

1. Introdução

Este documento apresenta todos os aspectos relacionados ao planejamento, desenvolvimento e testes de um sistema de Contas a Pagar. O objetivo é organizar e controlar as despesas e receitas dos usuários, oferecendo uma solução funcional, confiável e acessível.

2. User Storys

2.1. Quem é o usuário?

Usuário interessado em organizar suas despesas e receitas.

2.2. O que o usuário quer fazer?

O usuário deseja ajustar e controlar sua vida financeira incluindo no app suas despesas e receitas.

Exemplo: "Novo Lançamento", "Descrição", "Data de Lançamento", "Salvar"

2.3. Qual é o objetivo/benefício?

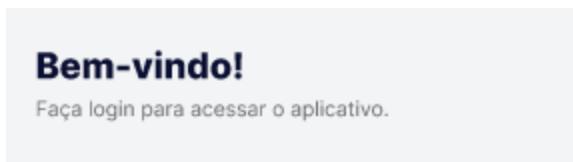
Ter controle e segurança sobre suas finanças pessoais.

2.4. User Story 1 – Cadastro de contas a pagar

Como usuário, eu quero cadastrar uma nova conta a pagar com data de vencimento, para que eu possa organizar minhas despesas e receitas.

3. Protótipo de Interfaces

3.1. Tela de Login



E-mail

Digite seu e-mail

Senha

Digite sua senha



[Esqueci minha senha](#)

Entrar

ou



[Ainda não tenho cadastro](#)

3.2. Tela de Cadastro

Cadastre-se!

Crie sua conta para acessar o aplicativo

E-mail

Digite seu e-mail

Nome

Digite seu nome

Senha

Digite sua senha

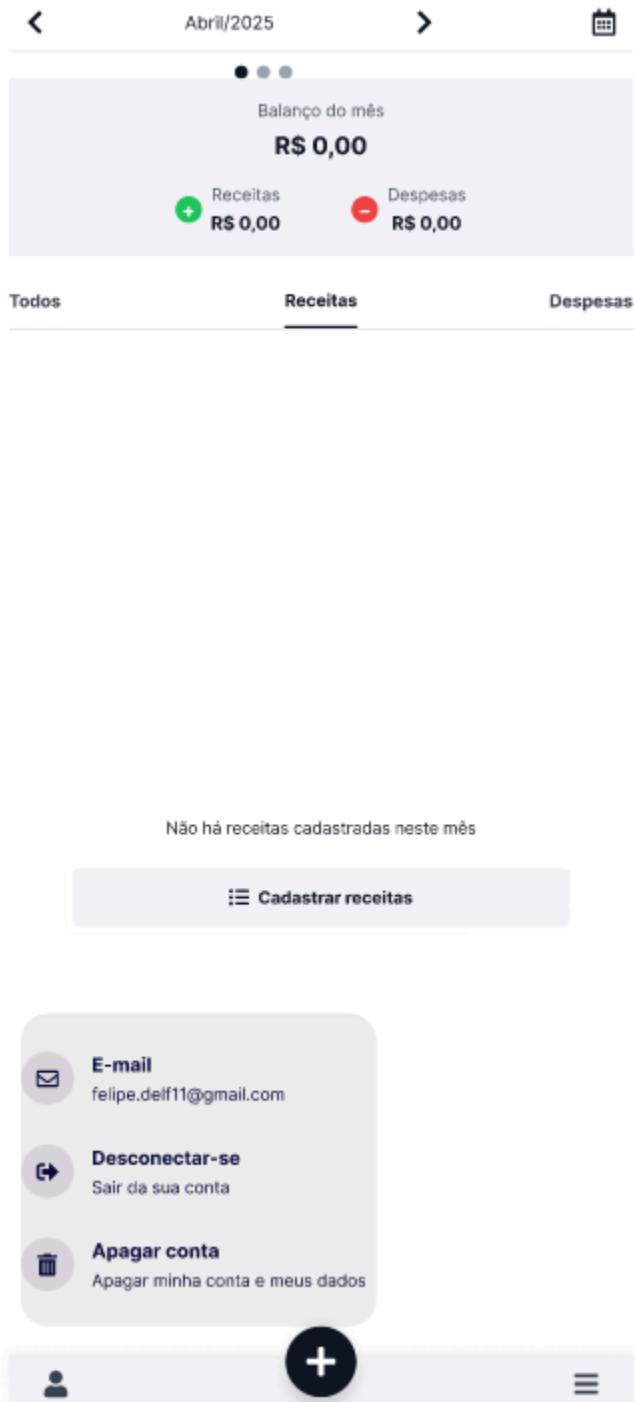
Confirmação de senha

Confirme sua senha

Cadastre-se

[Já tenho cadastro](#)

