



Faculdade de Tecnologia Dep. Júlio Julinho Marcondes de Moura

CURSO DE TECNOLOGIA EM ANÁLISE E DESENVOLVIMENTO DE SISTEMAS

IARA LEODORO TEOTONIO DE SOUZA MAYCON MOREIRA DE SÁ

FATEC: MEXPENSES

Garça
2022

Faculdade de Tecnologia Dep. Júlio Julinho Marcondes de Moura



CURSO DE TECNOLOGIA EM ANÁLISE E DESENVOLVIMENTO DE SISTEMAS
IARA LEODORO TEOTONIO DE SOUZA MAYCON MOREIRA DE SÁ

FATEC: MEXPENSES

Relatório técnico e/ou científico apresentado à Faculdade de Tecnologia de Garça – FATEC, como requisito para conclusão do Curso de Tecnologia em Análise e Desenvolvimento de Sistemas.

Orientador: Prof. Me. Luiz Carlos Querino Filho

Faculdade de Tecnologia Dep. Júlio Julinho Marcondes de Moura

CURSO DE TECNOLOGIA EM ANÁLISE E DESENVOLVIMENTO DE SISTEMAS

IARA LEODORO TEOTONIO DE SOUZA MAYCON MOREIRA DE SÁ

FATEC: MEXPENSES

Relatório técnico e/ou científico apresentado à Faculdade de Tecnologia de Garça – FATEC, como requisito para conclusão do Curso de Tecnologia em Análise e Desenvolvimento de Sistemas.

Data da Aprovação: ---/---/---

Prof. Me. Luiz Carlos Querino Filho
FATEC Garça

Prof. (a) (membro da banca)
FATEC Garça

Prof. (a) (membro da banca)
FATEC Garça

Dados Internacionais de Catalogação na Publicação (CIP)

Sá, Maycon Moreira de; Souza, Iara Leodoro Teotonio de

FATEC Título Mexpenses. - Garça, 2022.

32 f.

Relatório Técnico (Trabalho de Conclusão de Curso de Tecnologia em Análise e Desenvolvimento de Sistemas) – Faculdade de Tecnologia “Dep. Julio Julinho Marcondes de Moura” – FATEC, 2022. Orientador: Luiz Carlos Querino Filho

RESUMO

O presente projeto visa o desenvolvimento de um software de integração bancária como uma solução financeira, para inúmeras pessoas que acabaram tendo suas vidas afetadas por conta da Pandemia de COVID-19 que teve seu início no ano de 2020. Com isso houve grande aumento no número de pessoas desempregadas, e a expansividade no preço de insumos levando ao agravamento econômico e social brasileiro. Mediante ao exposto será desenvolvido um software para que os indivíduos possam ter acesso a suas economias de forma simples e prática facilitando o dia a dia e a organização financeira em um tempo de crises evitando que haja a necessidade de pessoas saírem de casa para ter acesso ao seu gerenciamento bancário.

Palavras-chave: Software. Desenvolvimento. Gerenciamento Bancário.

ABSTRACT

The present project aims to develop a banking application software as a financial solution for countless people who ended up having their lives affected because of the COVID-19 Pandemic that started in the year 2020. With this, there was a great increase in the number of unemployed people, and the increase in the price of inputs leading to the Brazilian economic and social aggravation. Based on the above, a software program will be developed so that individuals can have access to their savings in a simple and practical way, facilitating their daily lives and financial organization in a time of crisis, avoiding the need for people to leave their homes to have access to their bank management.

Keywords: Software. Development. Bank Management.

SUMÁRIO

| | |
|--|----|
| FATEC Garça..... | 3 |
| SUMÁRIO | 5 |
| 1 INTRODUÇÃO | 6 |
| 1.1 OBJETIVOS..... | 7 |
| 1.2 JUSTIFICATIVA..... | 8 |
| 2. DESENVOLVIMENTO..... | 8 |
| 2.1 ANÁLISE DA ESTRUTURA LÓGICA | 8 |
| 2.2 DESENVOLVIMENTO <i>FRONT-END</i> | 20 |
| 2.3 DESENVOLVIMENTO <i>BACK-END</i> | 22 |
| 3. DEMONSTRAÇÃO DA APLICAÇÃO | 24 |
| 4. REFERÊNCIAS..... | 32 |

1 INTRODUÇÃO

O presente projeto teve seu início a partir do cenário caótico de pandemia no ano de 2020, o principal intuito do aplicativo é trazer aos seus usuários facilidade em seu dia a dia ao necessitarem de um banco com praticidade e segurança, para que dessa forma todos tenham acesso a suas informações bancárias sem a necessidade de saírem do conforto e segurança da sua casa.

