

CENTRO ESTADUAL DE EDUCAÇÃO TECNOLÓGICA PAULA SOUZA  
FACULDADE DE TECNOLOGIA DE CAMPINAS  
CURSO SUPERIOR DE TECNOLOGIA EM ANÁLISE E  
**DESENVOLVIMENTO DE SISTEMAS**

MATHEUS BOSCO MARTINS DA SILVA

**DESENVOLVIMENTO DO FRONT-END DE UM  
APLICATIVO MOBILE PARA O CONTROLE DE FINANÇAS  
PESSOAIS**

CAMPINAS/SP  
2022

CENTRO ESTADUAL DE EDUCAÇÃO TECNOLÓGICA PAULA SOUZA  
FACULDADE DE TECNOLOGIA DE CAMPINAS  
CURSO SUPERIOR DE TECNOLOGIA EM ANÁLISE E  
**DESENVOLVIMENTO DE SISTEMAS**

MATHEUS BOSCO MARTINS DA SILVA

**DESENVOLVIMENTO DO FRONT-END DE UM  
APLICATIVO MOBILE PARA O CONTROLE DE FINANÇAS  
PESSOAIS**

Trabalho de Graduação apresentado por **Matheus Bosco Martins da Silva**, como pré-requisito para a conclusão do Curso Superior de Tecnologia em **Análise e desenvolvimento de sistemas**, da Faculdade de Tecnologia de Campinas, elaborado sob a orientação do Prof. **Anderson Luiz Barbosa**.

CAMPINAS/SP  
2022

**FICHA CATALOGRÁFICA**  
**CEETEPS - FATEC Campinas - Biblioteca**

**S586d**

**SILVA, Matheus Bosco Martins da**  
**Desenvolvimento do Front-End de um aplicativo mobile**  
**para o controle de finanças pessoais. Matheus Bosco**  
**Martins da Silva. Campinas, 2022.**

**45 p.; 30 cm.**

Trabalho de Graduação do Curso de Análise e Desenvolvimento de Sistema – Faculdade de Tecnologia de Campinas, 2022.

*Orientador: Prof. Me. Anderson Luiz Barbosa.*

1. Planejamento. 2. Finanças. 3. Tecnologia. 4. Aplicativos. 5. Interface.  
I. Autor. II. Faculdade de Tecnologia de Campinas. III. Título.

CDD 005.26

Catálogo-na-fonte: Bibliotecária: Aparecida Stradiotto Mendes – CRB8/6553

TG ADS 22.2

**Matheus Bosco Martins da Silva**

**Desenvolvimento do Front end de um aplicativo para o controle de  
finanças pessoais**

Trabalho de Graduação apresentado como exigência parcial  
para obtenção do título de Tecnólogo em Análise e  
Desenvolvimento de Sistemas, pelo CEETEPS / Faculdade  
de Tecnologia – Fatec Campinas.

Campinas, 02 de dezembro de 2022.

**BANCA EXAMINADORA**

*Anderson Luiz Barbosa*

---

Prof. Anderson Luiz Barbosa  
Fatec Campinas

*Cláudio Luis Vieira Oliveira*

---

Prof. Claudio Luis Vieira Oliveira  
Fatec Campinas

*Sandro Pinheiro de Assis Cosso*

---

Prof. Sandro Pinheiro de Assis Cosso  
Fatec Campinas

## **AGRADECIMENTOS**

A Deus pela vida, saúde e proteção durante todo este período de graduação.

A toda a minha família que não me deixou fraquejar nos momentos difíceis e por todo o apoio durante a minha jornada acadêmica.

Aos professores, pela paciência no processo de aprendizado e pelos ensinamentos que me permitiram apresentar um bom desempenho na minha formação profissional.

## RESUMO

O número de pessoas endividadas no Brasil segue quebrando recordes com o passar do tempo, portanto fazer o planejamento financeiro acaba se tornando a principal ferramenta quando o assunto se trata sobre o equilíbrio das finanças. Um problema que parece ser uma tarefa simples, porém manter uma consistência nesse planejamento, principalmente quando falamos à longo prazo pode se tornar complicado demais. Principalmente para as pessoas que possuem a famosa “*investment-phobia*”, ou seja pessoas que possuem dificuldades de falar sobre o dinheiro ou a até mesmo a falta dele. Para esclarecer esta problemática, neste estudo foi feita uma abordagem de uma maneira introdutória a respeito de educação financeira, além de mostrar os possíveis benefícios da tecnologia para o mapeamento de despesas e ganhos na vida monetária das pessoas, já que a popularização e acessibilidade, dos celulares e dispositivos mobile vem crescendo no Brasil e no mundo. Também foi feita uma pesquisa de mercado onde mostra os principais aplicativos para celular no meio financeiro, buscando entender quais seus pontos fracos e onde melhorias são necessárias. Com isso, o objetivo foi comprovar a possibilidade e desenvolver um aplicativo para tratar sobre o planejamento financeiro, com *front-end* simples, intuitivo, acessível e gratuito. Sobretudo com o objetivo de auxílio na diminuição da taxa das pessoas endividadas no Brasil.

**Palavras-chave:** planejamento, finanças, tecnologia, aplicativos e interface.

## **ABSTRACT**

The number of people in debt in Brazil continues to break records over time, so financial planning ends up becoming the main tool when it comes to balancing finances. A problem that seems to be a simple task, but maintaining consistency in this planning, especially when talking about the long term can become too complicated. Especially for people who have the famous “investment-phobia”, that is, people who have difficulty talking about money or even the lack of it. To clarify this problem, this study approached financial education in an introductory way, in addition to showing the possible benefits of technology for mapping expenses and gains in people's financial lives, since the popularization and accessibility of cell phones and mobile devices has been growing in Brazil and worldwide. A market survey was also carried out showing the main mobile applications in the financial environment, seeking to understand their weaknesses and where improvements are needed. With this, the objective was to verify the possibility of developing an application to deal with financial planning, with a simple, intuitive, accessible and free front-end. Above all, with the aim of helping to reduce the rate of indebted people in Brazil.

**Keywords:** planning, finance, technology, applications and interface.

## LISTA DE FIGURAS

<b>Figura 1</b> - UML - Diagrama de caso de uso: Login .....	23
<b>Figura 2</b> - UML - Diagrama de caso de uso: Visualização tela inicial.....	23
<b>Figura 3</b> - UML - Diagrama de caso de uso: Despesas .....	24
<b>Figura 4</b> - UML - Diagrama de caso de uso: Receitas .....	24
<b>Figura 5</b> - UML - Diagrama de caso de uso: Categorias de gasto.....	25
<b>Figura 6</b> - UML - Diagrama de caso de uso: Cartão de crédito .....	25
<b>Figura 7</b> - UML - Diagrama de caso de uso: Obejtivos .....	26
<b>Figura 8</b> - Protótipo: Tela de login .....	31
<b>Figura 9</b> - Protótipo: Tela principal .....	31
<b>Figura 10</b> - Protótipo: Adicionar despesas .....	32
<b>Figura 11</b> - Paleta de cores utilizada.....	32
<b>Figura 12</b> - Interface: Tela de login.....	33
<b>Figura 13</b> - Interface: Tela de cadastro.....	34
<b>Figura 14</b> - Interface: Tela de cadastro.....	34
<b>Figura 15</b> - Interface: Tela principal.....	35
<b>Figura 16</b> - Interface:Tela incial .....	35
<b>Figura 17</b> - Interface: Adicionar despesa .....	36
<b>Figura 18</b> - Interface: Adicionar receita .....	36
<b>Figura 19</b> - Interface: Tratamento de erro .....	37
<b>Figura 20</b> - Interface: Notificações.....	37
<b>Figura 21</b> – Reclamação da interface e usabilidade no aplicativo.....	38
<b>Figura 22</b> - Reclamação de interface .....	38
<b>Figura 23</b> - Reclamação de interface .....	39
<b>Figura 24</b> – Reclamação do aplicativo ao sair do gratuito para versão paga .....	39
<b>Figura 25</b> - Formulário de satisfação: Pergunta 01 .....	40
<b>Figura 26</b> - Formulário de satisfação: Pergunta 02 .....	40
<b>Figura 27</b> - Formulário de satisfação: Pergunta 03 .....	40
<b>Figura 28</b> - Formulário de satisfação: Pergunta 04 .....	41



## **LISTA DE QUADROS**

<b>Quadro 1</b> – Versão revisada das heurísticas de Nielsen.....	18
---	----

## **LISTA DE TABELAS**

<b>Tabela 1-</b> Descrição de requisitos.....	21
<b>Tabela 2-</b> Descrição do caso de uso.....	26
Tabela 3- Cronograma de etapas .....	43

## **LISTA DE ABREVIACES**

UML                    *Unified Modeling Language*

# SUMÁRIO

<b>1</b>	<b>INTRODUÇÃO</b>	12
1.1	CONTEXTUALIZAÇÃO	12
1.2	JUSTIFICATIVA/PROBLEMÁTICA	13
1.3	OBJETIVOS	13
<b>2</b>	<b>REVISÃO BIBLIOGRÁFICA</b>	14
<b>3</b>	<b>MATERIAIS E MÉTODOS</b>	17
3.1	MATERIAIS	17
3.1.1	DESCRIÇÃO DOS MATERIAIS	18
3.2	MÉTODOS	20
3.2.1	MODELAGEM	20
3.2.1.1	REQUISITOS	21
3.2.1.2	UML – DIAGRAMA DE CASO DE USO	22
3.2.1.3	DESCRIÇÃO DO DIAGRAMA DE CASO DE USO	26
3.2.2	INTERFACE	29
3.2.2.1	PROTÓTIPO	30
3.2.2.2	PALETA DE CORES E FONTE	32
3.2.2.3	TELAS DO APLICATIVO	33
<b>4</b>	<b>RESULTADOS E DISCUSSÃO</b>	38
<b>5</b>	<b>CONSIDERAÇÕES FINAIS</b>	42
	<b>REFERÊNCIAS BIBLIOGRÁFICAS</b>	44

# 1 INTRODUÇÃO

Finanças pessoais sempre foram um desafio para o brasileiro, desde contabilizar as moedas para ir comprar itens na mercearia, até o pagamento de boletos e contas em um banco ou lotérica. A administração financeira invariavelmente foi algo em que as pessoas deveriam sempre se manter informadas, seja por dicas de amigos, parentes, livros ou até mesmo (num contexto mais atual) em cursos via internet.

Também chamada de a “matéria esquecida” principalmente no ensino público, a sua ausência fez com que pessoas de origem humilde sempre obtivessem dificuldades e obstáculos em relação a vida monetária. Já que grande parte dos indivíduos que possuem o conhecimento de finanças de forma aprimorada, possuem geralmente famílias com poderes aquisitivos significativos ou já com um histórico e hábitos que proporcionam este aprendizado desde jovem.

Como esclarecido por Calixto *et al* (2022, p. 2), “O assunto dinheiro ainda é um tabu muito grande no Brasil. A falta de conversa sobre o tema em casa e nas escolas acarretam vários problemas na vida adulta do ser humano como a inadimplência e problemas psicológicos”.