6.3. Tela de Menu



6.4. Tela de Receitas

The screenshot shows the Receipts screen for the month of April 2025. At the top, there is a navigation bar with arrows for navigating between months, the text "Abril/2025", and a calendar icon. Below the navigation bar, there is a summary section titled "Balanço do mês" showing a balance of "+ R\$ 100,00". This section includes icons for "Receitas" (R\$ 100,00) and "Despesas" (R\$ 100,00). Below this summary, there are three tabs: "Todos", "Receitas" (which is selected and underlined), and "Despesas". A single receipt entry is listed: "Receita teste" dated "23/05/2025" with a value of "+ R\$ 100,00" and a note "1/2". Below this entry, a message states "Não há Receitas cadastradas neste mês". At the bottom, there is a button labeled "Cadastrar Receitas". In the footer, there is a link to "Política de Privacidade" (Privacy Policy) with a shield icon. The bottom navigation bar features icons for user profile, add (+), and menu.

6.5.1. Tela de Cadastro de Despesas (única)

< Novo lançamento

Despesa Receita

R\$ 0,00

Descrição

Data de vencimento

23/04/2025



Não repetir

Sempre

Parcelado

Salvar

6.5.2. Tela de Cadastro de Despesas (parcelado)

< Novo lançamento

Despesa Receita

R\$ 0,00

Descrição

Data de vencimento

23/05/2025



Não repetir

Sempre

Parcelado

Frequência

Mensalmente

Número de parcelas

Salvar

6.5.3. Tela de Cadastro de Despesas (recorrente)

< Novo lançamento

Despesa Receita

R\$ 0,00

Descrição

Data de vencimento

23/05/2025



Não repetir

Sempre

Parcelado

Frequência

Mensalmente

Salvar

6.6. Tela de Teste de Despesa

< Novo lançamento

Despesa Receita

R\$ 100,00

Descrição

Compra mercado Livre

Data de vencimento

23/04/2025

Não repetir Sempre Parcelado

Salvar

6.7. Tela de Despesa

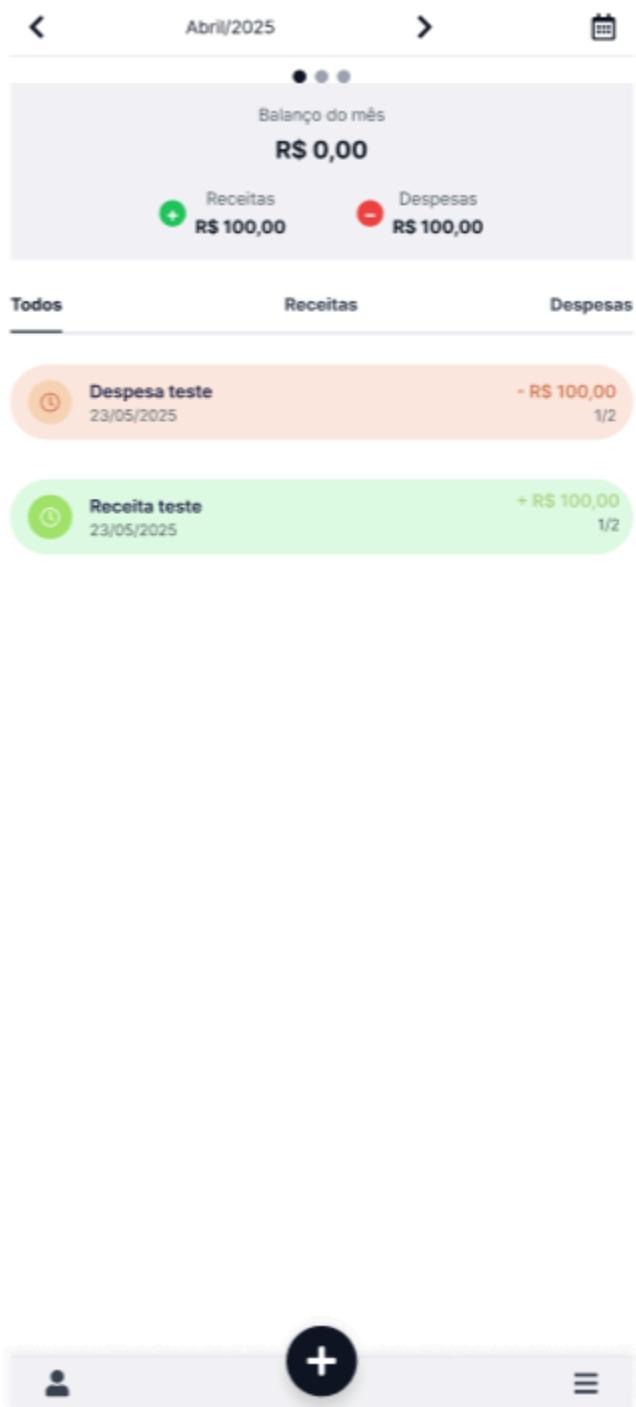
The screenshot shows the 'Despesas' (Expenses) screen of a mobile application. At the top, there is a header with navigation arrows for 'Abril/2025', a date selector icon, and a three-dot menu icon. Below the header, a summary box displays the month's balance: 'Balanço do mês -R\$ 100,00'. It also shows 'Receitas R\$ 100,00' with a green plus sign icon and 'Despesas R\$ 100,00' with a red minus sign icon. A three-dot menu icon is located above the summary box.

