

Plano de Gerência – Easy Invest

1. Introdução

1.1 Resumo do Projeto

O **Easy Invest** é um sistema web que auxilia iniciantes no mundo das finanças, permitindo simulações de investimentos e empréstimos de forma simples e acessível.

Funcionalidades principais:

- Cadastro e login de usuários
- Simulação de investimentos
- Simulação de empréstimos
- Relatórios e gráficos
- Interface web responsiva

1.2 Entregas Planejadas (Marcos)

Marco	Data	Entregas
A	01/10/2025 - 15/10/2025	Definição de requisitos, escopo e identidade visual
B	16/10/2025 – 30/10/2025	Protótipos navegáveis (Figma) e modelagem do banco (SQLModel).
C	21/11/2025 – 25/11/2025	Implementação da simulação de investimentos
D	26/11/2025 – 30/11/2025	Implementação da simulação de empréstimos
E	01/12/2025 – 04/12/2025	Relatórios, gráficos e integração
F	05/12/2025 – 08/12/2025	Testes, refinamentos e documentação final
Final	10/12/2025	Entrega e apresentação

2. Organização do Projeto

2.1 Processo de Desenvolvimento

O projeto seguirá a metodologia SCRUM, com sprints quinzenais.

O papel de Scrum Master será rotativo, iniciado com Teófilo e depois alternando entre os membros.

2.2 Estrutura Organizacional

Integrante	Função	Responsabilidade
George Kauan	Projeção e Frontend	Estruturação das páginas, layout e desenvolvimento em React
Ayslan	Frontend	Desenvolvimento das interfaces em React, integração com o backend
Ítalo Verde	Backend	Implementação em Flask + SQLModel, APIs de simulação de investimentos
Hiandro	Backend	Implementação em Flask + SQLModel, APIs de simulação de empréstimos
Teófilo	Rotacionando / Scrum Master	Apoio em frontend, backend, documentação e organização das sprints

3. Processo Gerencial

3.1 Cronograma

Sprint	Período	Atividades
Sprint 1	01/10 – 15/10	Requisitos, escopo, identidade visual
Sprint 2	16/10 – 30/10	Protótipos (Figma), modelagem com SQLModel
Sprint 3	21/11 – 25/11	Desenvolvimento inicial do frontend (React) e APIs de investimentos

Sprint 4	26/11 – 30/11	Desenvolvimento das APIs de empréstimos e integração frontend
Sprint 5	01/12 – 04/12	Relatórios, gráficos e ajustes de interface
Sprint 6	05/12 – 08/12	Testes finais, refinamentos e documentação
Final	10/12	Apresentação e entrega

3.2 Objetivos e Prioridades

- Garantir simplicidade e clareza para o usuário iniciante
- Cumprir os prazos definidos em cada sprint
- Manter consistência entre frontend (React) e backend (Flask/SQLModel)

3.3 Riscos

Nível	Risco	Mitigação
Alto	Atrasos no desenvolvimento frontend	Divisão clara de tarefas entre George e Ayslan.
Alto	Falta de domínio no SQLModel	Estudo de exemplos e implementação incremental.
Médio	Mudança de requisitos	Reuniões semanais com cliente/PO.
Baixo	Problemas de comunicação	Uso de GitHub Projects e reuniões fixas.

3.4 Controle

- Reuniões semanais de acompanhamento
- Uso de GitHub Projects (Kanban) para gerenciar tarefas
- Relatórios de sprint (feito/não feito/impedimentos)
- Testes contínuos (unitários e de integração)

4. Processo Técnico

4.1 Tecnologias

- **Backend:** Python (Flask + SQLModel)
- **Frontend:** React
- **Banco de Dados:** SQLModel (com SQLite em dev)
- **Versionamento:** GitHub
- **Design:** Figma
- **Gerência:** GitHub Projects
- **Testes:** PyTest (backend), React Testing Library (frontend)

4.2 Artefatos

- Plano de Gerência
- Documento de Visão e Requisitos
- Diagramas UML (caso de uso, banco, arquitetura)
- Protótipos (Figma)
- Código-fonte (backend/frontend)
- Documentação de API
- Relatórios de testes
- Manual do Usuário