A partir desse objetivo, foi se proposto o planejamento de criar uma aplicação que pudesse suprir a essa necessidade e poder facilitar todos os processos bancários de relacionamento entre clientes e bancos além de suprir o principal problema em meio a pandemia de COVID-19 manter usuários longe de aglomerações e contaminações.

Foi realizado um levantamento de requisitos para aprimoramento saber se essa aplicação realmente poderia trazer um impacto positivo a vida de seus usuários e poder suprir os maiores défices no desenvolvimento do projeto.

A pesquisa foi desenvolvida a partir de pesquisas em relações a aplicações bancárias e suas plataformas de acessos, em questões de pontos positivos, negativos e quais eram as maiores dores de seus atuais clientes, dessa forma foi criar uma forte base para o desenvolvimento do atual projeto auxiliando novas funcionalidades. Com base nos requisitos levantados foi-se notório a necessidade de uma aplicação bancário, intuitiva, completa e segura para seus usuários.

1.1 OBJETIVOS

1.1.1 Objetivo Geral

O presente projeto tem como objetivo estudar a vida da sociedade moderna em meio a pandemia de COVID-19, visando detectar oportunidades de melhorias na eficiência de processos e soluções bancárias. Avaliar individualmente cada usuário buscando a melhor forma de gerenciar suas finanças.

Sistema de administração de monitoramento participativo, como forma de obter informações sobre gastos econômicos trazendo ao usuário de forma constante uma análise de metas, gastos e limites. Gerenciamento total e análises financeiras ligadas diretamente a cada usuário de maneira rápida e interativa.

1.1.2 Objetivo Específico

- Facilitar processos bancários por uma plataforma completa;
- Desenvolver uma plataforma utilizando tecnologias como *React*, *JavaScript*, e *Bootstrap* (*front-end*);
 - Uso da linguagem *Python* par a construção de um api que seja consumida pela plataforma e forneça integração ao banco de dados *MySQL* (*back-end*).

1.2 JUSTIFICATIVA

O presente projeto visa o desenvolvimento de um software de aplicação bancária como uma solução financeira, para inúmeras pessoas que acabaram tendo suas vidas afetadas por conta da Pandemia de COVID-19 (CIOTTI, 2020, p. 365-388) que teve seu início no ano de 2020. Com isso houve grande aumento no número de pessoas desempregadas, e a expansividade no preço de insumos levando ao agravamento econômico e social brasileiro. Mediante ao exposto será desenvolvido um software para que os indivíduos possam ter acesso a suas economias de forma simples e prática facilitando o dia a dia e a organização financeira em um tempo de crises evitando que haja a necessidade de pessoas saírem de casa para ter acesso ao seu gerenciamento bancário.

2. DESENVOLVIMENTO

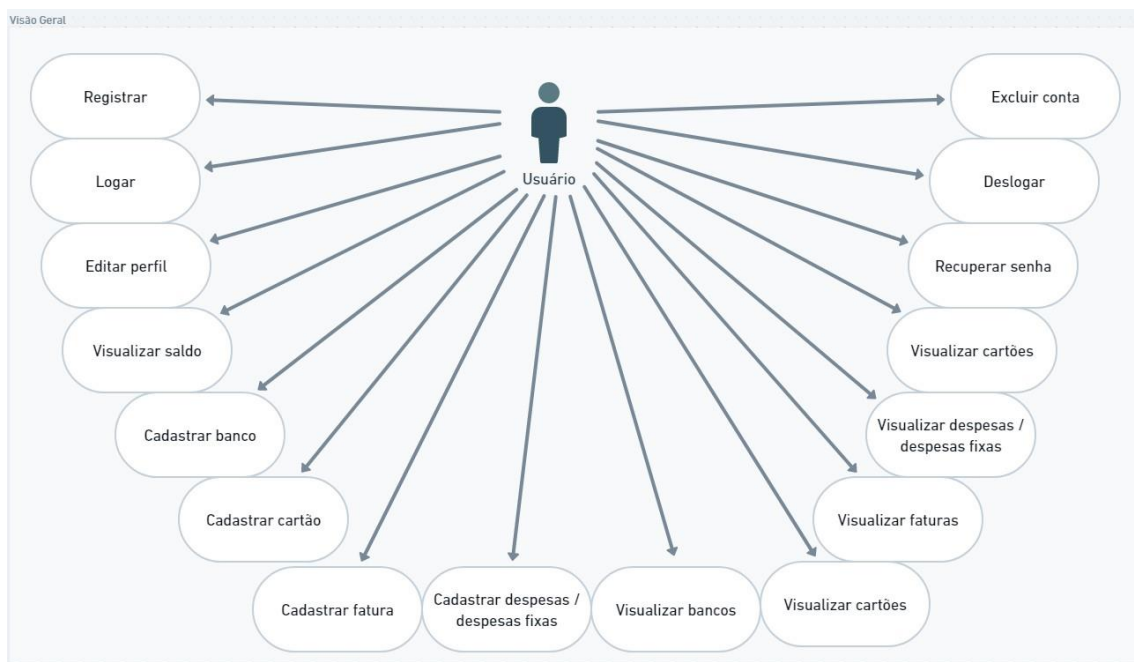
O desenvolvimento desse projeto ocorreu fundamentalmente em três fases, sendo:

- análise da estrutura lógica;
- desenvolvimento do *front-end*;
- e desenvolvimento do *back-end*, onde são abordadas as tecnologias que foram utilizadas no decorrer deste projeto.

2.1 ANÁLISE DA ESTRUTURA LÓGICA

Assim como foi citado anteriormente, este projeto tem como objetivo criar um aplicativo capaz de facilitar todo o processo de estágios supervisionados, portanto é de suma importância todas as etapas do desenvolvimento estarem bem definidas, para que então seja possível deixar o aplicativo com a maior facilidade de uso ao usuário final.

Figura 1- Visão Geral do Sistema

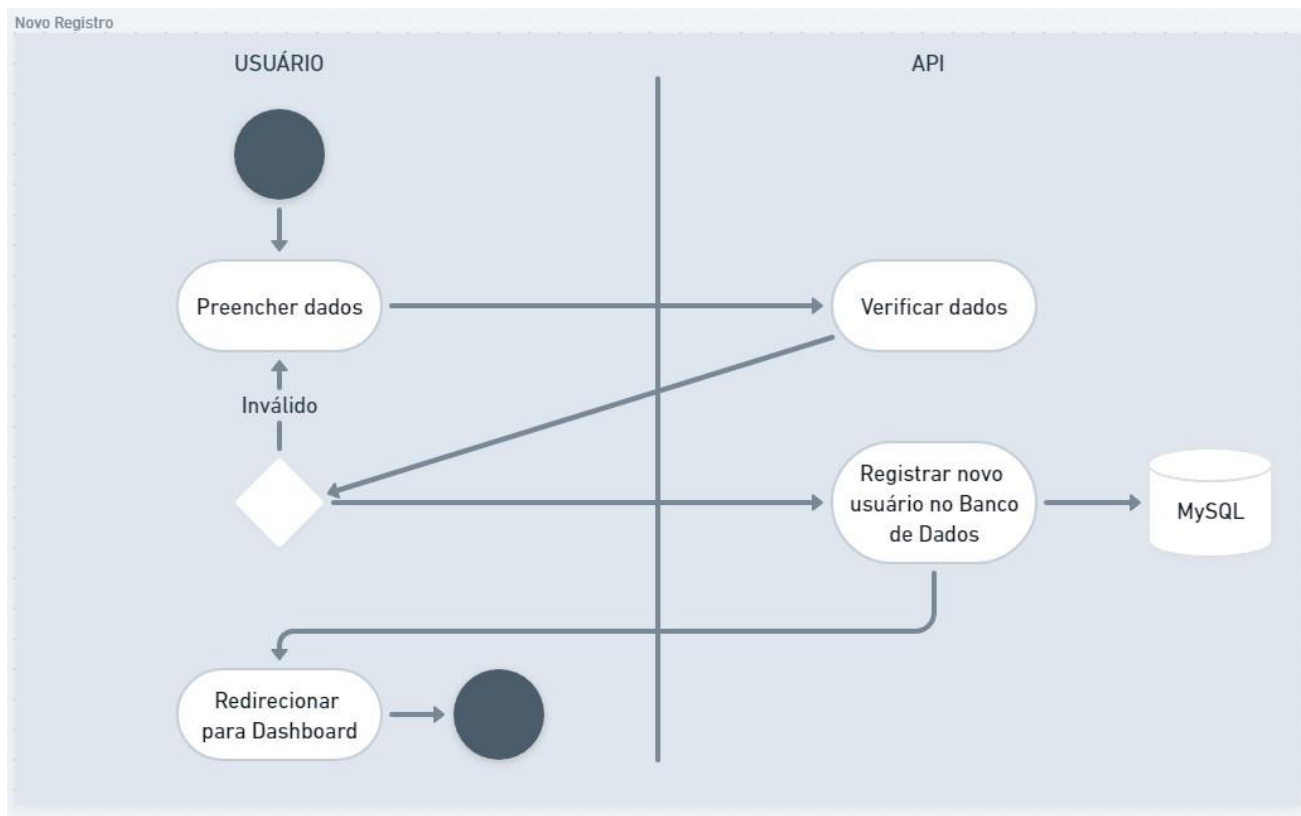


Fonte: o Autor