Com este cenário em vista, e com a atual situação financeira das pessoas no Brasil, é de interesse coletivo uma forma simples, eficaz e dinâmica de gerenciamento de suas finanças pessoais. Portanto, este trabalho de graduação irá tratar do desenvolvimento do *front-end* de um aplicativo *mobile* para celulares chamado “Myfinanz”, que irá realizar toda a parte administrativa e monitoramento das finanças para todo e qualquer usuário.

## 1.1 CONTEXTUALIZAÇÃO

No Brasil, quase 78% da população está endividada, segundo uma pesquisa feita em março de 2022 pela confederação Nacional do Comércio de Bens. Os números são impactantes, e evidenciam um assunto que a grande maioria dos brasileiros possui dificuldade ou até mesmo não dão devida atenção... O planejamento financeiro.

Muitas vezes, as pessoas acabam se perdendo nas próprias finanças e quando menos esperam estão com a conta “no vermelho”. O planejamento financeiro é um item essencial para que não haja surpresas ou até mesmo necessidades durante o mês. Porém anotar toda a

movimentação monetária de uma pessoa, seja em um caderno ou folha de papel não parece ser uma tarefa fácil ou segura. Por conta disso, utilizar a tecnologia para auxiliar neste problema se tornou uma opção viável, prática e acessível, já que no Brasil para cada um habitante existem dois dispositivos digitais como celulares ou computadores.

## 1.2 JUSTIFICATIVA/PROBLEMÁTICA

O planejamento financeiro de uma pessoa pode ser feito de diversas formas, seja ela através de meios tradicionais como fazer as anotações em um caderno, por exemplo. Até mesmo utilizando um aplicativo, site ou planilha que visam descomplicar todo o processo.

Contudo, apesar destas diversas alternativas tecnológicas, muitas delas podem parecer complexas de mais para usuários que não possuem uma afinidade tão grande com um celular ou computador. Fazendo com que as pessoas abandonem o processo mesmo antes de iniciar seus planejamentos e metas. Além destas lacunas que geram uma oportunidade de projeto, existe uma grande motivação do autor, que mesmo possuindo facilidade com a tecnologia, passou por diversas dificuldades utilizando aplicativos e sites existentes.

Neste contexto, foi desenvolvido um aplicativo *mobile* com um *front-end* simples e intuitivo, que visa facilitar a organização financeira de seus usuários e incentive os mesmos a introduzir esta nova rotina de planejar o uso do dinheiro em seu cotidiano. O trabalho teve uma grande influência das 10 heurísticas de Nielsen, com a finalidade de atribuir um visual funcional e agradável para a aplicação. Com essas características o aplicativo pode ser um aliado para a diminuição das pessoas endividadas no Brasil e ser útil para as pessoas que estão iniciando sua vida financeira.

## 1.3 OBJETIVOS

- A) Analisar o cenário da educação financeira e aspectos sociológicos econômicos de maneira breve, com caráter introdutório.
- B) Identificar as dificuldades que as pessoas possuem ao utilizar um aplicativo financeiro.
- C) Desenvolver uma interface simples, intuitiva e gratuita, que possibilite o planejamento das finanças do usuário.

## 2 REVISÃO BIBLIOGRÁFICA

A dificuldade de planejamento financeiro do brasileiro, não é nenhuma novidade nos tempos atuais, uma problemática tão antiga e presente no dia a dia da população, carece de alternativas simples e eficazes para a resolução desta adversidade. Como consequência deste grave problema, o Brasil segue aumentando de uma maneira exponencial a porcentagem dos devedores. Segundo pesquisa publicada em março de 2022 pelo G1: “77,5% das famílias brasileiras fecharam o mês com alguma dívida, segundo levantamento da Confederação Nacional do Comércio de Bens, Serviços e Turismo (CNC).”

O planejamento da utilização do capital é um do forte candidato para evitar que a falta de dinheiro não surpreenda no decorrer do mês. Na teoria a regra “gaste menos do que você ganha” parece ser simples, porém na prática, aparecem as dificuldades, principalmente no momento de mapear todos os gastos, que possibilita entender como reduzir as despesas ou até mesmo onde aumentar a aplicação do capital.

O orçamento doméstico é o principal instrumento para fazer o planejamento financeiro para hoje, amanhã e dias futuros. É utilizado como ferramenta para planejar o equilíbrio entre receitas e despesas das contas pessoais, mas para isso o indivíduo deverá se comprometer com o projeto de estruturação do orçamento pessoal e disposto a colaborar para que o resultado final seja atingido. (MOTA; CARLA *et al*, p.8,2015)

Dito isso, uma justificativa plausível para tal dificuldade, seria o medo de lidar com a vida financeira. Tal medo também pode ser conhecido como “*Investment Phobia*”, título de uma pesquisa realizada pela universidade de Cambridge em 2003, onde afirma que o temor de lidar com a falta de dinheiro e falar sobre o assunto, são o terceiro maior medo das pessoas no Reino Unido, onde uma a cada cinco pessoas possuem esta fobia.

Tendo em vista tais dificuldades, um fator que poderá contribuir e facilitar o planejamento financeiro, é a tecnologia, já que grande parte das pessoas possuem um dispositivo móvel como tablets ou celulares em suas casas. Através destes aparelhos, é possível criar aplicativos para que esta gestão de capital seja facilmente acessada e armazenada. Além de auxiliar na quebra deste tipo de fobia, através de interfaces lúdicas e amigáveis, além da introdução de novos hábitos na rotina dos utilizadores.

Os aplicativos, através dos dispositivos móveis portáteis, têm facilitado a vida das pessoas, ampliando o impacto da tecnologia na sociedade, permitindo a expansão da portabilidade e flexibilidade para vários tipos de relações, como um meio de automatização de diversas tarefas, até então realizadas manualmente. (MAIA e MARIN, p.2, 2021)

Contudo, apesar de existirem diversas plataformas onde é possível fazer o planejamento através do celular ou computador, sistemas com funcionalidades gratuitas e simples de serem

utilizadas são escassas no mercado. Muitas vezes os usuários com pouca familiaridade com tecnologia, possuem extrema dificuldade em operar os aplicativos e não possuem o devido apoio, como tutoriais por exemplo. Já que uma interface complexa para usuários iniciantes, pode acabar afastando ao invés de atrair seu utilizador. Segundo Nielsen e Loranger (2006), a usabilidade permite que pessoas com baixo grau de instrução possam ter empregos significativos, ela também permite conectar idosos com a comunidade e até mesmo fornecer informações e serviços a usuários portadores de deficiência física da mesma maneira que fornecemos a todas as outras pessoas. Para eles, a usabilidade beneficia os negócios e a humanidade.

Em vista destes fatores, foi desenvolvido um aplicativo para dispositivos *mobile* em conjunto com o aluno Vinicius Solon Silva, no qual desenvolveu o *back-end* da aplicação. O aplicativo “Myfinanz” conta com uma interface simples, intuitiva e acessível para pessoas com deficiências visuais, onde possibilita que o utilizador possa lidar com seu próprio dinheiro, podendo fazer planejamentos para os mais diferentes objetivos de uma forma ágil. Sobretudo o objetivo é contribuir para a diminuição da população endividada no país.

Para o desenvolvimento do *front-end*, foi utilizada a linguagem de programação *Python*, tendo em vista que é uma linguagem poderosa e versátil no mercado. “O *python* tem uma sintaxe clara e concisa que favorece a legibilidade do código-fonte, tornando a linguagem mais produtiva” (Borges, p.14, 2014). Dito isso, a linguagem possui um *framework* chamado *KivyMD* que possibilita o desenvolvimento de interfaces gráficas para dispositivos *mobile*. O *framework* apesar de ainda estar em desenvolvimento, tem como objetivo se aproximar ao máximo ao *Material design* do Google, contando com um bom desempenho e facilidade no seu uso. O *KivyMD* possui uma coleção de *widgets*, totalmente customizáveis e compatíveis com a estrutura gráfica *Kivy*.

Com o objetivo de desenvolver a aplicação dentro do prazo esperado e com as funcionalidades completas, foi utilizado a metodologia ágil *Scrum*. Segundo Michael (2018), apesar das implementações das metodologias ágeis serem recentes no universo do desenvolvimento de sistemas, o resultado de imediato já é muito satisfatório. Primeiramente ocorreu um pré-planejamento, onde foram definidos os objetivos das *sprints* iniciais juntamente com alocação de recursos que estarão disponíveis para o desenvolvimento do aplicativo. No momento de desenvolvimento, ocorreram reuniões diárias com um pequeno período de tempo, para que fossem esclarecidos todos os pontos que estão sendo trabalhados e suas dificuldades. Um dos benefícios do *scrum* para o projeto, foi a flexibilização da mudança de direção do



projeto, ou seja, nos casos em que foram necessárias alterações de requisitos ou de tecnologias no decorrer do desenvolvimento, os mesmos tiveram a possibilidade de serem aceitos.

### 3 MATERIAIS E MÉTODOS

A metodologia utilizada para o desenvolvimento teórico deste trabalho foi a pesquisa e fichamento de diversos artigos e estudos divulgados amplamente pela internet, como referência para a criação de um plano de desenvolvimento do aplicativo. Também foi feita uma pesquisa de mercado, onde foram analisados aplicativos para finanças já existentes, e pontuados os pontos que necessitam de melhoria.

Além disso, foi iniciada uma pesquisa sobre as possíveis tecnologias a serem utilizadas na criação do aplicativo, permitindo assim o aprofundamento dos estudos práticos com cada tecnologia escolhida. Inicialmente o material escolhido, para o *Front-end* foi a plataforma de serviços *Power Apps* da Microsoft, pois possui uma grande popularidade e dinamicidade de uso e desenvolvimento para plataformas *mobile*. Porém, após algumas pesquisas mais aprofundadas, foi constatado que o *framework KivyMD* juntamente com a linguagem *Python*, seriam as tecnologias mais adequadas para o cenário. O *software* desenvolvido, é capaz de ser executado até o momento no sistema operacional *Android*.

#### 3.1 MATERIAIS

Para a modelagem dos diagramas e requisitos, foi utilizada a plataforma LucidChart juntamente com a plataforma Drawio. Para prototipação das telas, foi utilizada a ferramenta Figma.

Além dessas, para o desenvolvimento do *front-end* do projeto, foi usada a linguagem de programação *Python* em conjunto com o *framework KivyMD*, que possibilita a criação de aplicativos *mobile* baseado na linguagem *Python*. A plataforma escolhida para ser executado os testes será o ambiente do sistema operacional *Android*.

Também serão utilizadas as Heurísticas de Nielsen para a implementação e validação das usabilidades do aplicativo.

### 3.1.1 DESCRIÇÃO DOS MATERIAIS

Para a modelagem dos requisitos e diagramas, foram utilizadas as plataformas de criação e edição gráfica *Lucidchart* e *Draw.io*, ambas são muito parecidas, contam com uma biblioteca grande de símbolos e notações para os mais diversos tipos de diagramas, como diagramas UML, banco de dados, diagrama de classe, máquina de estado, entre outros. É possível criar quadros e editá-los, tudo isso de uma forma gratuita.

