# **PROJETO: MyCost — FASE 1 (CONCEPÇÃO + ESCOPO)**

[PROJETO: MyCost](https://drive.google.com/drive/folders/12JmD2-AUbQz6ZF4CbSaZFAEBo3pICAbe?usp=sharing)

**Equipe:**

29948487 - Alex Muchau

30118271 - Gabriel Pepe

30092914 - Vinicius José Santana

45554862 - Vinícius Macedo dos Reis

45212333 - Willian G. Simão

### **1. DESCRIÇÃO DO SISTEMA**

O **MyCost** é uma aplicação web projetada para simplificar o controle financeiro pessoal. A plataforma permite que os usuários registrem suas despesas e receitas de forma rápida e intuitiva, categorizando cada transação para fornecer uma visão clara de seus hábitos de consumo. Ao final de cada período, a aplicação gera relatórios visuais e insights personalizados, ajudando o usuário a entender seus gastos, identificar oportunidades de economia e planejar seu orçamento com mais eficiência.

### **2. FUNCIONAMENTO DA EQUIPE**

Somos uma equipe de desenvolvimento acadêmico, organizada sob a metodologia ágil Scrum. Atuamos em um modelo híbrido, com reuniões de planejamento e revisão realizadas presencialmente na faculdade e alinhamentos diários (daily stand-ups) feitos de forma remota. Utilizamos ferramentas como o Notion para gestão de tarefas, Discord para comunicação e Google Drive para a documentação do projeto, garantindo colaboração e transparência no processo.

### **3. LISTA PRELIMINAR DE REQUISITOS**

* Cadastro e autenticação de usuários.
* Registro, edição e exclusão de transações financeiras (receitas e despesas).
* Criação e personalização de categorias de gastos.
* Geração de relatórios financeiros (semanal, mensal, anual).
* Definição de metas de economia.
* Criação de orçamentos por categoria.
* Dashboard visual com resumo financeiro.
* Sistema de notificações.
* Painel administrativo para suporte e gerenciamento do sistema.

### **4. ATORES DO PROCESSO**

| **Ator** | **Descrição** | **Responsabilidades** |
| --- | --- | --- |
| **Usuário Padrão** | Principal usuário do sistema, que busca gerenciar suas finanças pessoais. | Realizar cadastro e login, gerenciar transações, criar categorias e orçamentos, visualizar relatórios e definir metas. |
| **Suporte Técnico** | Membro da equipe de atendimento responsável por auxiliar os usuários. | Acessar painel de suporte, responder a tickets, visualizar dados de usuários (com permissão) para resolver problemas e acessar chat. |
| **Administrador do Sistema** | Responsável pela manutenção, segurança e gestão da aplicação. | Gerenciar a base de dados, monitorar a performance do sistema, gerenciar contas de usuários e publicar atualizações. |

### **5. HISTÓRIAS DE USUÁRIO**

**Integrante 1:**

* **US01:** Como **usuário**, quero **me cadastrar e fazer login de forma segura** para **proteger minhas informações financeiras**.
* **US02:** Como **usuário**, quero **registrar minhas despesas e receitas rapidamente, associando-as a categorias** para **organizar minhas finanças**.
* **US03:** Como **usuário**, quero **editar ou excluir uma transação lançada incorretamente** para **manter meus dados precisos**.
* **US04:** Como **suporte**, quero **ter acesso a um painel com um chat direto com o usuário** para **oferecer ajuda em tempo real**.

**Integrante 2:**

* **US05:** Como **usuário**, quero **visualizar um dashboard com gráficos que resumem minha situação financeira (receitas vs. despesas)** para **entender rapidamente para onde meu dinheiro está indo**.
* **US06:** Como **usuário**, quero **gerar relatórios financeiros (semanais, mensais, anuais)** para **analisar meus hábitos de consumo ao longo do tempo**.
* **US07:** Como **usuário**, quero **criar minhas próprias categorias de gastos (ex: "Lazer", "Educação")** para **personalizar o controle financeiro de acordo com minha realidade**.
* **US08:** Como **administrador**, quero **visualizar logs de erro e performance do sistema** para **garantir a estabilidade da aplicação**.

**Integrante 3:**

* **US09:** Como **usuário**, quero **definir orçamentos mensais para diferentes categorias (ex: R$ 500 para "Restaurantes")** para **controlar meus gastos e não exceder meus limites**.
* **US10:** Como **usuário**, quero **receber uma notificação quando meu gasto em uma categoria atingir 80% do orçamento definido** para **ficar alerta e evitar gastar demais**.
* **US11:** Como **usuário**, quero **criar metas de economia (ex: "Viagem de férias - R$ 3.000")** para **me manter motivado a poupar dinheiro**.
* **US12:** Como **suporte**, quero **acessar uma seção de FAQ e tutoriais** para **orientar os usuários sobre as funcionalidades do sistema**.

