**PÓS-GRADUAÇÃO EM DESENVOLVIMENTO FULL STACK**

**DESENVOLVIMENTO DE APLICAÇÃO DE FINANÇAS PESSOAIS**

LUCAS COIMBRA SCAGLIA

Orientador: ????

2025



**SUMÁRIO**

[1. Contextualização da proposta 3](#_Toc203632762)

[2. Objetivos da construção da solução 4](#_Toc203632763)

[3. Elaboração da jornada do usuário 5](#_Toc203632764)

[4. Apelo mercadológico da solução 6](#_Toc203632765)

[5. Ciclo de desenvolvimento da solução 7](#_Toc203632766)

[6. Mockup da proposta de solução 8](#_Toc203632767)

[7. Arquitetura de Software 9](#_Toc203632768)

[8. Validação da solução 10](#_Toc203632769)

[9. Registros das evidências do projeto 11](#_Toc203632770)

[10. Considerações finais e expectativas 12](#_Toc203632771)

[Referências 13](#_Toc203632772)

# Contextualização da proposta

As atuais aplicações disponíveis para controle de finanças pessoais, apesar dos vastos recursos, acabam sendo abandonadas em pouco tempo. O motivo disto é que esperam que o usuário as use diariamente, documentando cada compra, gasto ou investimento. Essa atividade acaba sendo considerada maçante, e o esquecimento torna os dados inconsistente com o tempo. Ao perceber o trabalho gerado, os usuários muitas vezes podem optar por outros meios onde se sintam mais confortáveis para fazer este controle, ainda que mais simples e com menos recursos, como uma planilha no computador ou um caderno.

Diante desta problemática, a proposta é de desenvolver uma aplicação que vise oferecer uma solução prática e minimalista, onde o planejamento financeiro é realizado somente uma vez por mês. A cada novo ciclo, o usuário conseguirá perceber se foi possível cumprir o planejamento anterior e fazer os ajustes necessários para o próximo mês, numa jornada de melhoria contínua e de autoconhecimento.

# Objetivos da construção da solução

Pretendo alinhar os objetivos deste trabalho de conclusão com minha motivação ao começar este curso, que é aprimorar meu conhecimento técnico e prático sobre o desenvolvimento de aplicações web.

Atualmente, sou coordenador de engenharia de software em uma das maiores empresas de varejo do Brasil. Ao longo da minha jornada, me especializei em otimizar os processos de equipes de desenvolvimento de software com ênfase em Qualidade, DevOps e Gestão de Releases. Contudo, apesar de ter conhecimento em desenvolvimento de automações e pipelines, em conversas mais técnicas não consigo absorver todo o contexto do problema, e assim, acabo não contribuindo como gostaria.

Dado este contexto, pretendo desenvolver uma aplicação simples de ponta a ponta, onde precisarei definir seu escopo e jornadas do usuário, tomar as decisões de arquitetura e modelagem de dados, realizar a programação, e, por fim, disponibilizar a solução rodando em nuvem.

# Elaboração da jornada do usuário

A seguir, serão descritas as principais jornadas do usuário.

* 1. **Jornadas de Onboarding**
     1. *Cadastro*

No primeiro acesso, o usuário precisará criar uma conta utilizando um e-mail e cadastrando uma senha.

* + 1. *Acesso*

O usuário deverá acessar informando o e-mail e a senha.

Após entrar, será exibida a tela que representa o mês corrente.

* + 1. *Dicas de utilização*

Ao acessar pela primeira vez, para cada campo obrigatório que ainda não tiver sido preenchido, como a Renda Líquida, Pagamentos Fixos e Investimentos, será apresentada uma mensagem exigindo uma ação do usuário.

* + 1. *Cadastro da renda líquida*

Este campo é obrigatório. Deve deixar claro para o usuário que o valor é aproximado, e poderá ser ajustado sempre que necessário.

Além do valor, o usuário deverá informar o dia do mês em que ocorre o recebimento.

* + 1. *Cadastro de pagamentos com vencimentos específicos*

Para cada pagamento cadastrado, deve ser informado o vencimento. O valor mensal configurado é aproximado, e poderá editado sempre que necessário.

Deve deixar claro para o usuário que contas variáveis feitas ao longo do mês não devem ser lançadas. E caso sejam feitas no cartão de crédito, lançar apenas o vencimento da fatura geral, e não cada valor individualmente.

Este campo é opcional.

