

Plano de Gerência de Software - Sistema de Gerenciamento de Tarefas (TDL)

Versão: Final

Data: 08 de dezembro de 2025

Autor(es): Ana Alice, Brian Kevelyn, João Victor, Luan Pablo

Histórico de Revisões

Versão	Autor(es)	Descrição	Data
Draft	Brian	Versão inicial do plano de gerência	15/09/2025
Preliminary	Luan	Revisão da versão inicial	20/09/2025
Preliminary	Ana Alice	Revisão da versão inicial	10/10/2025
Final	João Victor	Versão final do plano de gerência	08/12/2025

1. Introdução

Projeto a ser desenvolvido por alunos do curso de Projeto de Desenvolvimento de Sistemas para Internet como requisito da disciplina Projeto Interdisciplinar. O objetivo é aplicar práticas e metodologias das disciplinas técnicas em um sistema real. Este documento define o plano de gerência de software, incluindo processos organizacionais, cronograma, riscos, objetivos e artefatos que orientarão a execução do projeto.

1.1. Resumo do Projeto

O sistema TDL (To-Do List) é uma aplicação web responsiva projetada para auxiliar usuários no gerenciamento de tarefas. As principais funcionalidades incluem:

- Usuários - Cadastro, listagem, atualização e exclusão de contas de acesso.
- Tarefas - Criação, edição, exclusão e atualização de status (pendente, em progresso, concluída).

- Interface - Design minimalista, moderno e responsivo (desktop e mobile), utilizando glassmorphism, cores neutras e tipografia clean.

1.2. Entrega de Tarefas

Data	Marco	Tarefas Desenvolvidas	Gerente
30/09/2025	A	Definição do processo de desenvolvimento; Documento de Visão (escopo, requisitos); Diagramas de caso de uso e descrição; Diagrama de arquitetura inicial; Documento de risco; Documento de ambiente	Luan
20/10/2025	B	Revisão de documentos; Diagrama de arquitetura (camadas); Diagrama de domínio; Diagrama de banco de dados; Proposta de interface (Figma); Cenários de casos de uso	Brian
10/11/2025	C	Implementação inicial de casos de uso (CRUD de usuários e tarefas); Integração backend e frontend	João Victor
20/11/2025	D	Implementação de casos de uso; Testes unitários e de integração; Ajustes visuais (UI/UX)	Ana Alice
08/12/2025	Final	Produto final concluído; Testes finais; Revisão de documentação; Apresentação	João Victor

1.3. Evolução do Plano de Gerência de Software

Espera-se que as tarefas planejadas sejam cumpridas e entregues dentro dos prazos estabelecidos no cronograma.

Versão	Autor(es)	Descrição	Data
Draft	João Victor	Versão inicial do plano de gerência	15/09/2025

Final	Ana Alice	Versão final do plano de gerência	08/12/2025
-------	-----------	-----------------------------------	------------

1.4. Documentos

Os artefatos gerados e os documentos desenvolvidos para a configuração do ambiente estarão disponíveis no GitHub através das URLs:

- Repositório GitHub - <https://github.com/anaambr/Projeto-TDL.git>
- Ferramenta de Gerenciamento - GitHub Projects

2. Organização do Projeto

2.1. Processo

O desenvolvimento do projeto utilizará a metodologia Scrum, um processo ágil. A equipe será composta por 4 pessoas, com os papéis de Product Owner, Scrum Master, Desenvolvedores e Designer UX/UI. O projeto será dividido em ciclos quinzenais (sprints), onde cada integrante poderá atuar como Scrum Master em pelo menos um marco. No início de cada sprint, será realizada uma reunião com o cliente (Product Owner) para priorizar funcionalidades, e a equipe selecionará as atividades a serem desenvolvidas. Em caso de conclusão ou impedimento de uma tarefa, o responsável deverá relatar o progresso ou problema via GitHub Projects ou e-mail para o grupo.

2.2. Histórico de Tarefas

Tarefa	Data	Poderá haver mudanças?	Entregue ao cliente?	Autor
Diagrama de arquitetura (inicial)	28/09/2025	Sim	Sim	Luan
Diagrama de caso de uso	27/09/2025	Sim	Sim	Brian
Documento de risco	29/09/2025	Não	Sim	Ana Alice
Definir requisitos	26/09/2025	Sim	Sim	João Victor

Definir processo	29/09/2025	Sim	Sim	Ana Alice
Definir escopo	29/09/2025	Sim	Sim	João Victor
Documento de ambiente	29/09/2025	Sim	Sim	Luan
Cenários de casos de uso	19/10/2025	Sim	Sim	Brian
Diagrama de arquitetura (camadas)	18/10/2025	Sim	Sim	João Victor
Diagrama de banco de dados	18/10/2025	Sim	Sim	Luan
Proposta de interface (Figma)	19/10/2025	Sim	Sim	Brian
Revisão de documentos	18/10/2025	Sim	Sim	Brian
Plano de gerência	20/10/2025	Sim	Não	Ana Alice