**Integrante 4:**

* **US13:** Como **administrador**, quero **gerenciar as contas dos usuários (ativar, desativar, resetar senha)** para **manter a segurança e o controle da plataforma**.
* **US14:** Como **usuário**, quero **poder exportar meus relatórios em formato PDF ou CSV** para **compartilhar ou analisar os dados em outras ferramentas**.
* **US15:** Como **usuário**, quero **acessar a aplicação em qualquer dispositivo (desktop, tablet, celular)** para **gerenciar minhas finanças de onde eu estiver**.
* **US16:** Como **suporte**, quero **visualizar os dados de um usuário de forma anonimizada** para **diagnosticar um problema sem violar sua privacidade**.

### **6. REQUISITOS FUNCIONAIS**

* **RF01:** O sistema deve permitir o cadastro de novos usuários com e-mail e senha.
* **RF02:** O sistema deve permitir a autenticação de usuários (login).
* **RF03:** O sistema deve permitir o registro de transações financeiras, informando descrição, valor, data, tipo (receita/despesa) e categoria.
* **RF04:** O sistema deve permitir a edição e exclusão de transações existentes.
* **RF05:** O sistema deve permitir a criação, edição e exclusão de categorias de gastos personalizadas.
* **RF06:** O sistema deve exibir um dashboard principal com um resumo gráfico das finanças do mês corrente.
* **RF07:** O sistema deve gerar relatórios financeiros filtrando por período (semana, mês, ano).
* **RF08:** O sistema deve permitir que o usuário defina orçamentos mensais por categoria.
* **RF09:** O sistema deve enviar notificações ao usuário quando o gasto se aproximar do limite do orçamento.
* **RF10:** O sistema deve ter uma funcionalidade para o usuário criar e acompanhar o progresso de metas de economia.
* **RF11:** O sistema deve disponibilizar um painel de suporte para atendimento ao cliente.
* **RF12:** O painel de suporte deve incluir um chat para comunicação direta.
* **RF13:** O sistema deve permitir a exportação de dados em formato PDF e CSV.
* **RF14:** O painel de administração deve permitir o gerenciamento de contas de usuários.
* **RF15:** O painel de administração deve exibir logs de performance e erro.
* **RF16:** O sistema deve permitir que o suporte consulte dados de usuários de forma controlada e segura.

### **7. REQUISITOS NÃO FUNCIONAIS**

* **RNF01:** **Desempenho:** A geração de relatórios mensais deve ser concluída em até 3 segundos.
* **RNF02:** **Disponibilidade:** O sistema deve ter uma disponibilidade de 99,8% em horário comercial.
* **RNF03:** **Usabilidade:** A interface deve ser intuitiva e responsiva, adaptando-se a desktops, tablets e smartphones.
* **RNF04:** **Segurança:** Todas as informações de usuário e transações devem ser criptografadas no banco de dados.
* **RNF05:** **Segurança:** A comunicação entre o cliente e o servidor deve utilizar o protocolo HTTPS (TLS 1.2+).
* **RNF06:** **Conformidade:** O sistema deve estar em conformidade com a Lei Geral de Proteção de Dados (LGPD).
* **RNF07:** **Compatibilidade:** A aplicação web deve ser compatível com as versões mais recentes dos navegadores Google Chrome, Mozilla Firefox e Microsoft Edge.
* **RNF08:** **Escalabilidade:** A arquitetura deve suportar o crescimento para 10.000 usuários ativos sem degradação significativa da performance.
* **RNF09:** **Confiabilidade:** Backups diários e automáticos do banco de dados devem ser realizados.
* **RNF10:** **Manutenibilidade:** O código-fonte deve seguir padrões de projeto e ser bem documentado para facilitar futuras manutenções.
* **RNF11:** **Feedback Imediato:** O sistema deve fornecer uma resposta visual (como uma animação ou mensagem de confirmação) após o usuário registrar uma transação, confirmando que a ação foi concluída com sucesso.
* **RNF12:** **Política de Senha Forte:** O sistema deve impor, no momento do cadastro e da troca de senha, uma política que exija no mínimo 8 caracteres, contendo letras maiúsculas, minúsculas, números e um símbolo especial.

