



UNIVERSIDADE FEDERAL DO PIAUÍ – UFPI
CAMPUS SENADOR HELVÍDIO NUNES DE BARROS - CSHNB
CURSO: Bacharelado em Sistemas de Informação
DISCIPLINA: Engenharia de Software I
PROFESSORA: Pâmela Carvalho

DOCUMENTO DE REQUISITOS E DIAGRAMAS DO SOFTWARE DIGITALBANK

**PICOS – PI
MARÇO DE 2023**

INTEGRANTES DO PROJETO

NOME		FUNÇÃO
<u>João dos Santos Neto</u>	-	<u>designer</u>
<u>Armando Luz Borges</u>	-	<u>desenvolvedor</u>
<u>Jamile Jovita da Silva</u>	-	<u>entrevistada</u>

1. INTRODUÇÃO

1.1 Propósito do documento

Este documento tem o intuito de detalhar o sistema Digital Bank. O propósito é permitir que a instituição realize suas operações bancárias de forma eficiente e segura, utilizando sistemas informatizados para processar transações financeiras, gerenciar contas e realizar outras atividades relacionadas a serviços bancários. Além de garantir a eficiência, segurança e a confiabilidade das operações bancárias, permitindo que a instituição ofereça serviços financeiros de alta qualidade aos seus clientes. O sistema também ajuda a reduzir erros humanos, melhorar a velocidade e a precisão do processamento de transações e fornecer informações valiosas sobre o desempenho das atividades bancárias.

Por ser um banco digital oferece serviços bancários de forma mais acessível, eficiente e conveniente, utilizando tecnologia para simplificar processos e reduzir custos.

1.2 Escopo do sistema

Objetivos:

- Desenvolver um sistema bancário digital.
- Permitir que os usuários possam ter acesso e controle sobre seu saldo, possibilitando a movimentação do mesmo.
- Construir uma interface minimalista e simples.

Funcionalidades:

- Cadastro rápido de novas contas.
- Controle de saldo, com atualização automática do saldo após alguma movimentação.
- Geração de extratos bancários, exibindo todas as movimentações feitas pelo usuário.
- Integração multiplataforma.

Limites:

- O sistema será desenvolvido apenas para adultos, maiores de 18 anos.
- O sistema não irá integrar a plataforma mobile inicialmente.
- O sistema não terá opções de investimento do saldo.

Critérios de aceitação:

- O sistema deve possuir estabilidade para suportar diversos clientes simultâneos.
- O sistema deve possuir eficiência razoável na execução de tarefas.
- O sistema deve ser testado e livre de erros que comprometam a experiência do usuário.

1.3 Descrição do restante dos capítulos do documento

O restante do documento está dividido em mais quatro capítulos, nos quais cada um descreve partes do sistema, bem como informações acerca do mesmo. Sendo assim, O capítulo 2 fornece uma descrição geral do sistema, relatando sobre seu domínio, funções, características e restrições. O capítulo 3 apresenta os requisitos específicos, como os atributos do sistema, ferramentas e frameworks utilizados, requisitos funcionais e não funcionais, além das técnicas de validação de requisitos que serão usadas para avaliar o sistema. O capítulo 4 apresenta os diagramas do sistema, bem como uma breve apresentação do protótipo. Por fim, o capítulo 5 mostra a evolução do sistema apresentando os requisitos validados e os trabalhos futuros propostos a esse sistema.

2. DESCRIÇÃO GERAL

2.1 Domínio do Produto

O domínio de um software bancário envolve diversos aspectos relacionados às operações bancárias, incluindo segurança, transações financeiras, produtos bancários, serviços de atendimento ao cliente, entre outros. Alguns exemplos de elementos do domínio de um software bancário que são por exemplo:

Regulamentação: precisa estar de acordo com diversas regulamentações, como leis de proteção ao consumidor, de segurança da informação e etc..

Produtos e serviços: o software deve suportar diversos produtos e serviços bancários, como contas correntes, cartões de crédito, empréstimos, investimentos, entre outros.

Segurança: É necessário ter um alto nível de segurança para proteger as informações dos clientes e as transações financeiras realizadas.

Transações financeiras: o software bancário deve suportar diversas transações financeiras, como transferências, pagamentos, saques, depósitos, entre outras. É importante que essas transações sejam rápidas, seguras e confiáveis.

Sendo assim o domínio de um software bancário envolve diversos aspectos mais complicados e específicos do setor bancário, que devem ser cuidadosamente considerados e implementados para garantir um produto de alta qualidade e de confiança.