Below the summary box, there are three tabs: 'Todos', 'Receitas', and 'Despesas'. The 'Despesas' tab is selected, indicated by an underline. Underneath the tabs, a single expense item is listed: 'Despesa teste' from '23/05/2025' with a value of '- R\$ 100,00' and a status of '1/2'. This item is highlighted with an orange rounded rectangle.

In the center of the screen, a message states 'Não há despesas cadastradas neste mês' (No expenses registered this month). At the bottom, a button labeled 'Cadastrar despesa' with a plus sign icon is available for adding new expenses.

At the very bottom of the screen, there is a navigation bar with icons for user profile ('Pessoa'), a central plus sign ('+'), and a three-dot menu ('Mais').

6.8. Tela de Todos



4. Definição de Pontos de Função do Aplicativo de Contas a Pagar

4.1. Classificação dos Requisitos Funcionais

4.1.1. Entradas Externas (EE)

Tipo de Função	Descrição	TD (TDE)	RL	Complexidade
EE	Registro de Contas Únicas	3 (valor, data, descrição)	1 (Contas)	Baixa(3)
EE	Registro de Contas Recorrentes	3 (valor, data, descrição)	1 (Contas)	Média(4)
EE	Autenticação de Usuário	4 (nome, senha, e-mail, id)	1 (Usuário)	Média(4)
EE	Gestão de Categorias	2 (nome, status)	1 (Categoria)	Baixa(3)
EE	Registro de Receitas	3 (valor, data, descrição)	1 (Receita)	Baixa(3)
EE	Marcação de Pagamentos	2 (status, data de pagamento)	1 (Pagamento)	Baixa(3)

- RF01: Registro de Contas Únicas
- RF02: Registro de Contas Recorrentes
- RF03: Autenticação de Usuários
- RF07: Gestão de Categorias
- RF08: Registro de Receitas
- RF09: Marcação de Pagamentos

Total de Entradas Externas: 20

4.1.2. Saídas Externas (SE)

Tipo de Função	Descrição	TD (TDE)	RL	Complexidade
SE	Backup Automático	3 (dados a salvar, timestamp, status)	1 (Backup)	Baixo(4)
• RF04: Backup Automático (considerando que gere uma saída de confirmação)				

Total de Saídas Externas: 4

4.1.3. Consultas Externas (CE)

Tipo de Função	Descrição	TD (TDE)	RL	Complexidade
CE	Funcionamento Offline	5 (valor, tipo, status, data, descrição)	1 (Contas)	Baixa
• RF05: Funcionamento Offline (consultas aos dados armazenados)				

