado



#### Flexibilidade



#### Comunidade

Ampla documentação e suporte ativo



#### Integração



# Especificação de Requisitos

Definição clara das funcionalidades e características de qualidade



## Requisitos Funcionais

RF001

### Criação de Tarefas

Título obrigatório e descrição opcional

RF002

### Visualização de Tarefas

Lista organizada e responsiva

RF003

### Marcação de Conclusão

Alternar status concluído/pendente

RF004

### Exclusão de Tarefas

Remoção com confirmação



## Requisitos Não Funcionais

RNF001

### Usabilidade

Interface intuitiva e responsiva

RNF002

### Performance

Resposta em menos de 2 segundos

RNF003

### Compatibilidade

Principais navegadores web

RNF004

### Confiabilidade

Persistência segura de dados



## Escopo do Sistema



### Incluído no Escopo

- CRUD completo de tarefas



### Fora do Escopo

- Autenticação de usuários

# </> Implementação do Sistema

Sistema funcional demonstrando conceitos de engenharia de software

## Interface do Sistema

### Gerenciador de Tarefas

Organize suas atividades

Nova Tarefa

Adicionar

Implementar casos de teste

Concluir

Excluir

## Arquitetura Implementada

### Frontend

Interface responsiva com HTML5, CSS3 e JavaScript vanilla

### Backend API

Flask com endpoints RESTful e CORS habilitado

### Persistência

SQLite com SQLAlchemy ORM para modelagem de dados

## Funcionalidades Implementadas



Criar



Visualizar



Concluir



Excluir



# Planejamento de Testes

Estratégia abrangente para garantia da qualidade



## Estratégia de Testes



### Abordagem Sistemática

Baseada em Gonçalves et al. e melhores práticas



### Cobertura Completa

Testes funcionais e não funcionais



### Execução Estruturada

Cronograma detalhado com marcos definidos



## Métricas do Plano

12

Casos de Teste

25

Dias de Execução

7

Marcos Definidos

100%

Cobertura RF



## Cronograma de Execução

Preparação

Dias 1-3

Unitários

Dias 4-6

Integração

Dias 7-10

Funcionais

Dias 11-15

Usabilidade

Dias 16-18

Correções

Dias 19-22

Aceitação

Dias 23-25



## Casos de Teste Principais



## CrITÉRIOS de Aprovação



# Gestão de Configuração

Controle sistemático de mudanças e versionamento



## Controle de Versões



### Git + GitHub

Sistema distribuído com repositório remoto



### Estratégia de Branches

Main protegido, develop para integração



### Versionamento Semântico

Tags v1.0.0, v1.1.0 para releases



## Itens de Configuração



Código Fonte

Controlado



Documentação

Versionado



Configurações

Rastreado



Dados de Teste

Incluído



## Fluxo de Trabalho Git



**Desenvolvimento**

Feature branches



**Pull Request**

Code review



**Merge**

Branch principal



**Release**

Tag + Deploy



# Resultados e Conclusões

Síntese dos resultados obtidos e lições aprendidas

## Projeto Concluído com Sucesso

Todos os objetivos alcançados dentro do prazo estabelecido

100%

Requisitos Atendidos

12/12

Casos de Teste Aprovados

4

Semanas de Desenvolvimento

7/7

Marcos Atingidos



### Principais Conquistas



#### Sistema Funcional Completo

Todas as funcionalidades implementadas e testadas



#### Documentação Abrangente

Todos os artefatos exigidos pelo Projeto A3



#### Qualidade Validada

Testes sistemáticos e controle de qualidade



#### Controle de Mudanças



### Lições Aprendidas



#### Planejamento é Fundamental

Decisões bem fundamentadas facilitam execução



#### Requisitos Claros

Especificação detalhada evita ambiguidades



#### Testes Sistemáticos

Identificação precoce garante qualidade



#### Controle de Mudanças