* + 1. *Cadastro de economias recorrentes*

O usuário deverá cadastrar as reservas que costuma fazer mensalmente, seja para investimentos ou algum objetivo específico. O valor mensal configurado é aproximado, e poderá editado sempre que necessário.

Este é opcional.

* + 1. *Cadastro do saldo de economias*

O usuário poderá configurar reservas com o nome que desejar, apenas para ter uma visualização mensal sobre o saldo acumulado. Este valor é sempre informado manualmente.

* 1. **Jornada Principal: controle financeiro mensal**
     1. *Notificação de recebimento*

Quando chegar o dia do mês em que o usuário recebe seu pagamento, a aplicação enviará uma notificação lembrando o usuário de fazer seu planejamento mensal. Será possível configurar o dia de recebimento nas configurações.

* + 1. *Revisão e ajuste dos valores para o mês corrente*

A cada jornada mensal, o usuário deverá ser orientado a revisar se os valores previstos se confirmaram, e editar os valores se necessário.

#### *Alteração dos valores previamente informados*

Todos os campos informados pelo usuário será editáveis posteriormente.

#### *Inclusão e Edição de itens*

Será criar ou editar itens de pagamento ou investimentos a qualquer momento, bem como cadastrar saldos de economias.

* + 1. *Marcação do checklist de transações realizadas*

Será possível marcar os itens de pagamento ou investimento a fim de auxiliar o usuário a controlar quais movimentações já foram realizadas.

* + 1. *Cálculo do saldo disponível*

Assim que o usuário confirmar o valor recebido, ainda que não marque nenhum dos itens como realizado, o saldo disponível será projetado, diminuindo o valor recebido pelo total dos itens.

* + 1. *Fechamento do período*

Deve haver uma opção para fechar o período, onde o usuário deverá confirmar se o Saldo disponível projetado é realmente o valor que sobrou para o resto do mês. Nesta ocasião, ele poderá optar por: a) confirmar que está correto; b) voltar para ajustar; c) editar o saldo manualmente e encerrar o período sem ajustar.

* + 1. *Visualização das notificações semanais de saldo disponível projetado*

O usuário poderá configurar as notificações para receber lembretes de qual saldo deveria ter proporcional ao tempo que falta para encerrar o mês, a fim de se organizar. Ele poderá escolher quais dias da semana este lembrete ocorre. Por padrão, a aplicação irá sugerir notificações uma vez por semana, nas quintas-feiras.

# Apelo mercadológico da solução

A proposta da solução é simplificar ao máximo a experiência do usuário, permitindo um controle financeiro mensal, ao invés de interações diárias como é comumente exigido nas aplicações já disponíveis no mercado, como Guiabolso, Organizze e Mobills.

A aplicação desenvolvida será focada apenas em jornadas simples e práticas. Seu apelo é focado na relação entre o esforço do usuário e o valor percebido. Dito isso, não há a pretensão de desenvolver funcionalidades robustas como integrações bancárias, relatórios complexos e cadastros de metas financeiras.

Do ponto de vista mercadológico, o público-alvo são pessoas que já tentaram utilizar outras ferramentas, e que desistiram por falta de tempo ou disciplina. Um público que valorize a autonomia para decidir quais itens são relevantes cadastrar e que gostaria de apoio não só para organizar essas contas, mas para conseguir passar o mês com o valor que foi programado.

O desenvolvimento desta solução vai além de um exercício técnico. Ela demonstra potencial para explorar um nicho que ainda é pouco atendido.

# Ciclo de desenvolvimento da solução

O Desenvolvimento da aplicação conduzido através das seguintes etapas:

1. Ideação da solução.
2. Especificação das jornadas de usuário e requisitos.
3. Criação do Mockup da solução utilizando o Miro.
4. Definição da arquitetura, escolhendo as tecnologias que serão utilizadas na solução.
5. Desenho conceitual e modelagem do banco de dados.
6. Criação dos repositórios de desenvolvimento e armazenamento no GitHub.
7. Criação do repositório Backend em Node.js com Express.
8. Criação do repositório Frontend em React com Vite.
9. Criação do banco de dados Postgress.
10. Configuração e instalação e de dependências.
11. Configuração da aplicação para rodar em containers.
12. Criação de tela de testes garantindo a comunicação entre frontend, backend e banco de dados.
13. Desenvolvimento Full-Stack da solução.