2.2 Funções do Produto

- **Login:** essa função realiza a autenticação do cliente no sistema, recebendo o login informado para verificar se existe no banco de dados o respectivo cliente, caso o login exista o usuário é redirecionado para a respectiva interface, caso não é informado que os dados do login não existem.
- **Cadastrar:** essa função realiza cadastro de clientes, exigindo alguns dados para criação da conta no sistema.

ENGENHARIA DE SOFTWARE – DOCUMENTO DE REQUISITOS E DIAGRAMAS

- **Transferir:** essa função permite que o cliente realize transferência para outras contas pertencentes ao DigitalBank ou em outros bancos.
- **Depositar:** essa função permite a realização de depósitos na respectiva conta do cliente.
- **Sacar:** essa função realiza saques, com restrição de que o saque não pode ser maior que o saldo atual da conta, ou seja, não pode realizar saques maiores que a quantia em depósito.
- **Extrato:** essa função exibe na interface de usuário todas as operações realizadas pelo cliente(saques,depósitos e transferências).

2.3 Características dos Usuários

- **Faixa etária:** O software deve ser projetado para atender as necessidades da ampla faixa etária, a partir de usuários mais jovens, acima da idade mínima requerida, até usuários mais velhos.
- **Nível de escolaridade:** O sistema deve ser projetado com acessibilidade suficiente para atender aos mais diversos níveis de escolaridade. Neste sentido, o software deve ser intuitivo, simples e fácil de usar.
- **Faixa de renda:** O software deve ser construído de forma otimizada, a fim de ser utilizável em diversos tipos de hardware, atendendo às necessidades da maioria das parcelas da população.

2.4 Restrições Gerais

- **Prazo:** O software deve ser construído dentro de um prazo de seis meses. Neste sentido, as funcionalidades essenciais do software, que tange a movimentação do saldo do cliente, devem ser priorizadas.
- **Recursos:** A equipe de desenvolvimento deve utilizar ferramentas e tecnologias open-source ou versões community para desenvolver o sistema. Além disso, os recursos de hardware podem ser considerados inicialmente como o próprio objeto de trabalho.
- **Tecnologias:** As tecnologias devem ser open-source ou versões community. A plataforma de desenvolvimento deve ser padrão para todos os desenvolvedores. É sugerido a utilização do Visual Studio Code. A criação de interfaces gráficas deve ser construída com a versão gratuita do framework PyQt5. Os dados devem ser armazenados em um banco de dados. É sugerido a versão community do MySQL. A conexão e comunicação do cliente com o servidor deve ser estabelecida através de *sockets*. É sugerida a linguagem Python para construir a aplicação.

ENGENHARIA DE SOFTWARE – DOCUMENTO DE REQUISITOS E DIAGRAMAS

- **Padrão:** O padrão ISO/IEC 12207 deve ser considerado no processo de desenvolvimento, verificação e validação do sistema.

3. REQUISITOS ESPECÍFICOS

3.1 Atributos dos Sistemas com especificação de prioridade.

- **Atributos prioritários:**
 - **Segurança:** O sistema deve ser capaz de proteger os dados do usuário, mantendo a sua perfeita integridade, mesmo em situações de ataque internos ou externos. Palavras-chave devem ser criptografadas antes de serem armazenadas no banco de dados.
 - **Portabilidade:** O sistema deve ser portátil para diversas plataformas desktop, possibilitando assim a sua execução nos mais diversos sistemas operacionais.
 - **Manutenibilidade:** O sistema deve ser construído de forma extensiva e manutenível para aumentar o seu ciclo de vida.
 - **Confiabilidade:** O software deve ser estável, sendo capaz de operar sem falhas continuamente, e ser capaz de se recuperar rapidamente a eventuais erros no sistema.
- **Atributos necessários:**
 - **Usabilidade:** O sistema deve ser simples de usar e entender pelos usuários, possibilitando que o usuário realize tarefas de forma intuitiva e rápida.
 - **Desempenho:** O sistema deve ser otimizado e eficiente para que este não seja muito custoso em hardwares mais simples, não prejudicando assim a experiência do usuário.