Total de Consultas Externas: 6

4.1.4. Arquivos Lógicos Internos (ALI)

Tipo de Função	Descrição	TD (TDE)	RL	Complexidade
ALI	Contas	5 (Valor, tipo, Data, Descrição, Status)	2 (Contas Únicas, Contas recorrentes)	Baixa
ALI	Autenticação de usuário	3 (Nome, E-mail, Senha)	1 (Usuário)	Baixa
ALI	Backup	3 (Data/hora, Status, Dados envolvidos)	2 (Backup, Logs)	Baixa
ALI	Categorias	3 (Tipo, Descrição, Status)	2 (Despesa, Receita)	Baixa
ALI	Receitas	5 (Valor, Data, Descrição)	1 (Receitas)	Baixa

ALI	Pagamentos	4 (Status, Valor, Data)	1 (Conta)	Baixa
•	RF01: Contas Únicas			
•	RF02: Contas Recorrentes			
•	RF03: Autenticação de Usuários			
•	RF04: Backup			
•	RF07: Categorias			
•	RF08: Receitas			
•	RF09: Pagamentos			

Total de Arquivos Lógicos Internos: 42

4.1.5. Arquivos de Interface Externa (AIE)

Tipo de Função	Descrição	TD (TDE)	RL	Complexidade
EE	Autenticação de Usuários	2 (Token, Id)	1 (Usuário)	Baixa

- RF03: Autenticação de Usuários (interage com serviços externos como Google e Facebook)

Total de Arquivos de Interface Externa: 5

4.2. Total de Pontos de Função

Tipo de Função	Complexidade	Quantidade	Total
Entrada Externa (EE)	Baixo(3)	4	12
Entrada Externa (EE)	Médio(4)	2	8
Saída Externa (SE)	Baixo(4)	1	4
Consulta Externa (CE)	Baixa(3)	1	6
Arquivo Lógico Interno (ALI)	Baixa(3)	6	42
Arquivo de Interface Externa (AIE)	Baixa(3)	1	5

Total = 12 + 8 + 4 + 6 + 42 + 5 = 83 pontos de função

4.3. Reajuste dos Pontos de Função

	Grau de Influência (0-5)	
Descrição		Justificativa
Comunicação de dados	3	Sincronização automática com a nuvem e operação offline/online.
Processamento distribuído	2	Backup e sincronização exigem algum processamento remoto.
Desempenho	5	Garantia de bom desempenho, inclusive offline.
Operação em ambiente restrito	2	Compatibilidade com diferentes Androids.
Transações online	3	Backup automático, autenticação social e sincronização.
Eficiência do usuário final	5	Interface simples e intuitiva.
Atualização online	2	Atualizações via Google Play.
Complexidade de processamento	3	Relatórios, recorrência, autenticação, sincronização.
Reusabilidade	2	Aplicativo modular e preparado para internacionalização.
Facilidade de instalação	3	Distribuição via Google Play.
Facilidade de operação	5	Requisitos de usabilidade.
Facilidade de mudança	3	Atualizações constantes.
Confiabilidade	5	Estabilidade, confiabilidade e segurança destacadas.

Facilidade de teste	3	Testes para garantir segurança, confiabilidade e performance.
---------------------	---	---

4.4. Definição Final dos Pontos de Função

Item	Valor
PF Não Ajustados (PFNA)	83
Fator de Ajuste (FA)	1,11
PF Ajustados (PFA)	92,13

5. Ciclo de Vida (Modelo Cascata)

Utilizamos para o ciclo de vida do nosso software o modelo cascata, que é uma abordagem tradicional e sequencial para o desenvolvimento de software, onde cada fase do processo depende da conclusão da anterior. É especialmente eficaz quando os requisitos são bem definidos desde o início e mudanças durante o desenvolvimento são mínimas.

5.1. Como Funciona o Modelo Cascata?

5.1.1. Levantamento de Requisitos

- Coleta e documentação das necessidades do cliente e dos usuários.
- Estudo de viabilidade técnica e financeira do projeto. Profissionais de Tecnologia.

5.1.2. Análise e Especificação de Requisitos

- Detalhamento dos requisitos funcionais e não funcionais do sistema.
- Criação de documentos que servirão de base para as próximas fases.

5.1.3. Projeto do Sistema

- Desenvolvimento da arquitetura do sistema e design de componentes.
- Definição de interfaces e fluxos de dados.