# Mockup da proposta de solução

As telas a seguir foram feitas utilizando o Miro e foram criadas como protótipo do design e usabilidade da aplicação. O material aqui proposto não representa o design final da aplicação.

Tela de Login

Interface gráfica do usuário, Diagrama, Aplicativo

O conteúdo gerado por IA pode estar incorreto.

Tela de Planejamento Mensal

Interface gráfica do usuário, Tabela

O conteúdo gerado por IA pode estar incorreto.

Utilização da tela de Planejamento

Uma imagem contendo Linha do tempo

O conteúdo gerado por IA pode estar incorreto.

Tela de Reservas:

Interface gráfica do usuário, Gráfico, Aplicativo

O conteúdo gerado por IA pode estar incorreto.

Tela de Configurações:

Interface gráfica do usuário, Texto, Aplicativo, chat ou mensagem de texto

O conteúdo gerado por IA pode estar incorreto.

**Exemplos de notificações:**

1. **Controle do saldo de despesas variáveis:**

* “Tá sobrando mês pro seu dinheiro? Faltam 15 dias e você deveria ter uns R$ 2.000 de saldo. Confere aí se tá tudo certo!”
* “Seu dinheiro precisa durar mais 15 dias. R$ 2.000 seria o saldo ideal agora…”
* “Faltam 15 dias e o saldo ideal é R$ 2.000. Tá tranquilo ou já tá pedindo socorro?”
* “Metade do mês já foi… e os R$ 2.000 ainda tão por aí? Ou já foi tudo?”
* “Se o mês fosse uma maratona, você deveria estar com R$ 2.000 de saldo agora. Tá no ritmo certo?”

1. **Notificações do dia de recebimento:**

* “Dinheiro na conta? Hora do plano de ataque!”
* “Já fez seu plano do mês ou vai deixar o dinheiro fugir de novo?”
* “O salário chegou... e já quer ir embora? Planeja ele aí!”
* “Bora dar uma olhada nos boletos antes que eles te surpreendam.”
* “Seu "eu do futuro" agradece se você planejar agora.”

# Arquitetura de Software

Desenho da arquitetura:

Diagrama

O conteúdo gerado por IA pode estar incorreto.

# Validação da solução

< farei após o desenvolvimento >

Tão importante quando o desenvolvimento da solução é a garantia da qualidade desta. O desenvolvimento de uma solução requer que teste sejam realizados a fim de reduzir ao máximo falhas de lógica, bem como para garantir a coerência entre versões que são elaboradas.

Descreva as estratégias empregadas para a validação da solução implementada.

# Registros das evidências do projeto

O trabalho está disponível no GitHub: <https://github.com/lucascoimbra-lab/finance-app>.

Na pasta “evidências” que está na raíz do projeto, estão disponíveis:

* Imagens do Mockup feito no Miro.
* Desenho do modelo conceitual do banco de dados.
* Script de criação utilizado para modelar o banco de dados.
* Collection do Postman para testes dos endpoints da aplicação.
* Arquivo de PowerPoint utilizado para o desenho da arquitetura.
* <inserir outras conforme forem surgindo>

# Considerações finais e expectativas

A trajetória do curso e o fechamento deste com a elaboração do trabalho aqui apresentado certamente são motivo de exaltação. Como fechamento deste documento, apresente um pouco da sua trajetória ao longo do curso e quais expectativas imagina surgir a partir de tal formação.

# Referências

Citar todas as referências utilizadas no trabalho (seguir as normas da ABNT).

**REDAÇÃO E LAYOUT**

A redação precisa ser clara e fluida, prezando pelo layout do template previsto.

O Trabalho de Conclusão de Curso deve ser autoral e inédito. A originalidade é um requisito essencial à aprovação do TCC, neste sentido, não serão aceitos trabalhos já apresentados ou submetidos a avaliações anteriores, para obtenção de grau em outros cursos ou níveis de ensino, publicações em periódicos, revistas ou outros meios, bem como, obras derivadas, reproduções totais ou parciais de produções do próprio autor ou de terceiros.

Todas as referências a trabalhos e obras de terceiros ou do próprio autor, somente podem ser incorporadas ao texto por meio de citações devidamente referenciadas como citação direta ou indireta, segundo as normas da ABNT.

Quaisquer trechos oriundos de outros materiais, obras de terceiros ou do próprio autor inseridos no TCC não documentados como citação, caracterizarão trechos com similaridade, falta de originalidade e consequentemente poderão ensejar a reprovação do aluno.