Para a prototipação das telas do aplicativo, foi utilizada a ferramenta *Figma*, que possibilita a edição do design, onde possui uma versão web completa, com inúmeras ferramentas. Além de fazer a emulação deste protótipo com navegação de telas, entre outras funcionalidades.

Neste projeto, foram implementadas as Heurísticas de Nielsen, na quais consistem em um método no qual é conduzido por um avaliador ou um conjunto de avaliadores que examinam o ambiente gráfico (interface gráfica) e julgam suas propriedades com o auxílio de princípios de usabilidade, denominados heurísticas (Quadro 1). (ROCHA; BARANAUSKAS, 2000).

*Quadro 1 – Versão revisada das heurísticas de Nielsen*

<p><b>1. Visibilidade do status do sistema</b></p> <ul style="list-style-type: none"> <li>• sistema precisa manter os usuários informados sobre o que está acontecendo, fornecendo um feedback adequado dentro de um tempo razoável.</li> </ul>
<p><b>2. Compatibilidade do sistema com o mundo real</b></p> <ul style="list-style-type: none"> <li>• sistema precisa falar a linguagem do usuário, com palavras, frases e conceitos familiares ao usuário, ao invés de termos orientados ao sistema. Seguir convenções do mundo real, fazendo com que a informação apareça numa ordem natural e lógica.</li> </ul>
<p><b>3. Controle do usuário e liberdade</b></p> <ul style="list-style-type: none"> <li>• usuários frequentemente escolhem por engano funções do sistema e precisam ter claras saídas de emergência para sair do estado indesejado sem ter que percorrer um extenso diálogo. Prover funções <i>undo</i> e <i>redo</i>.</li> </ul>
<p><b>4. Consistência e padrões</b></p>

<ul style="list-style-type: none"> <li>usuários não precisam adivinhar que diferentes palavras, situações ou ações significam a mesma coisa. Seguir convenções de plataforma computacional.</li> </ul>
<p><b>5. Prevenção de erros</b></p> <ul style="list-style-type: none"> <li>melhor que uma boa mensagem de erro é um design cuidadoso o qual previne o erro antes dele acontecer.</li> </ul>
<p><b>6. Reconhecimento ao invés de relembração</b></p> <ul style="list-style-type: none"> <li>tornar objetos, ações e opções visíveis. O usuário não deve ter que lembrar informação de uma para outra parte do diálogo. Instruções para uso do sistema devem estar visíveis e facilmente recuperáveis quando necessário.</li> </ul>
<p><b>7. Flexibilidade e eficiência de uso</b></p> <ul style="list-style-type: none"> <li>usuários novatos se tornam peritos com o uso. Prover aceleradores de formar a aumentar a velocidade da interação. Permitir a usuários experientes "cortar caminho" em ações frequentes.</li> </ul>
<p><b>8. Estética e design minimalista</b></p> <ul style="list-style-type: none"> <li>diálogos não devem conter informação irrelevante ou raramente necessária. Qualquer unidade de informação extra no diálogo irá competir com unidades relevantes de informação e diminuir sua visibilidade relativa.</li> </ul>
<p><b>9. Ajudar os usuários a reconhecer, diagnosticar e corrigir erros</b></p> <ul style="list-style-type: none"> <li>mensagens de erro devem ser expressas em linguagem clara (sem códigos) indicando precisamente o problema e construtivamente sugerindo uma solução.</li> </ul>
<p><b>10. Help e documentação</b></p> <ul style="list-style-type: none"> <li>embora seja melhor um sistema que possa ser usado sem documentação, é necessário prover <i>help</i> e documentação. Essas informações devem ser fáceis de encontrar, focalizadas na tarefa do usuário e não muito extensas.</li> </ul>

## 3.2 MÉTODOS

Para o desenvolvimento do projeto, foram utilizados os conhecimentos demonstrados nas aulas da disciplina engenharia de software no decorrer do curso. Foi necessário iniciar o projeto em um estágio conceitual, onde foram desenvolvidos os requisitos e toda a modelagem através de diagramas UML feitos através da plataforma *Lucidchart*. Após a modelagem, foi elaborado o protótipo de baixa fidelidade, através da plataforma *Figma*. O protótipo auxiliou na montagem e design de cada tela.

Logo em seguida, se deu início ao desenvolvimento do *front-end*, através da linguagem de programação Python em conjunto com o framework *KivyMD* para a criação da interface. Para o ambiente de desenvolvimento, foi utilizado editor de código *Visual Studio Code*, pois conta com um suporte para depuração de código, versionamento *Git*, realce de sintaxe e uma vasta biblioteca de extensões que podem auxiliar durante o desenvolvimento das telas do aplicativo Myfinaz.

O projeto foi desenvolvido com a metodologia ágil Scrum, já que após a produção das telas, o *back-end* do aplicativo foi implementado pelo aluno Vinícius Solon Silva. Através desta metodologia ágil, foi possível uma melhor organização, estabelecimento dos prazos e objetivos, para que as sprints pudessem ser concluídas paralelamente de uma forma organizada.

Para o versionamento do código, foram utilizadas as plataformas *Git* e *Github*, nas quais auxiliaram no desenvolvimento contribuído e dinâmico do aplicativo. Também Vale ressaltar que todas as partes passaram por testes após suas implementações

### 3.2.1 MODELAGEM

Na grande maioria dos processos de engenharia de software, o processo inicial que auxilia na clareza de ideias, é a modelagem. Através da modelagem é possível enxergar o sistema como um todo e planejar onde será o início do desenvolvimento do programa. Segundo Sommerville (2011), os melhores modelos são aqueles que são mais simples, que deixam de lado os detalhes, para assim extrair as características mais salientes do projeto, tendo em mente que um modelo é uma abstração do sistema final.

Portanto no projeto de desenvolvimento do *front-end* do aplicativo Myfinaz, a modelagem foi desenvolvida em três etapas: Requisitos, diagrama de caso de uso e a descrição

do diagrama de caso de uso. Que foram fundamentais no processo de consolidação das ideias, sempre visando o objetivo de suprir as metas mostradas previamente.

### 3.2.1.1 REQUISITOS

Primordialmente, foram estabelecidos todos os requisitos que o sistema teria e quais necessidades o aplicativo iria suprir. Foi de suma importância a definição destes pontos logo de início, pois auxiliou a consolidar as ideias e direcionar a prioridade do desenvolvimento.

“Os requisitos de um sistema são as descrições do que o sistema deve fazer, os serviços que oferece e as restrições ao seu funcionamento. Esses requisitos refletem as necessidades dos clientes para um sistema que serve a uma finalidade determinada, como controlar um dispositivo, colocar um pedido ou encontrar informações” (Sommerville, p.71, 2011)

Dito isso, os requisitos funcionais e não funcionais, estão descritos na tabela abaixo:

*Tabela 1- Descrição de requisitos*

Requisitos Funcionais		Requisitos não Funcionais	
<b>RF01</b>	Fazer login	<b>RNF01</b>	Manutenção do aplicativo
<b>RF02</b>	Fazer Cadastro	<b>RNF02</b>	<i>Front-End em Python</i>
<b>RF03</b>	Adicionar despesas	<b>RNF03</b>	Front-End com o framework KivyMD
<b>RF04</b>	Categorizar despesas	<b>RNF04</b>	Interface simples e intuitiva
<b>RF05</b>	Indicar se a despesa já foi paga	<b>RNF05</b>	Acessível para pessoas com menos habilidade com tecnologia
<b>RF06</b>	Adicionar receita	<b>RNF06</b>	Interface responsiva
<b>RF07</b>	Categorizar receitas	<b>RNF07</b>	Padronização de cores
<b>RF08</b>	Adicionar cartões de crédito	<b>RNF08</b>	Cores com contraste entre si, para auxílio de pessoas com visão prejudicada
<b>RF09</b>	Visualizar o total a pagar do cartão de crédito	<b>RNF09</b>	Padronização do tamanho das fontes
<b>RF10</b>	Visualizar saldo atual da conta	<b>RNF10</b>	Padronização dos ícones
<b>RF11</b>	Visualizar despesas pendentes	<b>RNF11</b>	Visualizar status do aplicativo
<b>RF12</b>	Visualizar histórico de despesas por categoria	<b>RNF12</b>	Estética agradável
<b>RF13</b>	Visualizar histórico de receitas	<b>RNF13</b>	Design minimalista
<b>RF14</b>	Adicionar novas categorias de gasto	<b>RNF14</b>	Satisfazer as 10 heurísticas de Nielsen
<b>RF15</b>	Traçar objetivos		
<b>RF16</b>	Estabelecer limites para cada categoria de gasto		
<b>RF17</b>	Visualização de notificações		

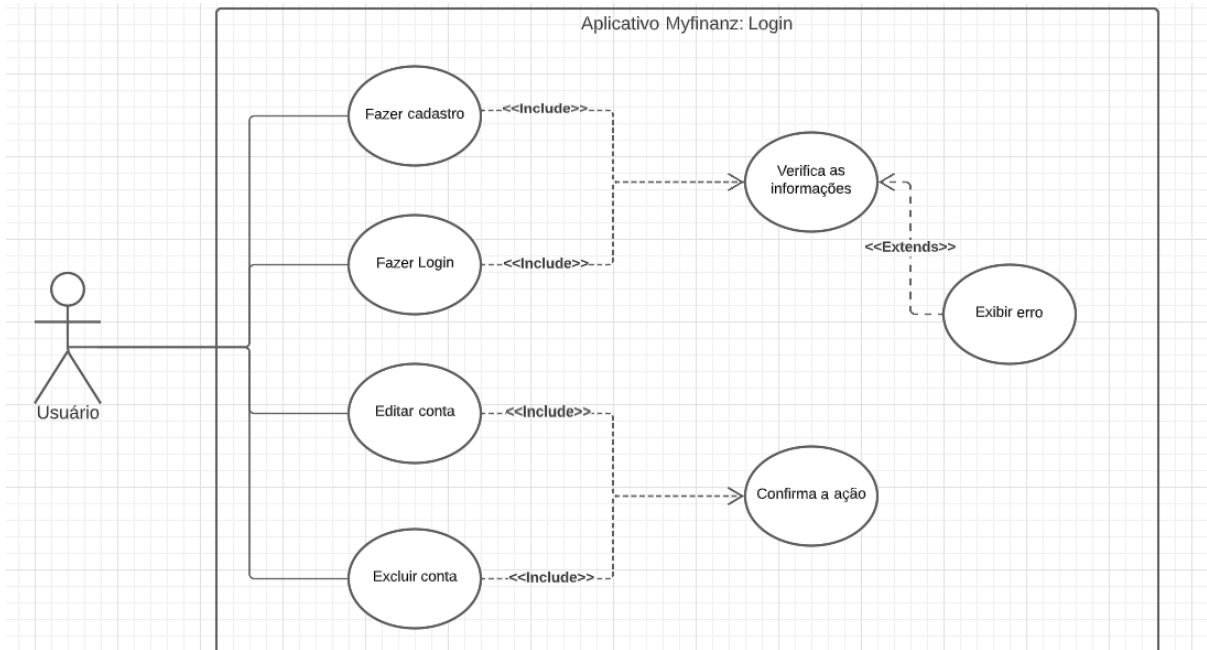
<b>RF18</b>	Editar conta		
<b>RF19</b>	Excluir conta		
<b>RF20</b>	Visualização de tutoriais para usar o aplicativo		
<b>RF21</b>	Excluir despesa		
<b>RF22</b>	Excluir receita		
<b>RF23</b>	Editar despesa		
<b>RF24</b>	Editar receita		
<b>RF25</b>	Editar cartão		
<b>RF26</b>	Excluir cartão		
<b>RF27</b>	Editar objetivo		
<b>RF28</b>	Excluir objetivo		

### 3.2.1.2 UML – DIAGRAMA DE CASO DE USO

Após o desenvolvimento dos requisitos, a etapa seguinte foi o desenvolvimento do diagrama de caso de uso, através da linguagem UML. Onde ajudou a exemplificar principalmente a ação dos atores com o sistema, e como o aplicativo iria responder tais ações dos usuários. Segundo Booch *et al* (p.13, 2005), a *Unified Modeling Language* (UML) é uma linguagem padrão para a visualização, especificação, construção e documentação da estrutura de projetos de software. Uma das vantagens de construir um diagrama deste modelo, foi a possibilidade de construir um modelo que permitisse de fato como a aplicação final iria ser programada.