3.2 IDE, Banco de Dados, Linguagem de Programação utilizada, etc.

- IDE utilizada para desenvolvimento do sistema foi Visual Studio Code 1.76
- O sistema a linguagem SQL utilizada para armazenar os dados: MySQL
- Linguagem de Programação utilizada: Python

3.3 Requisitos Funcionais e Não funcionais do sistema.

- **Requisitos funcionais:**
 - O sistema deve permitir que os usuários criem contas rapidamente.
 - O sistema deve permitir que os usuários façam login em suas contas bancárias.
 - O sistema deve possibilitar a realização de transações com o saldo.
 - O sistema deve permitir que o usuário possa ver o histórico de transações.
 - O sistema deve ter uma tela de perfil dos usuários, exibindo informações sobre sua conta.
- **Requisitos não funcionais:**
 - O sistema deve proteger os dados dos clientes contra ataques internos e externos.

- O sistema deve ser otimizado para ter bom desempenho em hardwares mais fracos.
- O sistema deve ser simples para incluir usuários iniciantes.
- O sistema deve ser intuitivo para incluir usuários de todos os níveis de escolaridade.
- O sistema deve ser confiável e suportar falhas e erros sem quebrar sua execução.
- O sistema deve ser manutenível e extensível para aumentar seu ciclo de vida.
- O sistema deve ser portátil para diversas plataformas desktop.
- O sistema deve possuir interface gráfica minimalista e limpa.

3.4 Técnicas de validação de requisitos que serão usadas

Nossas técnicas de validação baseiam-se no feedback retornado dos nossos Usuários Primários e Secundários. Onde nesse feedback é feito um questionário, que consiste em avaliar de 1 a 5, a respeito de diversos aspectos que o aplicativo apresenta: segurança, responsabilidade, visibilidade, confiança e eficiência.

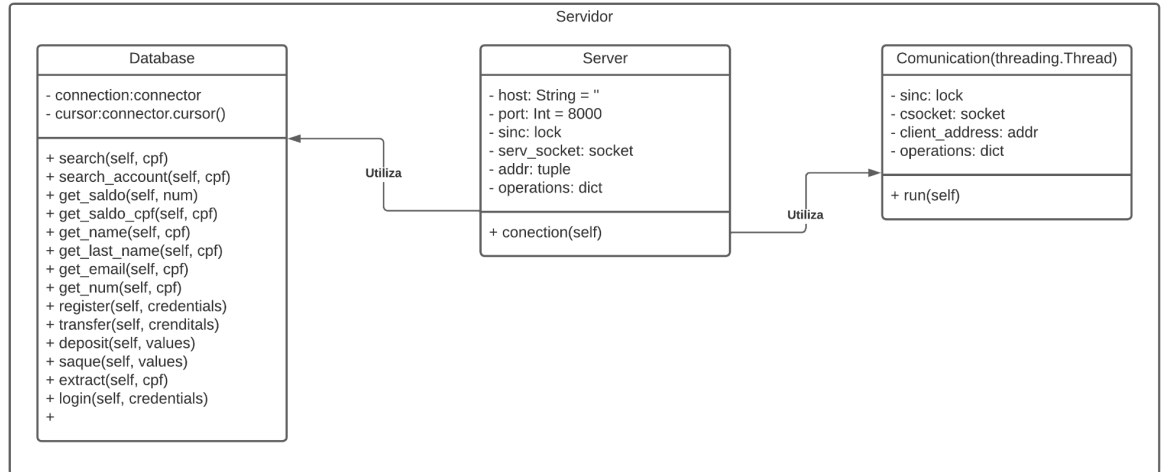
Feito toda essa captura de feedback, os dados são enviados para a efetuação dos cálculos que consistem em gerar uma média ponderada das respostas enviadas pelos usuários que responderam.

Estima-se que uma média de aceitação de 7 nos aspectos responsabilidade e visibilidade, 8 em segurança e confiança e 9 em eficiência.