5.1.4. Implementação (Codificação)

- Programação do sistema conforme as especificações do projeto.
- Integração de módulos e componentes.

5.1.5. Testes

- Verificação da funcionalidade e desempenho do sistema.
- Identificação e correção de erros e falhas.

5.1.6. Implantação

- Instalação do sistema em ambiente de produção.
- Treinamento de usuários e entrega do produto final.

5.1.7. Manutenção

- Correção de defeitos pós-implantação.
- Atualizações e melhorias conforme necessário.

5.2. Vantagens e Desvantagens

As vantagens são que temos estruturas mais claras onde cada etapa tem objetivos e entregáveis bem definidos, facilitando o planejamento e o controle do projeto. Temos uma documentação mais completa e detalhada em cada etapa, útil para manutenção futura e conformidade regulatória, e outra vantagem é a facilidade de gerenciar. As desvantagens do modelo cascata são a rigidez, que tem dificuldade em acomodar mudanças nos requisitos após o início do desenvolvimento; o feedback tardio do cliente, pois o cliente só visualiza o produto final após a conclusão de todas as fases, o que pode resultar em desalinhamento com suas expectativas; e, por último, o risco de retrabalho, pois erros identificados nas fases finais podem exigir revisões significativas nas etapas anteriores.

5.3 Adicionando o sistema agile no cascata

5.3.1 Scrum

- Baseado em sprints (ciclos curtos de trabalho).
- Equipes auto-organizadas e reuniões diárias (daily scrum).
- Foco em entregas incrementais e feedback constante.

5.3.2 Kanban

- Visualização do fluxo de trabalho com quadros e cartões.
- Limita tarefas em andamento para melhorar o fluxo.
- Ideal para projetos com demandas contínuas.

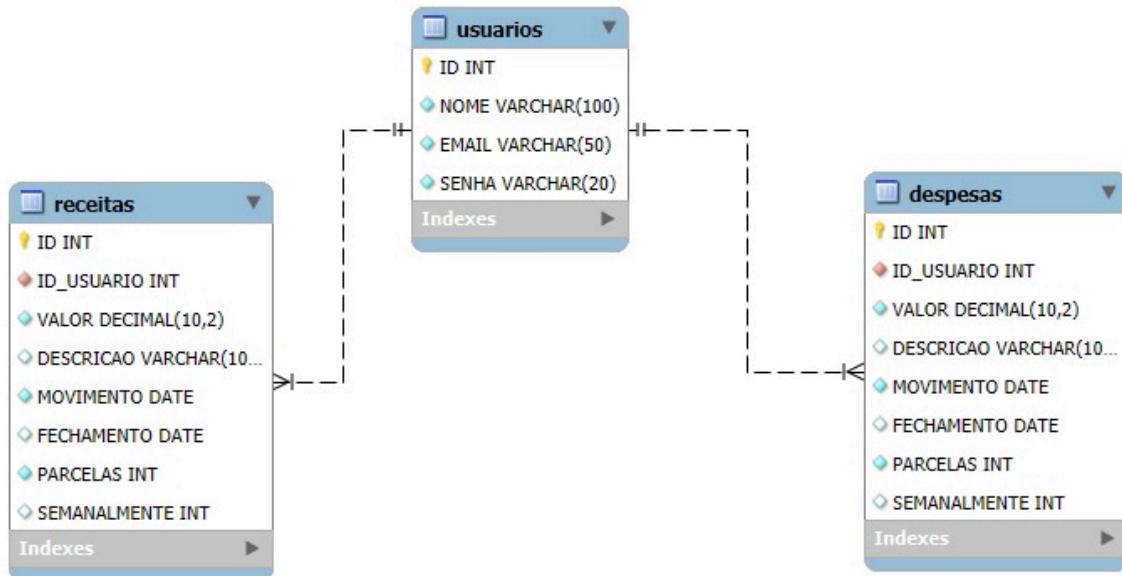
5.3.3 XP (Extreme Programming)

- Ênfase em práticas como programação em par, testes automatizados e feedback rápido.
- Foco em qualidade de código e colaboração com o cliente.