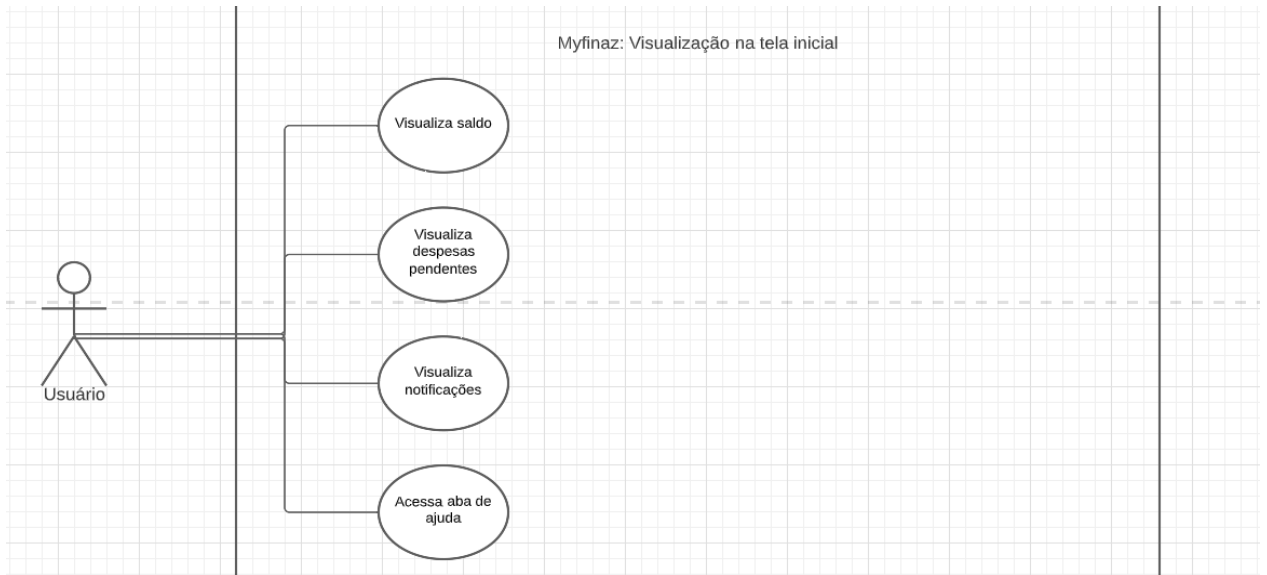
No modelo construído foi incluído um ator, cujo nome é “usuário”. O diagrama visa exemplificar as ações possíveis de um utilizador dentro do aplicativo Myfinanz. Para melhorar a visualização neste documento, o diagrama completo foi fracionado em algumas partes como ilustrados da figura 1 até a figura 7.

Figura 1 - UML - Diagrama de caso de uso: Login



Fonte: [https://drive.google.com/file/d/16\\_7gbEuHXtZhRZeGRnJQ9rZIMxAA5E4/view?usp=share\\_link](https://drive.google.com/file/d/16_7gbEuHXtZhRZeGRnJQ9rZIMxAA5E4/view?usp=share_link)

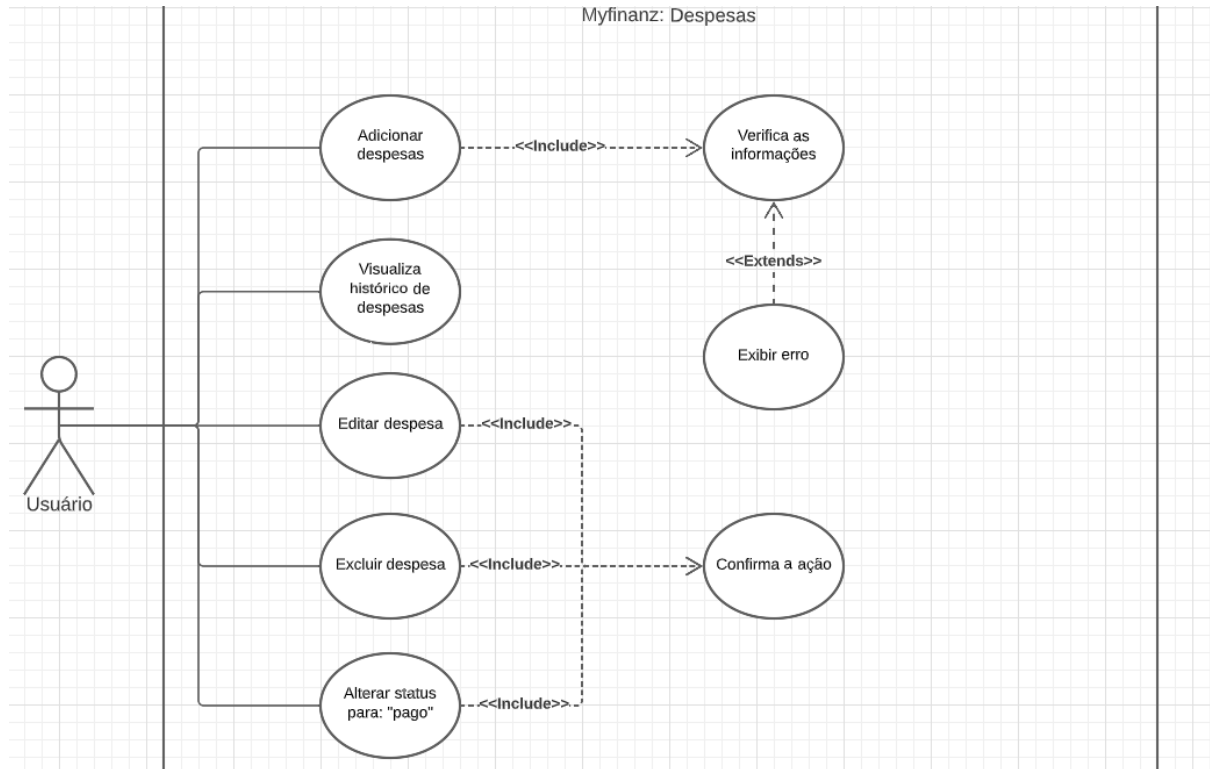
Figura 2 - UML - Diagrama de caso de uso: Visualização tela inicial



Fonte: [https://drive.google.com/file/d/16\\_7gbEuHXtZhRZeGRnJQ9rZIMxAA5E4/view?usp=share\\_link](https://drive.google.com/file/d/16_7gbEuHXtZhRZeGRnJQ9rZIMxAA5E4/view?usp=share_link)