### **8. TABELA UNIFICADA (Exemplos)**

| **Estória de Usuário** | **Requisito Funcional (RF)** | **Requisito Não-Funcional (RNF)** | **Categoria** | **Critérios de Aceitação** |
| --- | --- | --- | --- | --- |
| **US01:** Como usuário, quero me cadastrar e fazer login de forma segura... | **RF01, RF02** | **RNF04, RNF05** | Segurança | A senha deve ser armazenada com hash. A comunicação deve ser via HTTPS. |
| **US06:** Como usuário, quero gerar relatórios financeiros... | **RF07, RF13** | **RNF01** | Desempenho | O relatório do último mês deve carregar em menos de 3 segundos. O usuário deve conseguir baixar o relatório em PDF. |
| **US15:** Como usuário, quero acessar a aplicação em qualquer dispositivo... | **Todos os RFs de interface** | **RNF03** | Usabilidade | A interface se ajusta corretamente em uma tela de 360px de largura (mobile) e 1920px (desktop). |

### **9. EAP - ESTRUTURA ANALÍTICA DE PROJETO**

1. Gestão do Projeto  
   1.1. Elaborar o plano do projeto e cronograma  
   1.2. Realizar reuniões de planejamento, revisão e retrospectiva (Sprints)  
   1.3. Gerenciar o backlog do produto (Notion)
2. Desenvolvimento Frontend  
   2.1. Implementar módulo de autenticação (cadastro, login, recuperação de senha)  
   2.2. Desenvolver o Dashboard principal  
   2.3. Desenvolver a tela de gerenciamento de transações (CRUD)  
   2.4. Implementar a tela de relatórios e gráficos  
   2.5. Desenvolver a funcionalidade de orçamentos e metas  
   2.6. Desenvolver painel de configurações do usuário
3. Desenvolvimento Backend  
   3.1. Modelar e implementar o banco de dados (Usuários, Transações, Categorias, Metas)  
   3.2. Desenvolver API RESTful para comunicação  
   3.2.1. Endpoints de autenticação (JWT)  
   3.2.2. Endpoints para CRUD de transações, categorias, etc.  
   3.2.3. Endpoints para geração de dados para relatórios  
   3.3. Implementar sistema de notificações  
   3.4. Implementar painel de administração e suporte
4. Qualidade e Testes (QA)  
   4.1. Elaborar casos de teste  
   4.2. Executar testes unitários e de integração  
   4.3. Realizar testes de usabilidade e interface  
   4.4. Executar testes de segurança (SQL Injection, XSS)
5. Implantação (DevOps)  
   5.1. Configurar ambientes (desenvolvimento, produção)  
   5.2. Configurar pipeline de CI/CD (Integração e Entrega Contínua)  
   5.3. Implementar sistema de monitoramento de logs e performance  
   5.4. Realizar o deploy da versão inicial (MVP)

### **10. EVIDÊNCIAS DE METODOLOGIA ADOTADA**

* **Abordagem:** A equipe adotou a metodologia ágil **Scrum**, organizando o trabalho em Sprints de duas semanas. Até o momento não foram definidos os papéis de Product Owner (responsável por definir as prioridades), Scrum Master (facilitador do processo) e a Equipe de Desenvolvimento, todos atuaram de maneira equiparada nas funções realizadas.
* **Rituais:** Encontros semanais na Universidade para definição do Roadmap e alinhamentos assíncronos por meio do Whatsapp para acompanhamento do andamento do projeto.
* **Ferramentas:** **Notion** para gestão do backlog e das tarefas, **GitHub** para versionamento do código, **Google Drive** para documentação e **Whatsapp** para comunicação assíncrona.

# **ENTREGA FASE 2 - Gerenciamento das Comunicações + Recursos Humanos + Interessados**

# [PROJETO: MyCost](https://drive.google.com/drive/folders/12JmD2-AUbQz6ZF4CbSaZFAEBo3pICAbe?usp=sharing) **Equipe:** Gabriel Pepe, Alex Muchau, Vinícius Santana, Vinícius M., Willian Simão.

## **1. Diretório e Matriz de Responsabilidades da Equipe**

| **Atividade / Entregável** | **Responsável** |
| --- | --- |
| Gestão do Projeto, Requisitos e Comunicação | Gabriel Pepe |
| Desenvolvimento do Backend (API e Banco de Dados) | Vinícius Santana e Vinícius M. |
| Desenvolvimento do Frontend (Interface do Usuário) | Alex Muchau e Gabriel Pepe |
| Garantia de Qualidade (Testes) e Infraestrutura (DevOps) | Willian Simão |
| Documentação Técnica do Projeto | Willian Simão e Vinícius M. |