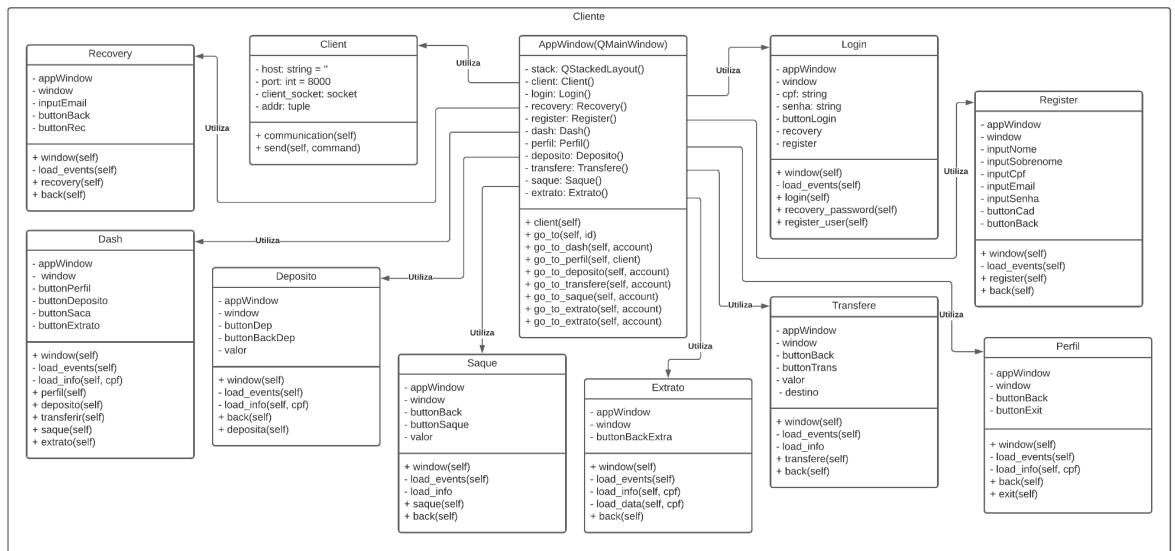
4. MODELOS DO SISTEMA

5.1 Diagrama de Classes

- Diagrama do servidor



- Diagrama do cliente



5.2 Interfaces do Sistema

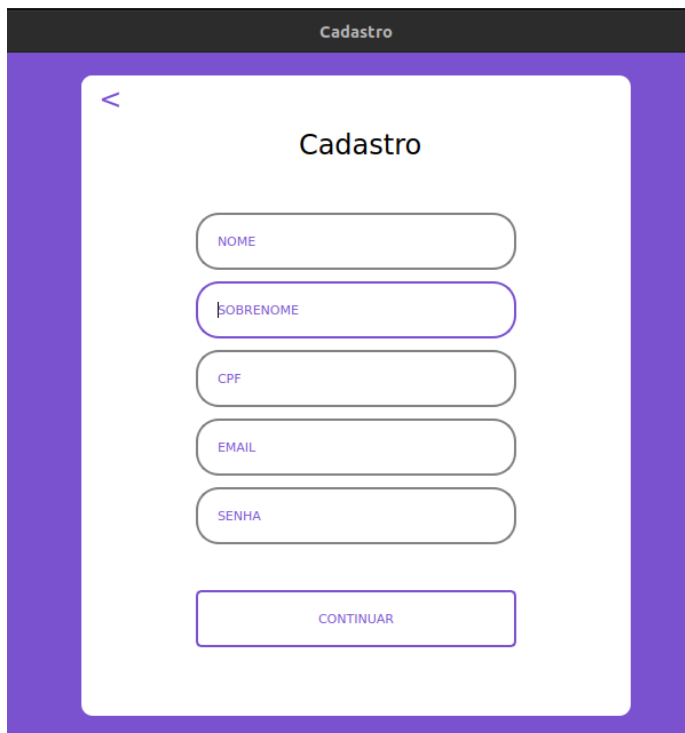
- Tela de login

ENGENHARIA DE SOFTWARE – DOCUMENTO DE REQUISITOS E DIAGRAMAS



A mockup of a login screen. It features a dark purple header bar with the title "Login" and standard window control icons. The main content area has a light purple background. In the center, there is a white rounded rectangle containing the text "Faça seu login". Below this, there are three input fields: the first is labeled "CPF", the second "SENHA", and the third is a button labeled "CONTINUAR". At the bottom of the white rectangle, there are two links: "Esqueci minha senha >" and "Ainda não sou cliente >".