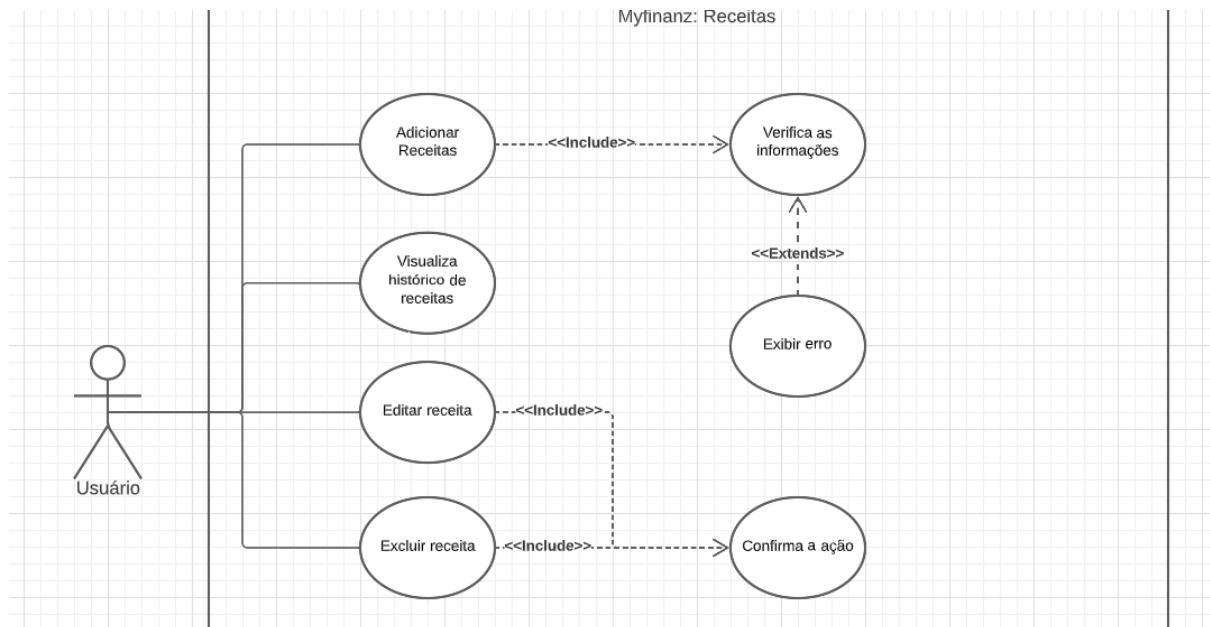


Figura 3 - UML - Diagrama de caso de uso: Despesas



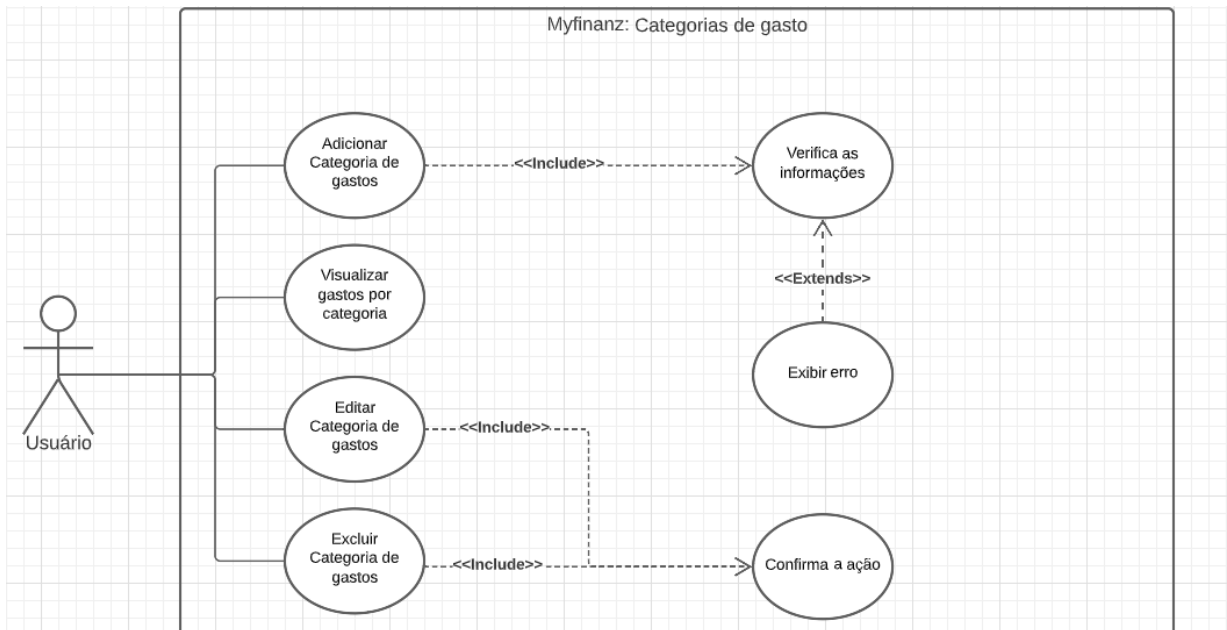
Fonte: [https://drive.google.com/file/d/16\\_7gbEuHXtZhrZeGRnJQ9rZIMxAA5E4/view?usp=share\\_link](https://drive.google.com/file/d/16_7gbEuHXtZhrZeGRnJQ9rZIMxAA5E4/view?usp=share_link)

Figura 4 - UML - Diagrama de caso de uso: Receitas



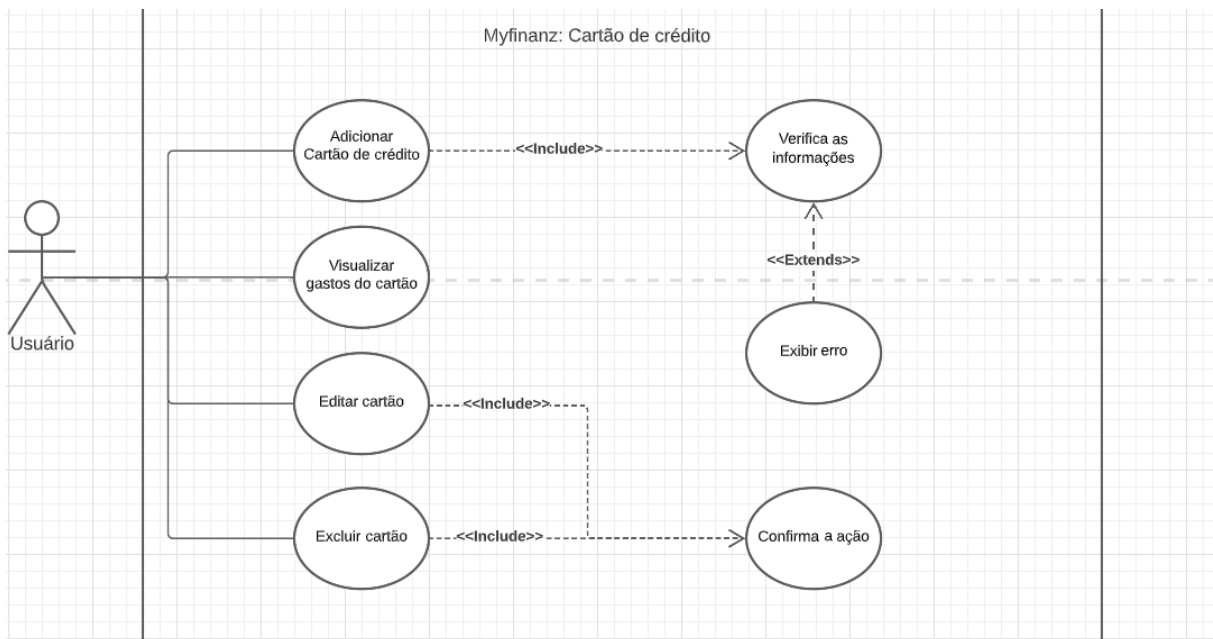
Fonte: [https://drive.google.com/file/d/16\\_7gbEuHXtZhrZeGRnJQ9rZIMxAA5E4/view?usp=share\\_link](https://drive.google.com/file/d/16_7gbEuHXtZhrZeGRnJQ9rZIMxAA5E4/view?usp=share_link)

Figura 5 - UML - Diagrama de caso de uso: Categorias de gasto



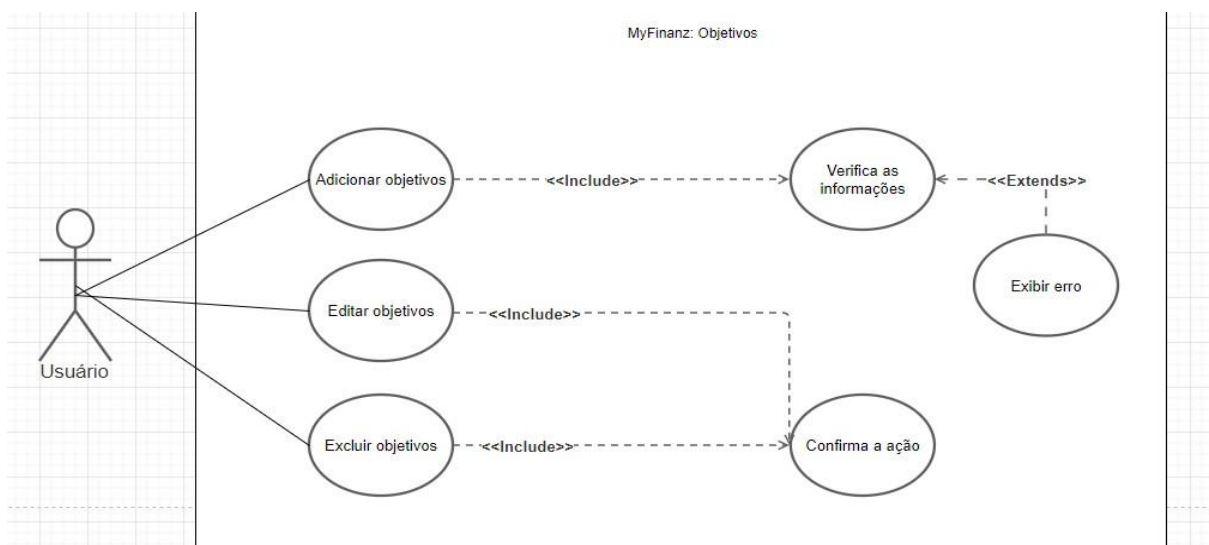
Fonte: [https://drive.google.com/file/d/16\\_7gbEuHXtZhrZeGRnJQ9rZIMxAA5E4/view?usp=share\\_link](https://drive.google.com/file/d/16_7gbEuHXtZhrZeGRnJQ9rZIMxAA5E4/view?usp=share_link)

Figura 6 - UML - Diagrama de caso de uso: Cartão de crédito



Fonte: [https://drive.google.com/file/d/16\\_7gbEuHXtZhrZeGRnJQ9rZIMxAA5E4/view?usp=share\\_link](https://drive.google.com/file/d/16_7gbEuHXtZhrZeGRnJQ9rZIMxAA5E4/view?usp=share_link)

Figura 7 - UML - Diagrama de caso de uso: Obejtivos



Fonte: [https://drive.google.com/file/d/16\\_7gbEuHXtZhRZeGRnJQ9rZIMxAA5E4/view?usp=share\\_link](https://drive.google.com/file/d/16_7gbEuHXtZhRZeGRnJQ9rZIMxAA5E4/view?usp=share_link)

### 3.2.1.3 DESCRIÇÃO DO DIAGRAMA DE CASO DE USO

Tabela 2- Descrição do caso de uso

Caso de uso	Referência	Pré-condições	Fluxo básico
Fazer cadastro	RF02	Aplicativo baixado no dispositivo e conexão com a internet	1 – Usuário entra na tela de cadastro 2 – Preenche os campos corretamente 3 – Pressiona o botão “cadastrar”
Fazer Login	RF01	Aplicativo baixado no dispositivo, conexão com a internet e conta previamente criada	1 – Usuário entra na tela de login 2 – Preenche os campos corretamente 3 – Pressiona o botão “entrar”
Editar conta	RF18	Aplicativo baixado no dispositivo, conexão com a internet e conta previamente criada	1 – Usuário faz login 2 – Entra na tela de ajustes 3 – Edita a informação que deseja 4 – Confirma a alteração
Excluir conta	RF19	Aplicativo baixado no dispositivo, conexão com a internet e conta previamente criada	1 – Usuário faz login 2 – Entra na tela de ajustes 3 – Pressiona o botão “Excluir conta” 4 – Confirma a alteração
Visualizar saldo	RF10	Aplicativo baixado no dispositivo, conexão com a	1 – Usuário faz login 2 – Visualiza na tela principal

		internet e conta previamente criada	
Visualizar despesas pendentes	RF11	Aplicativo baixado no dispositivo, conexão com a internet, conta previamente criada e adicionar as despesas previamente	1 – Usuário faz login 2 – Visualiza na tela principal
Visualiza notificações	RF17	Aplicativo baixado no dispositivo, conexão com a internet e conta previamente criada	1 – Usuário faz login 2 – Pressiona o ícone de notificações
Acessa aba de ajuda	RF20	Aplicativo baixado no dispositivo, conexão com a internet e conta previamente criada	1 – Usuário faz login 2 – Pressiona o ícone de ajustes
Adicionar despesas	RF03, RF05	Aplicativo baixado no dispositivo, conexão com a internet e conta previamente criada	1 – Usuário faz login 2 – Pressiona o botão “Adicionar despesas” 3 – Preenche os campos corretamente 4 – Pressiona o botão “Adicionar”
Visualizar histórico de despesas	RF11, RF12	Aplicativo baixado no dispositivo, conexão com a internet, conta previamente criada e adicionar as despesas previamente	1 – Usuário faz login 2 – Pressiona o card de despesas
Editar despesa	RF05, RF23	Aplicativo baixado no dispositivo, conexão com a internet, conta previamente criada e adicionar as despesas previamente	1 – Usuário faz login 2 – Pressiona o card de despesas 3 – Identifica a despesa 4 – Aperta o ícone de editar 5 – Confirma a alteração
Excluir Despesa	RF21	Aplicativo baixado no dispositivo, conexão com a internet, conta previamente criada e adicionar as despesas previamente	1 – Usuário faz login 2 – Pressiona o card de despesas 3 – Identifica a despesa 4 – Aperta o ícone de excluir 5 – Confirma a alteração
Alterar status para: “pago”	RF05, RF23	Aplicativo baixado no dispositivo, conexão com a internet, conta previamente criada e adicionar as despesas previamente com status de não pago.	1 – Usuário faz login 2 – Pressiona o card de despesas 3 – Identifica a despesa 4 – Aperta o ícone de editar 5 – Confirma a alteração
Adicionar receita	RF06	Aplicativo baixado no dispositivo, conexão com a internet e conta previamente criada	1 – Usuário faz login 2 – Pressiona o botão “Adicionar receitas” 3 – Preenche os campos corretamente