5.3.4 Team development

- Liderança eficaz
- Comunicação aberta
- Papéis bem definidos
- Confiança mútua
- Gestão de conflitos
- Feedback constante
- Clareza nos objetivos

6. Arquitetura



7. Plano de Testes

7.1. Escopo dos Testes

7.1.1. Funcionalidades a Serem Testadas

- Cadastro de Contas a Pagar: Inclusão de novas contas com campos como fornecedor, valor, data de vencimento e categoria.
- Teste de Telas: Verificação se todas as telas do módulo estão funcionando corretamente e conforme o layout.
- Teste de Registro de Despesas e Receitas: Validação dos fluxos que permitem cadastrar lançamentos financeiros de saída (despesas) e entrada (receitas).
- Visualizar Todas as Despesas: Conferência da funcionalidade que exibe a listagem de despesas lançadas com filtros e totais.

7.1.2. Funcionalidades Fora de Escopo

- Edição de contas (fluxo de edição será tratado em um módulo separado).
- Interações com sistemas externos (e.g., sistemas bancários).
- Funcionalidades amplas de contas a receber que não estejam diretamente ligadas ao registro de receitas.

7.2. Estratégia de Teste

7.2.1. Tipos de Teste

- Teste Funcional: Verificar se cada funcionalidade atende aos requisitos especificados.
- Teste de Interface do Usuário (UI): Avaliar a conformidade das telas com os padrões definidos.
- Teste de Usabilidade: Assegurar que o sistema seja intuitivo e fácil de usar.
- Teste de Regressão: Garantir que novas alterações não afetem funcionalidades existentes.
- Teste de Compatibilidade: Verificar o funcionamento em diferentes navegadores e dispositivos.

7.2.2. Ferramentas de Teste

- Gerenciamento de Testes: PyTest para organização dos casos e execução automatizada.
- Automação de Testes: PyAutoGUI para simulação de interações na interface gráfica.
- Controle de Versão: Git para versionamento de scripts de teste automatizados.

7.3. Critérios de Aceitação

- Todos os casos de teste devem ser executados com sucesso, sem falhas críticas ou bloqueadoras.
- A interface deve estar em conformidade com o design funcional aprovado.
- O sistema deve atender aos requisitos de desempenho, respondendo as ações do usuário em tempo aceitável.

7.4. Ambiente de Teste

- Ambiente de Teste: Servidor de homologação com dados de teste pré-carregados.
- Ambiente de Produção: Servidor de produção para validação final antes do lançamento.

7.5. Cronograma

Atividade	Início	Término
Planejamento dos Testes	09/06/2025	10/06/2025
Preparação do Ambiente	11/06/2025	11/06/2025
Execução dos Testes	12/06/2025	16/06/2025
Relatório de Resultados	17/06/2025	17/06/2025

7.6. Responsabilidades

- Analista de QA: Elaboração e execução dos casos de teste, registro de defeitos.
- Desenvolvedor: Correção de defeitos identificados durante os testes.
- Gerente de Projeto: Acompanhamento do progresso dos testes e tomada de decisões estratégicas.

7.7. Riscos e Mitigações

Risco	Mitigação
Atraso na Disponibilização do Ambiente de Teste	Planejar com antecedência e solicitar o ambiente com prazo adequado.
Mudanças de Requisitos Durante os Testes	Estabelecer processo de controle de mudanças e avaliar impactos.

7.9. Aprovação

Este plano de teste será considerado aprovado mediante validação por todas as partes envolvidas e assinatura do documento.

7.10. Detalhamento de Testes

Caso de Teste 1 – Cadastro de Conta Válida

Entrada: Descrição = "Loja X", Valor = "150.00", Data = "20/06/2025"

Saída Esperada: Conta cadastrada com sucesso é exibida na listagem

Caso de Teste 2 – Campo Obrigatório em Branco

Entrada: Valor = "", Data = "20/06/2025"

Saída Esperada: Mensagem de erro "Campo Valor é obrigatório"

Caso de Teste 3 – Inserção de Valor Inválido

Entrada: Valor = "-100", Data = "20/06/2025"

Saída Esperada: Mensagem de erro "Valor não pode ser negativo"

8. Considerações Finais

Este documento consolidado proporciona uma visão abrangente do projeto de um sistema de Contas a Pagar, desde a concepção dos requisitos até a execução dos testes. A adoção de práticas tradicionais e ágeis visa garantir a qualidade, eficiência e aderência às necessidades dos usuários, promovendo uma solução estável, segura e funcional para controle financeiro pessoal.