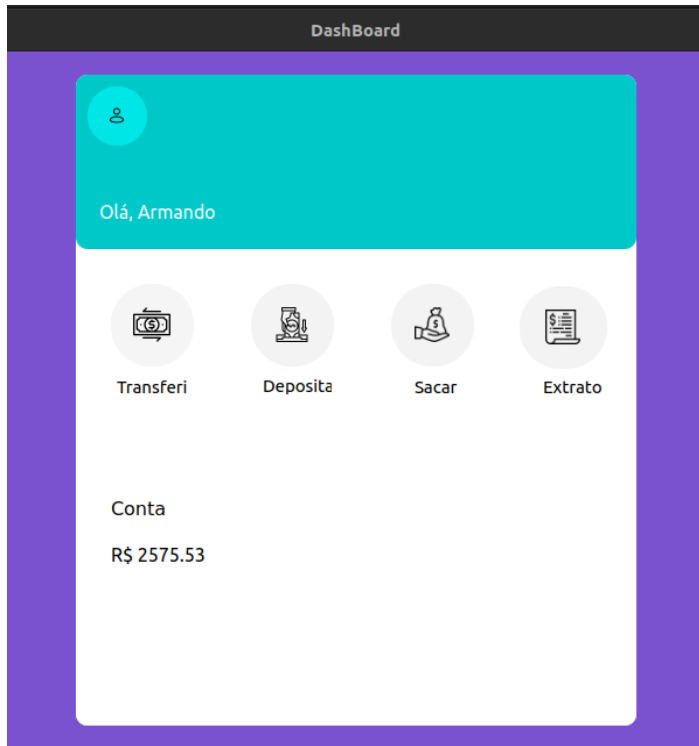
- **Tela de cadastro**



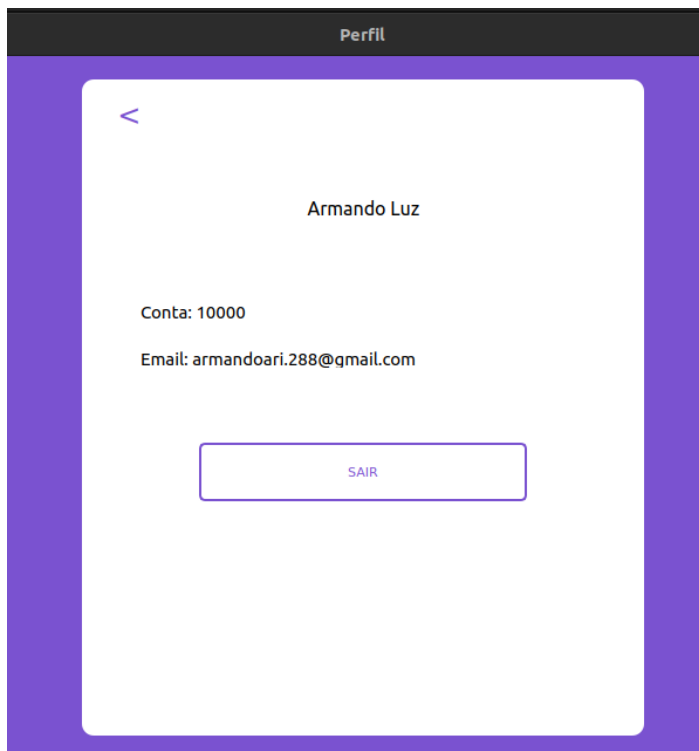
A mockup of a registration screen. It features a dark purple header bar with the title "Cadastro". The main content area has a light purple background. In the center, there is a white rounded rectangle containing the text "Cadastro". Below this, there are five input fields: "NOME", "SOBRENOME", "CPF", "EMAIL", and "SENHA". At the bottom of the white rectangle, there is a button labeled "CONTINUAR". A back arrow icon is located in the top left corner of the white rectangle.