			4 – Pressiona o botão “Adicionar”
Visualizar histórico de receita	RF07, RF13	Aplicativo baixado no dispositivo, conexão com a internet, conta previamente criada e adicionar as receitas previamente	1 – Usuário faz login 2 – Pressiona o card de receitas
Editar receita	RF24	Aplicativo baixado no dispositivo, conexão com a internet, conta previamente criada e adicionar as receitas previamente	1 – Usuário faz login 2 – Pressiona o card de receita 3 – Identifica a receita 4 – Aperta o ícone de editar 5 – Confirma a alteração
Excluir receita	RF22	Aplicativo baixado no dispositivo, conexão com a internet, conta previamente criada e adicionar as receitas previamente	1 – Usuário faz login 2 – Pressiona o card de receitas 3 – Identifica a receita 4 – Aperta o ícone de excluir 5 – Confirma a alteração
Adicionar categoria de gastos	RF04, RF14	Aplicativo baixado no dispositivo, conexão com a internet e conta previamente criada	1 – Usuário faz login 2 – Pressiona o botão “categorias” 3 – Preenche os campos corretamente 4 – Pressiona o botão “Adicionar”
Visualizar gastos por categoria	RF12	Aplicativo baixado no dispositivo, conexão com a internet, conta previamente criada, adicionar as despesas previamente e adicionar categorias previamente	1 – Usuário faz login 2 – Pressiona o card de gastos por categoria
Editar categoria de gastos	RF14, RF16	Aplicativo baixado no dispositivo, conexão com a internet, conta previamente criada, adicionar as despesas previamente e adicionar categorias previamente	1 – Usuário faz login 2 – Pressiona o card de categorias 3 – Identifica a categoria 4 – Aperta o ícone de editar 5 – Confirma a alteração
Excluir categoria de gastos	RF14, RF16	Aplicativo baixado no dispositivo, conexão com a internet, conta previamente criada, adicionar as despesas previamente e adicionar categorias previamente	1 – Usuário faz login 2 – Pressiona o card de categorias 3 – Identifica a categoria 4 – Aperta o ícone de excluir 5 – Confirma a alteração
Adicionar Cartão de crédito	RF08	Aplicativo baixado no dispositivo, conexão com a internet e conta previamente criada	1 – Usuário faz login 2 – Pressiona o botão “cartões” 3 – Preenche os campos corretamente 4 – Pressiona o botão “Adicionar”

Visualizar gastos de cartão	RF09	Aplicativo baixado no dispositivo, conexão com a internet, conta previamente criada, cartão previamente criado e adicionar as despesas previamente na categoria cartão	1 – Usuário faz login 2 – Pressiona o botão “cartões”
Editar cartão	RF25	Aplicativo baixado no dispositivo, conexão com a internet, conta previamente criada e cartão previamente criado	1 – Usuário faz login 2 – Pressiona o botão “cartões” 3 – Identifica o cartão 4 – Aperta o ícone de editar 5 – Confirma a alteração
Excluir cartão	RF26	Aplicativo baixado no dispositivo, conexão com a internet, conta previamente criada e cartão previamente criado	1 – Usuário faz login 2 – Pressiona o botão “cartões” 3 – Identifica o cartão 4 – Aperta o ícone de excluir 5 – Confirma a alteração
Adicionar Objetivo	RF15	Aplicativo baixado no dispositivo, conexão com a internet e conta previamente criada	1 – Usuário faz login 2 – Pressiona o botão “objetivos” 3 – Preenche os campos corretamente 4 – Pressiona o botão “Adicionar”
Editar objetivo	RF27	Aplicativo baixado no dispositivo, conexão com a internet, conta previamente criada e objetivo criado	1 – Usuário faz login 2 – Pressiona o botão “objetivos” 3 – Identifica o objetivo 4 – Aperta o ícone de editar 5 – Confirma a alteração
Excluir objetivo	RF28	Aplicativo baixado no dispositivo, conexão com a internet, conta previamente criada e objetivo criado	1 – Usuário faz login 2 – Pressiona o botão “objetivos” 3 – Identifica o objetivo 4 – Aperta o ícone de excluir 5 – Confirma a alteração

### 3.2.2 INTERFACE

Com o término da parte conceitual do aplicativo, ocorreu o início do tema central do projeto, o desenvolvimento do *front-end*. Este processo teve como principal fonte de conhecimento as 10 heurísticas de Nielsen, onde desde o desenvolvimento do protótipo, até a codificação das telas, as regras foram aplicadas como base.

Primeiramente o desenvolvimento do protótipo teve a intenção de montar a estrutura das telas sem dar tamanha ênfase a estética e outros fatores, visando principalmente exemplificar a usabilidade e os padrões do aplicativo. Este processo foi desenvolvido através da plataforma *Figma*, pois ela permite desde o desenvolvimento visual até a navegação das telas do protótipo. Em conjunto a esta etapa, também ocorreu a definição da paleta de cores, fonte e padronização da mesma, visando o contraste e identidade do projeto.

Por fim foi iniciada a codificação do projeto através da linguagem de programação Python, uma das principais linguagens do mercado e uma das mais versáteis, por conta da existência de diversas bibliotecas desenvolvidas e sua praticidade. Em conjunto ao Python foi utilizado framework *KivyMD*, esta tecnologia possui criação recente, mas já se mostrou muito útil para os desenvolvedores, já que consegue desenvolver interfaces gráficas e fazer a execução delas em plataformas Android nativamente.

### **3.2.2.1 PROTÓTIPO**

O objetivo desta etapa foi mapear a disposição dos itens na tela, com o fim de atender os objetivos do aplicativo mencionados anteriormente, sempre tendo como base as heurísticas de Nielsen, para que os conceitos de padronização, minimalismo e cores ao serem combinados, fornecessem uma ótima experiência ao usuário independente de sua idade ou habilidade com tecnologia.

Partindo dos princípios deste projeto, é possível ver nas figuras 8, 9 e 10 que o objetivo é trazer uma tela simples, objetiva e intuitiva. Visando principalmente os usuários com mais idade ou menos afinidade com tecnologia, para que não acabem se perdendo no uso do software e desistindo de organizar sua vida financeira. E mesmo em uma tela com mais opções como na figura 9, existe um agrupamento em blocos dos elementos na tela como: saldo, ações, gastos mensais, entre outros.

A visibilidade do status da tela também é um conceito primordial quando o assunto é passar segurança ao usuário e mostrar controle sobre a aplicação. Como mostrado nas figuras 9 e 10, na parte superior da tela é sempre visível o local em que o utilizador se encontra dentro do sistema, possibilitando sempre a navegação para a tela anterior como é mostra a imagem 10 através do ícone da seta apontada para a esquerda.

A etapa do protótipo também foi extremamente importante para a definição do estilo dos formulários e como eles estariam dispostos, com a cor dos campos diferentes da cor do

fundo e o nome do campo sempre acima, para contribuir com o contraste, além da definição do tamanho e cores dos botões, cumprindo a heurística de padronização e consistência.

Figura 8 - Protótipo: Tela de login



Figura 9 - Protótipo: Tela principal





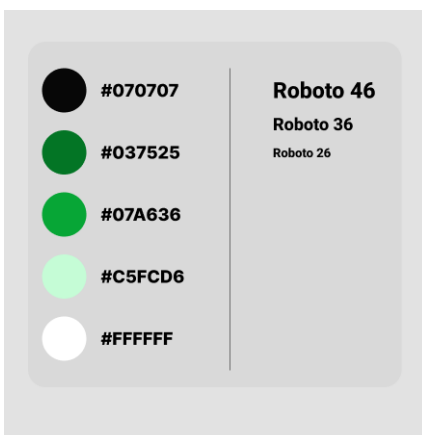
Figura 10- Protótipo: Adicionar despesas



### 3.2.2.2 PALETA DE CORES E FONTE

Um ponto determinante para uma agradável experiência no uso de aplicativos é o contraste entre as cores, desde o início do desenvolvimento a paleta de cores foi escolhida com muito cuidado, para que até os usuários com problemas de visão ou daltonismo pudessem identificar os itens na tela sem qualquer tipo de problema. Foi escolhida a cor verde como a principal do aplicativo Myfinaz, pois é a cor que remete fortemente a lembrança do dinheiro, além de ser uma cor que transmite esperança. Com a sua variação entre tons mais claros e escuros, a paleta definida está na figura 11, onde está presente também a fonte que será utilizada ao lado dos tamanhos escolhidos. A escolha da fonte em tamanhos maiores, foi justamente pensando nas pessoas com celulares menores ou algum tipo de deficiência na visão.

Figura 11- Paleta de cores utilizada



### 3.2.2.3 TELAS DO APLICATIVO

Com a etapa de prototipação concluída, foi iniciado o desenvolvimento do *front-end* através do framework *KivyMD*. O “*Kivy*” como é popularmente chamado, é um framework que possibilita a criação de interfaces gráficas que “por baixo dos panos” é executada pela linguagem de programação Python. A sintaxe do framework é bem parecida com CSS, onde através de chaves e valores é possível dar identidade para as telas.

Com o objetivo de manter o design fiel ao protótipo, principalmente com a padronização dos formulários, botões e cores, o *front-end* foi desenvolvido. Os ícones foram todos padronizados e são da mesma biblioteca de ícones do *KivyMD* como mostrados nas figuras 12 até 20, fortalecendo a identidade visual e trazendo consistência. O design foi mantido de uma forma minimalista e organizada de forma que o usuário não se perca com muitas opções na tela.

Os formulários também foram desenvolvidos cuidadosamente para prevenção de erros, como mostram as figuras 13 e 14, onde os números de caracteres são mostrados para que não haja erro na hora do preenchimento e caso tenha algum tipo de erro, o aplicativo direciona o utilizador para que seja a ação seja executada corretamente como na figura 19.