2.3. Estrutura Organizacional

Marco A

Integrante	Função	Responsabilidades
Ana Alice	Desenvolvedor	Documento de risco

João Victor	Desenvolvedor	Requisitos
Brian	Desenvolvedor	Diagrama de caso de uso
João Victor	Gerente	Processo
Luan	Desenvolvedor	Diagrama de arquitetura
Ana Alice	Desenvolvedor	Escopo

Marco B

Integrante	Função	Responsabilidades
Ana Alice	Desenvolvedor	Banco de dados e implementação
Brian	Desenvolvedor	Interfaces (Figma)
João Victor	Desenvolvedor	Arquitetura (camadas)
Luan	Desenvolvedor	Documentação e implementação
Brian	Gerente	Revisão e implementação
Brian	Desenvolvedor	Documentação

Marco C

Integrante	Função	Responsabilidades
Luan	Desenvolvedor	Interface e implementação (frontend)
Brian	Desenvolvedor	Interface e implementação (frontend)
João Victor	Gerente	Implementação (backend)
Brian	Desenvolvedor	Pesquisa, documentação e implementação
Ana Alice	Desenvolvedor	Implementação (backend)

Marco D

Integrante	Função	Responsabilidades
Luan	Gerente	Implementação e testes
Luan	Desenvolvedor	Implementação
Brian	Desenvolvedor	Implementação e testes
Ana Alice	Desenvolvedor	Implementação
João Victor	Desenvolvedor	Implementação

Marco Final

Integrante	Função	Responsabilidades
João Victor	Gerente	Testes finais e apresentação
Brian	Desenvolvedor	Testes
Luan	Desenvolvedor	Testes
Ana Alice	Desenvolvedor	Testes
João Victor	Desenvolvedor	Documentação e apresentação

3. Processo Gerencial

3.1. Planejamento

Marco	Data Inicial	Data Final	Descrição
A	15/09/2025	30/09/2025	Estabelecer contato com o cliente, definir requisitos e horários de reuniões semanais.
B	01/10/2025	20/10/2025	Validar proposta de arquitetura e interface, revisar documentação e coletar requisitos adicionais.
C	21/10/2025	10/11/2025	Concluir validação de arquitetura e interface, implementar CRUD inicial, definir contrato com o cliente.

D	11/11/2025	20/11/2025	Implementar, testar e integrar casos de uso; validar com o cliente.
Final	21/11/2025	08/12/2025	Corrigir, testar e integrar funcionalidades; validar com o cliente; preparar apresentação final.

3.2. Objetivos e Prioridades

- Garantir a execução eficiente do processo de desenvolvimento para produzir artefatos de qualidade.
- Assegurar uma especificação clara do projeto e uma implementação consistente.
- Entregar os artefatos ao cliente dentro dos prazos estabelecidos.

3.3. Riscos

3.3.1. Riscos Altos

- Falta de conhecimento em tecnologias - A aquisição de conhecimentos técnicos durante o projeto pode limitar o progresso devido à experiência limitada em algumas ferramentas.
- Fator tempo - O prazo curto para entrega (dezembro de 2025) pode comprometer a qualidade.
- Fuga do escopo - Inclusão de funcionalidades não previstas, desviando o foco do projeto.

3.3.2. Riscos Médios

- Comprometimento da equipe - Baixa dedicação ou atrasos por parte dos membros.
- Imprevistos pessoais - Doenças ou outros problemas que afetem a participação dos membros.
- Problemas no ambiente de desenvolvimento - Falhas em dependências ou ferramentas.

3.4. Mecanismos de Controle

O acompanhamento das tarefas será realizado por meio de:

- GitHub Projects - Gerenciamento de tarefas e progresso.
- E-mails - Comunicação de relatórios e impedimentos.
- Reuniões semanais - Dailies para monitoramento.

4. Processo Técnico

4.1. Métodos, Ferramentas e Técnicas

O projeto será desenvolvido com as seguintes tecnologias e práticas:

- Linguagem - Python (FastAPI) para o backend; JavaScript (React com Vite) para o frontend.
- IDE - VS Code.
- Banco de Dados - SQLmodel(desenvolvimento).
- Diagramas UML - Figma.
- Prototipação - Figma.
- Testes de API - Postman / Thunder Client.
- Controle de Versão - GitHub.
- Gerenciamento - GitHub Projects.
- Metodologia - Scrum (sprint planning, dailies, review e retrospectiva).

4.2. Artefatos

Os artefatos gerados no desenvolvimento do projeto incluem:

- Plano de Gerência de Software
- Documento de Visão
- Documento de Risco
- Diagramas de Arquitetura
- Documento de Ambiente
- Diagrama de Caso de Uso
- Diagrama de Domínio
- Diagrama de Banco de Dados
- Cenários de Casos de Uso
- Protótipo de Interface (Figma)
- Códigos-fonte (Backend e Frontend)
- Documentação Técnica (APIs, banco, arquitetura)
- Manual do Usuário
- Apresentação Final