## **2. Partes Interessadas (Stakeholders)**

| **Parte Interessada** | **Papel/Descrição no Projeto** | **Expectativas em relação ao projeto** |
| --- | --- | --- |
| Usuários Finais (Clientes) | Pessoas que buscam uma solução para organizar suas finanças pessoais. Podem ser estudantes, jovens profissionais ou famílias. | Um aplicativo seguro, fácil de usar, com relatórios claros e funcionalidades que agreguem valor (como metas e orçamentos). |
| Investidores | Um grupo de "investidores anjo" que aportou o capital inicial para viabilizar o desenvolvimento da plataforma. | Retorno sobre o investimento (ROI), crescimento rápido da base de usuários, um modelo de negócio sustentável (ex: versão Premium) e relatórios de progresso. |
| Equipe de projeto | Os próprios fundadores e desenvolvedores da solução (Gabriel, Alex, Vinícius S., Vinícius M., Willian). | Construir um produto de sucesso, desenvolver habilidades técnicas e de gestão, criar um portfólio robusto e, futuramente, gerar receita. |
| Consultor Financeiro (Parceiro) | Um especialista em finanças pessoais que atua como consultor para garantir que as funcionalidades do app sejam relevantes e precisas. | Que a ferramenta siga boas práticas financeiras e que possa, futuramente, ser usada como um canal para seus próprios serviços de consultoria. |
| ANPD (Agência Nacional de Proteção de Dados) | Órgão regulador responsável por fiscalizar a conformidade com a Lei Geral de Proteção de Dados (LGPD). | Que o MyCost trate os dados sensíveis dos usuários com a máxima segurança, transparência e em total conformidade com a legislação. |
| Futura Equipe de Marketing | Time responsável por divulgar o aplicativo e atrair novos usuários após o lançamento. | Um produto estável, com uma proposta de valor clara e diferenciais competitivos para serem usados em campanhas de marketing digital. |

## **3. Plano de Comunicação**

Definir como será feita a comunicação de pelo menos três itens do projeto. Exemplos: reuniões, ferramentas digitais, relatórios, feedbacks, etc.

| **Item de Comunicação** | **Objetivo** | **Frequência** | **Meio/Ferramenta** | **Responsável pela comunicação** |
| --- | --- | --- | --- | --- |
| Reunião de Planejamento da Sprint (Sprint Planning) | Definir o escopo e as metas da próxima Sprint, selecionando itens do backlog. | A cada 2 semanas (início da Sprint) | Presencial / Google Meet | Scrum Master |
| Reunião Diária (Daily Stand-up) | Sincronizar as atividades diárias, identificar impedimentos e garantir o alinhamento da equipe. | Diariamente (30 minutos) | Discord (canal de voz) | Toda a equipe de desenvolvimento |
| Relatório de Status para o Professor | Apresentar o progresso do projeto, as entregas concluídas e os próximos passos, conforme o cronograma da disciplina. | Mensalmente ou conforme solicitado | E-mail formal / Documento PDF no Google Drive | Equipe |
| Revisão da Sprint (Sprint Review) | Apresentar o incremento de software funcional desenvolvido na Sprint para os stakeholders e obter feedback. | A cada 2 semanas (final da Sprint) | Presencial (em aula) / Google Meet | Toda a equipe de desenvolvimento |

## **4. Evidências da Metodologia Adotada**

| **Abordagem (metodologia em cascata, espiral, incremental etc:** | O projeto MyCost adota a **metodologia ágil Scrum**. A escolha se justifica pela natureza iterativa e incremental do desenvolvimento de software, que permite adaptações rápidas a novos requisitos e feedbacks. A equipe se organiza com papéis definidos: um Product Owner (responsável por gerenciar o backlog e definir as prioridades, papel acumulado pelo Scrum Master), um Scrum Master (responsável por facilitar os rituais e remover impedimentos) e a Equipe de Desenvolvimento. |
| --- | --- |
| **Rituais:** | * + **Sprint Planning:** Reuniões quinzenais para planejar o trabalho.   + **Daily Stand-ups:** Encontros diários de 30 minutos para sincronia.   + **Sprint Review:** Demonstração do trabalho concluído ao final da Sprint.   + **Sprint Retrospective:** Reunião interna para discutir melhorias no processo. |
| **Ferramentas:** | * + **GitHub:** Utilizado como repositório central para o controle de versão do código-fonte.   + **Trello:** Adotado para a gestão visual do projeto. O board é organizado com as colunas: Backlog, To Do (A Fazer), In Progress (Em Andamento), Code Review (Revisão) e Done (Concluído).   + **Google Drive:** Usado para armazenar toda a documentação do projeto, incluindo este documento, atas de reunião e artefatos.   + **Discord:** Plataforma de comunicação principal da equipe, com canais de texto para discussões gerais e um canal de voz para as reuniões diárias. |
| **Evidências anexas:** | Print do board com colunas definidas;    <https://ibb.co/5xztrHk5>  Link do repositório;  <https://github.com/vinicius-mreis/projet-mycost.git> |