*Figura 12- Interface: Tela de login*

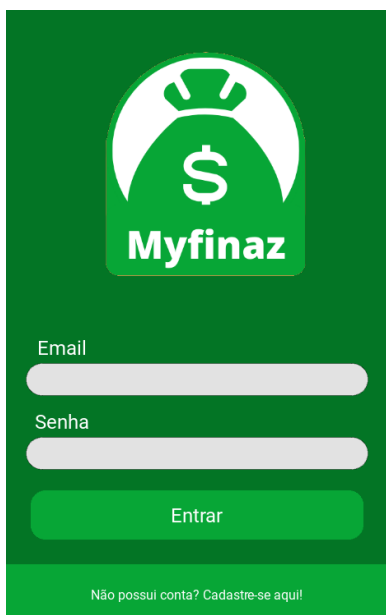


Figura 13- Interface: Tela de cadastro

The screenshot shows a mobile application interface for personal information registration. At the top, there is a dark green header with a white left-pointing arrow and the text 'Informações pessoais'. Below the header, the form consists of five input fields, each with a label above it: 'Nome completo:', 'Email', 'Data de Nascimento:', 'Senha:', and 'Confirmar Senha:'. Each input field is a light gray rounded rectangle. At the bottom of the form is a prominent green button with the white text 'Prosseguir'.

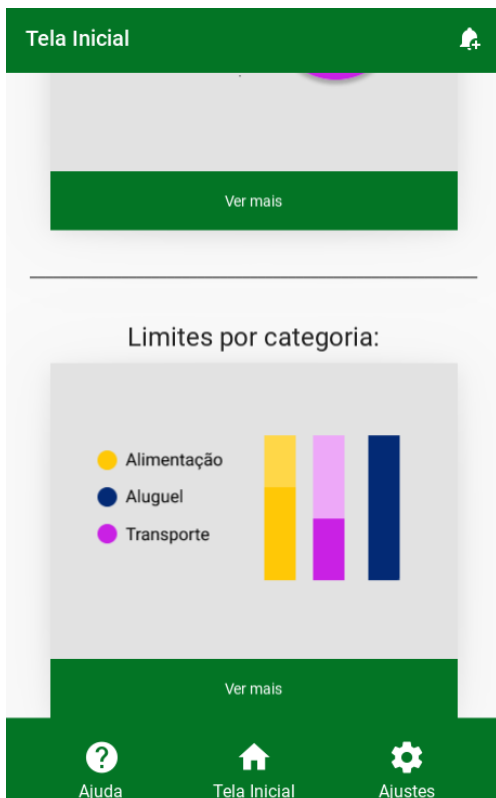
Figura 14- Interface: Tela de cadastro

The screenshot shows a mobile application interface for professional information registration. At the top, there is a dark green header with a white left-pointing arrow and the text 'Informações profissionais'. Below the header, the form consists of two input fields, each with a label above it: 'Cargo profissional:' and 'Salário mensal:'. Each input field is a light gray rounded rectangle. Below these fields is a checkbox with the text 'Concordo com os termos do aplicativo Myfinaz'. At the bottom of the form is a prominent green button with the white text 'Cadastrar'.

Figura 15 - Interface: Tela principal



Figura 16 - Interface:Tela inicial



*Figura 17- Interface: Adicionar despesa*

← Adicionar Despesas

Insira o valor:  
R\$ 00,00

Descrição

Categoria

Data

Pago

Adicionar

Ajuda Tela Inicial Ajustes

*Figura 18- Interface: Adicionar receita*

← Adicionar Receitas

Insira o valor:  
R\$ 00,00

Descrição

Categoria

Data

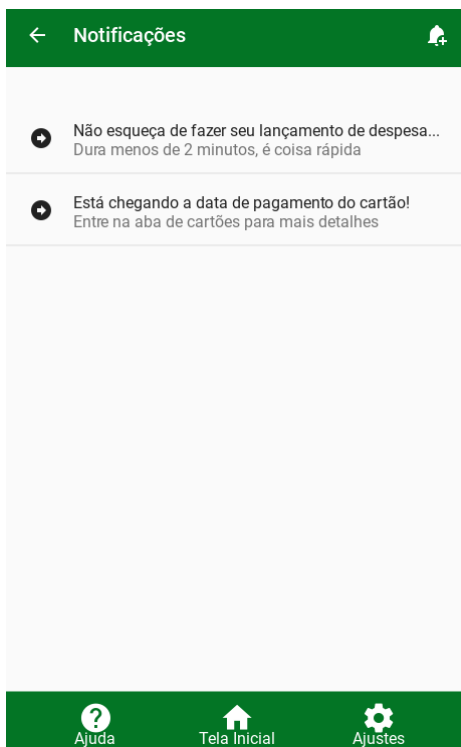
Adicionar

Ajuda Tela Inicial Ajustes

Figura 19- Interface: Tratamento de erro



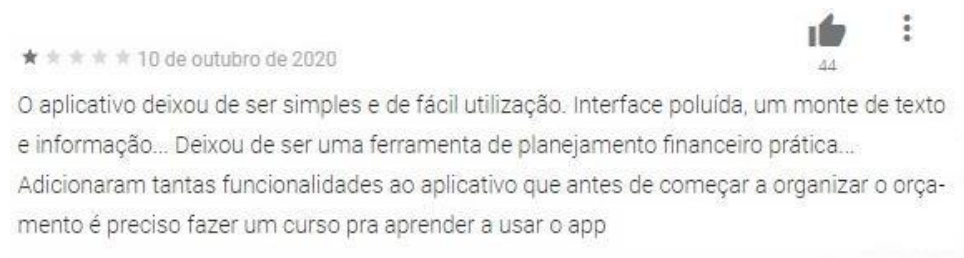
Figura 20- Interface: Notificações



## 4 RESULTADOS E DISCUSSÃO

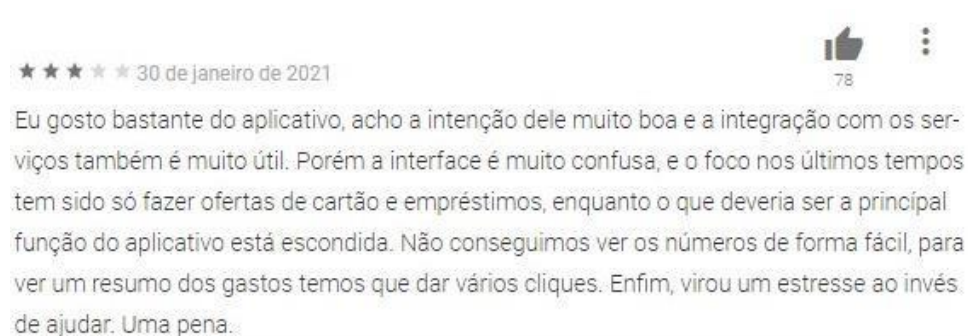
Como visto anteriormente, os índices de pessoas endividadas no Brasil são grandes e em constante crescimento, para auxiliar no entendimento desta problemática, no desenvolvimento deste projeto foi necessário analisar o cenário da educação financeira e aspectos sociológicos econômicos de maneira breve para que, fosse possível a criação do aplicativo Myfinanz, com o intuito de ajudar no planejamento financeiro dos usuários. Apesar do grande número de sistemas existentes com tal finalidade, o mercado carece de soluções com interfaces simples, intuitivas, acessíveis e gratuitas, como mostram as figuras 21, 22, 23 e 24 listadas abaixo, onde foram coletadas avaliações de alguns dos mais famosos sistemas *mobile* no ramo de planejamento financeiro:

*Figura 21 – Reclamação da interface e usabilidade no aplicativo.*



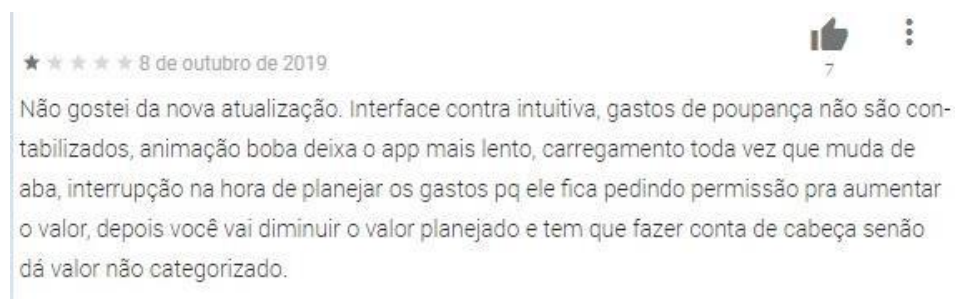
Fonte: Google play

*Figura 22 - Reclamação de interface*



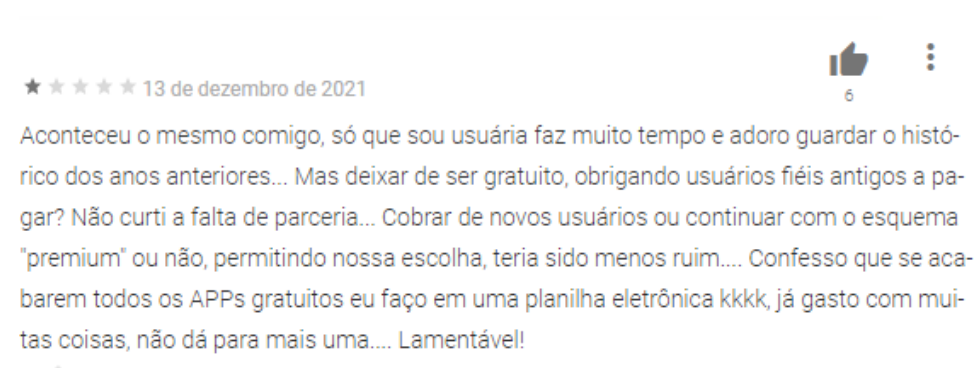
Fonte: Google play

**Figura 23- Reclamação de interface**



Fonte: Google play

**Figura 24 – Reclamação do aplicativo ao sair do gratuito para versão paga**



Fonte: Google play

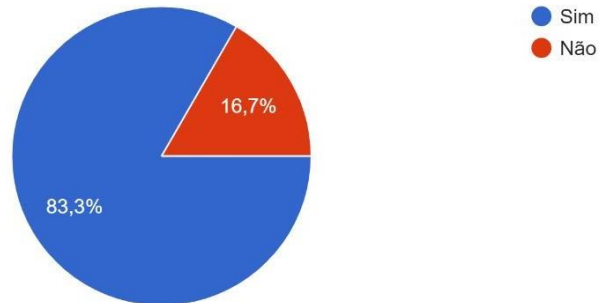
Levando em conta tais avaliações negativas dos concorrentes, o aplicativo Myfinanz se mostrou extremamente útil para as necessidades dos usuários. Todo o *front-end* passou pelos mais diferentes testes como: Testes funcionais, regressivos, além da seleção de alguns participantes de diferentes idades para o uso da plataforma, e na grande maioria das ocasiões o *front-end* foi muito bem aceito e elogiado. Após a etapa de teste da interface, por usuários convidados, foi criado um formulário com algumas perguntas simples, para receber as impressões dos utilizadores, os resultados estão presentes nas figuras 25, 26, 27 e 28.



**Figura 25- Formulário de satisfação: Pergunta 01**

Você possui algum tipo de dívida?