- **Dashboard**

ENGENHARIA DE SOFTWARE – DOCUMENTO DE REQUISITOS E DIAGRAMAS



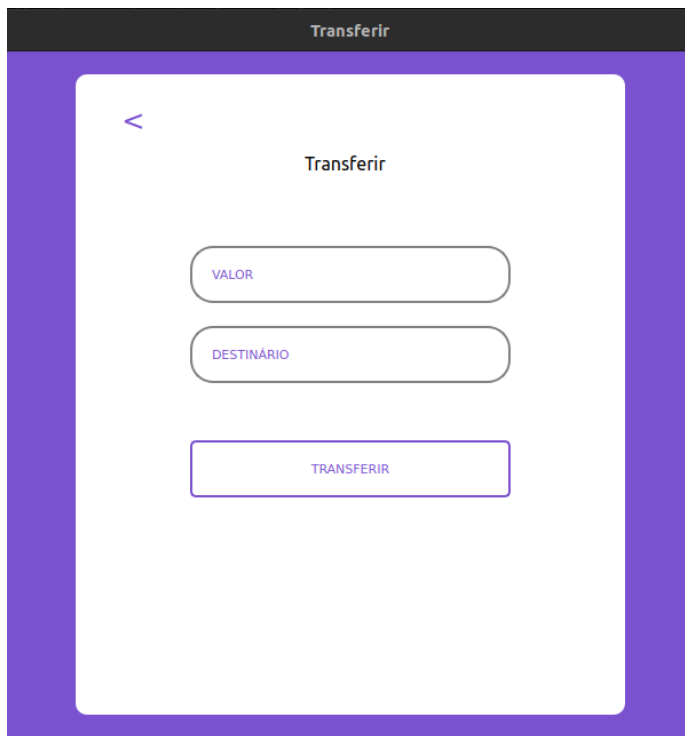
- **Tela de perfil**



- **Tela de transferência**

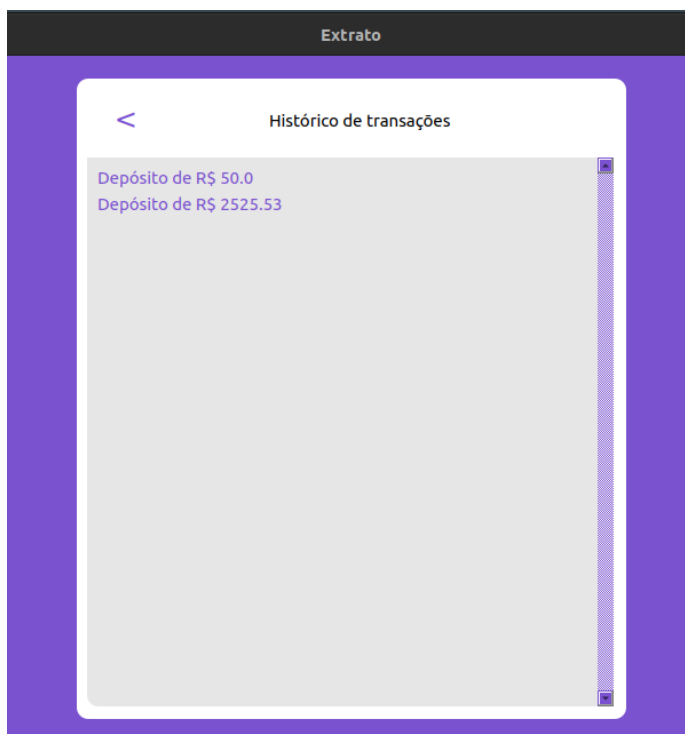
ENGENHARIA DE SOFTWARE – DOCUMENTO DE REQUISITOS E DIAGRAMAS

As demais funcionalidades que tem por objetivo movimentar o saldo seguem a mesma estrutura desta tela.



The image shows a mobile app screen titled "Transferir" in a dark header. The main content area has a white background with a purple border. It features a back arrow icon in the top left, the title "Transferir" in the center, and three input fields: "VALOR", "DESTINÁRIO", and a "TRANSFERIR" button at the bottom.

- **Tela de extrato**



The image shows a mobile app screen titled "Extrato" in a dark header. The main content area has a white background with a purple border. It features a back arrow icon in the top left, the title "Histórico de transações" in the center, and a list of transactions: "Depósito de R\$ 50.0" and "Depósito de R\$ 2525.53".

5. EVOLUÇÃO DO SISTEMA

6.1 Requisitos validados

Dos requisitos totais do sistema foram validados responsabilidade e visibilidade com uma aprovação de 8,6 de média. Dos requisitos de segurança e confiança o sistema apresentou uma aprovação de 7,4 de média e do aspecto eficiência obteve uma aprovação de 9,5 de média, contendo diversos comentários positivos a respeito do custo-benefício.

6.2 Trabalho Futuros

O DigitalBank atualmente consiste em um sistema simples e minimalista, nossa equipe de desenvolvedores planejam futuramente a implementação de novas funções, tais como realização de transferência via PIX, realização de geração de QR Code, login via Face ID de usuário (esta função pode, ou não, ser habilitada pelo usuário).

Na parte de interface de usuário, nossos designers pretendem deixar o sistema mais fluido, com movimentações mais leves e aconchegantes, novas mudanças em cores e bordas, visibilidade de informações rápidas para facilitar o uso do sistema. Também será implementada a possibilidade de usar temas claros e escuros.

Outra disponibilidade que será implementada é a geração de cartões de crédito e débito, para uso de saques e efetuar compras em lojas, respectivamente. Planos futuros, ainda nesse mesmo tópico, são a criação de cartões virtuais.