6 respostas

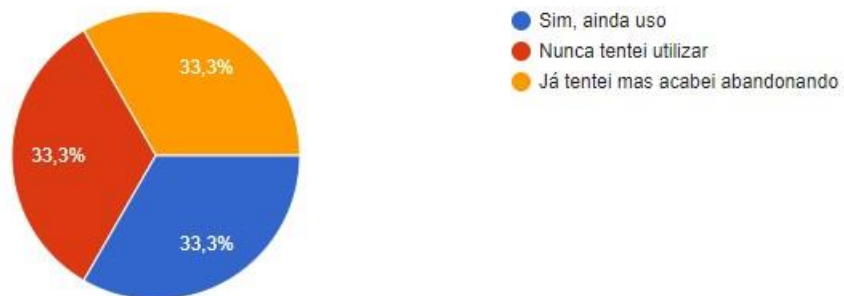


**Figura 26- Formulário de satisfação: Pergunta 02**

Já tentou utilizar algum aplicativo para fazer o planejamento financeiro mensal?

6 respostas

 Copiar



**Figura 27- Formulário de satisfação: Pergunta 03**

Após o teste do aplicativo Myfinanz, você utilizaria o sistema novamente?

6 respostas

 Copiar

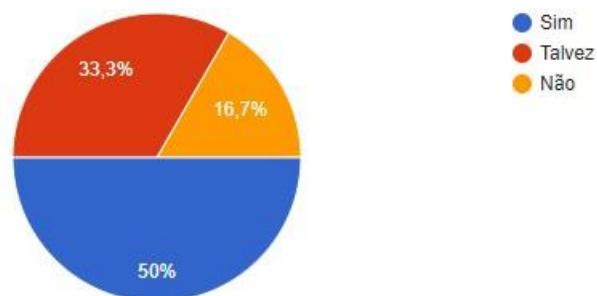
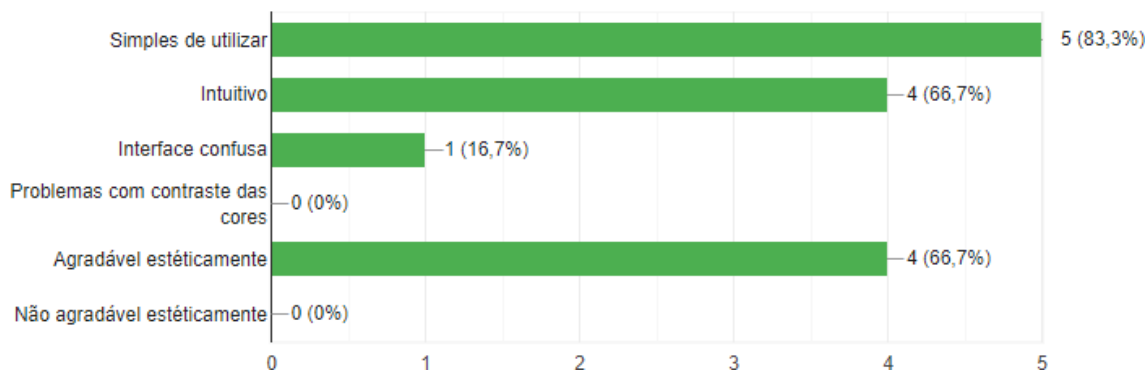


Figura 28- Formulário de satisfação: Pergunta 04

Após o teste do aplicativo Myfinanz, quais são as características mais marcantes da interface? (Escolha quantas achar necessário) [Copiar](#)

6 respostas



Concluindo, o aplicativo demonstrou muito valor e se tornou um perfeito aliado para o combate do endividamento no Brasil, já que os objetivos de manter uma interface organizada, simples, interativa, acessível e gratuita foram alcançados com o desenvolvimento do projeto.

## 5 CONSIDERAÇÕES FINAIS

O objetivo deste trabalho, foi mostrar a viabilidade da construção do *front-end* de um aplicativo *mobile*, para o controle de finanças pessoais chamado Myfinanz. Para alcançar tal objetivo, foi necessário evidenciar os números das pessoas endividadas no Brasil, além de mostrar de maneira breve os pontos positivos do planejamento financeiro e como a tecnologia pode facilitar esta tarefa. Por outro lado, também foi realizada uma pesquisa de mercado procurando os principais pontos de melhoria em aplicativos no segmento de finanças pessoais.

Dados estes objetivos, um dos pilares para a saúde financeira da população é o planejamento de capital pessoal, onde tem o papel de auxiliar o planejador a entender onde estão seus acertos e erros no pagamento das suas contas no decorrer do mês. O planejamento quando feito com seriedade e compromisso, possibilita a quitação de dívidas, liberdade financeira, entre diversos outros objetivos.

Ainda por cima, a tecnologia foi apresentada como fator facilitador destas tarefas de mapeamento de finanças, já que celulares e dispositivos móveis vieram para automatizar tarefas manuais. A popularização e acessibilidade de dispositivos móveis cresceram de maneira exponencial no mundo todo, no Brasil não foi diferente, inclusive dados mostram que existe a proporção de 2 aparelhos para um habitante, segundos dados da Fundação Getúlio Vargas (FGV).

Visto que a acessibilidade de dispositivos móveis não é um problema que inviabilize o desenvolvimento do sistema, foram procuradas avaliações de aplicativos famosos de controle de finanças, com o objetivo de buscar pontos de melhorias nos mesmos. Com isso foram identificadas diversas reclamações sobre a interação com as interfaces, a complexidade dos layouts, além da escassez de recursos gratuitos e completos.

Dessa forma, foi levantada a solução do desenvolvimento de um novo sistema, que possui em seu *front-end*, facilidade, clareza, acessibilidade. Tudo isso de uma forma gratuita e baseadas nas heurísticas de Nielsen, com o objetivo principal de ajudar a população, principalmente aqueles que possuem baixa afinidade com tecnologia e com os problemas de planejamento financeiro. O projeto foi desenvolvido de forma conjunta com o aluno Vinicius Solon Silva, no qual ficou responsável pelo desenvolvimento do *back-end* e de questões ligadas à Lei Geral de Proteção de dados (LGP).

Tabela 3- Cronograma de etapas

De 01 a 15 de agosto	Desenvolvimento dos diagramas e requisitos da interface
De 15 a 30 de agosto	Desenvolvimento do protótipo
De 01 a 30 de setembro	Desenvolvimento do <i>front-end</i>
De 01 a 15 de outubro	Integração do <i>front-end</i> com as funcionalidades do <i>back-end</i> .
De 15 a 30 de outubro	Execução dos testes
De 01 a 30 de novembro	Ajustes finais, possíveis <i>troubleshooting</i> com banco de dados e disponibilização do aplicativo para usuários.

## REFERÊNCIAS BIBLIOGRÁFICAS

BORGES, Luiz Eduardo. **Python para desenvolvedores**: Aborda Python 3.3. Rio de Janeiro: Novatec, 2014

CALIXTO, L. A.; SOUZA, D. S.; SANTOS, F. K. G. PLANEJAMENTO DE FINANÇAS PESSOAIS: UM ENSAIO PARA UMA VIDA ECONÔMICA SEGURA. **Caderno de Graduação - Ciências Humanas e Sociais - UNIT - SERGIPE**, [S. l.], v. 7, n. 2, p. 83, 2022. Disponível em: <https://periodicos.set.edu.br/cadernohumanas/article/view/8256>. Acesso em: 11 maio. 2022.

Globo.com. Disponível em: <https://g1.globo.com/economia/noticia/2022/03/31/endividamento-bate-recorde-em-marco-puxado-pelo-cartao-de-credito-diz-cnc.ghtml>. Acessado em 14 abril de 2022.

LEAL, Cicero; NASCIMENTO, José. Planejamento Financeiro Pessoal. **Revista de Ciências Gerenciais**. Brasília, v. 15, n. 22, jul. 2015.

MAIA, JS; MARIN, HF. Aplicativos móveis para as sociedades menos favorecidas. **Acta Paul Enferm.** 2021;34:eAPE002214

MOTA, Carla; CHIMELO, Géssica; BENDER, Cíntia; NESPOLO, Daniele; BORELLI, Verena; FACHINELLI, Ana. Organização Financeira Pessoal: Análise dos fatores que influenciam no endividamento e inadimplência dos jovens. **Revista Eletrônica de Ciências Sociais Aplicadas - RECSA** . Garibaldi, RS, Brasil. v.4, n.1, jan/jun, 2015.

NIELSEN, Jakob. 10 Usability Heuristics for User Interface Design. 1995.

Disponível em: <http://www.nngroup.com/articles/ten-usability-heuristics/>. Acessado em: 20 de Abril de 2022.

NIELSEN, Jakob; BUDIU, Raluca. Usabilidade Móvel. Rio de Janeiro: Elsevier, 2014.

NIELSEN, Jakob. How to Conduct a Heuristic Evaluation. 1995. Disponível em:

<<http://www.nngroup.com/articles/how-to-conduct-a-heuristic-evaluation/>>. Acessado em: 20 de abril de 2022.

NIELSEN, Jakob; LORANGER, Hoa. Usabilidade na Web: Projetando Websites com qualidade. Rio de Janeiro: Elsevier, 2006.

NIELSEN, Jakob. Usability 101: Introduction to Usability. 2012. Disponível em <<http://www.nngroup.com/articles/usability-101-introduction-to-usability/>>. Acessado em: 20 de abril de 2022.

NIELSEN, Jakob. Usability Engineering. San Diego (CA): Academic Press, 1993.

PICCINI, R.A.; PINZETTA, G. Planejamento Financeiro Pessoal e Familiar. **Unoesc & Ciência - ACSA**, Joaçaba, v. 5, n. 1, p. 95-102, jan./jun. 2014.

Radio Agência Nacional. Disponível em: <<https://agenciabrasil.ebc.com.br/radioagencia-nacional/economia/audio/2022-03/quase-78-da-populacao-esta-endividada-no-brasil>>. Acessado em 13 de abril de 2022.

RIBEIRO, Rodrigo; LARA, Ricardo. O endividamento da classe trabalhadora no Brasil e o capitalismo manipulatório, **Serv. Soc. Soc.**, São Paulo, n. 126, p. 340-359, maio/ago. 2016.

ROCHA, H. V. da; BARANAUSKAS, M. C. C. **Design e Avaliação de Interfaces Humano-Computador**. São Paulo: NIED/Unicamp, 2003. E-book. Disponível em: <https://www.nied.unicamp.br/biblioteca/design-e-avaliacao-de-interfaces-humano-computador/>. Acesso em: 17 maio. 2022.

SOARES, Michael. Metodologias Ágeis Extreme Programming e Scrum para o Desenvolvimento de Software. **Revista Eletrônica de Sistemas de Informação**. Jan-Abr 2018, v.17, n.1

SOMMERVILLE, Ian. **Engenharia de software**. 9 ed. São Paulo: Person Education, 2011.

BOOCH, G.; RUMBAUGH, J.; JACOBSON, I. **UML Guia do usuário**. 2 ed. São Paulo: Elsevier Editora Ltda, 2005.