Diante o conteúdo demonstrado na figura 1 está descrito o diagrama de caso de uso do aplicativo. É ponderado que há um ator usuário e suas permissões.

O ator herda as funcionalidades de um usuário comum que poderá realizar *login*; caso o usuário tiver sido cadastrado é possível realizá-lo, recuperar a senha se necessário, editar perfil, visualizar saldo, cadastrar banco, cadastrar cartão, cadastrar fatura, cadastrar despesas, visualizar bancos, visualizar cartões, visualizar faturas, visualizar despesas, visualizar cartões, desconectar do aplicativo e excluir sua conta.

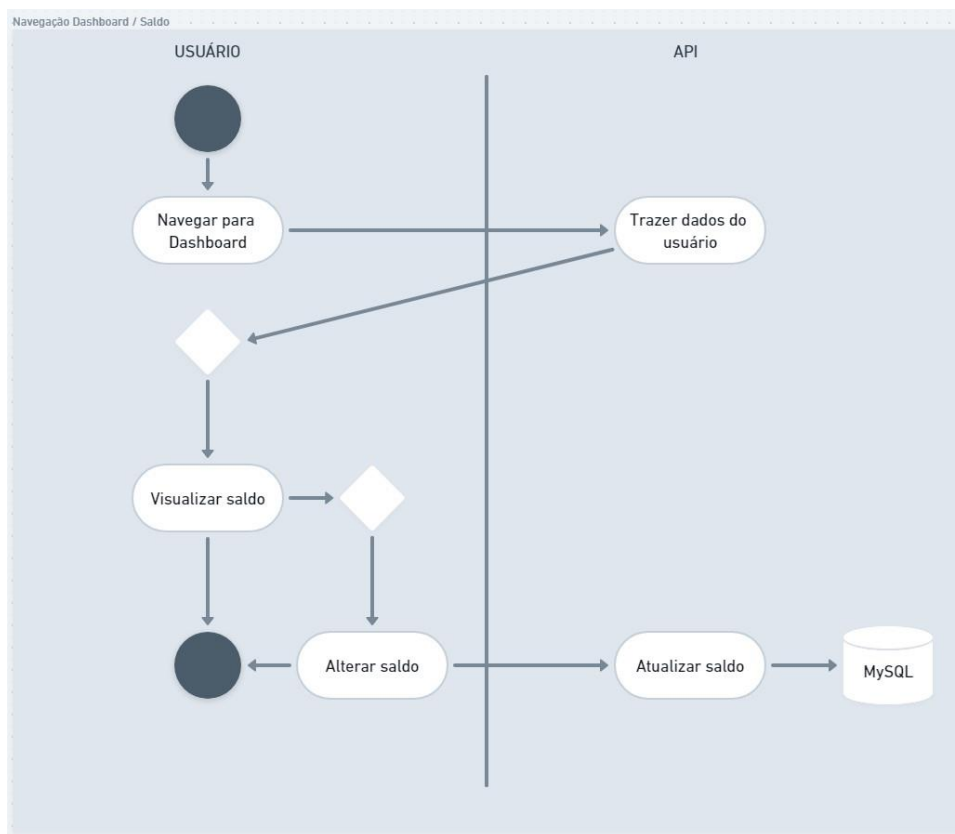
Figura 2 – Cadastro



Fonte: o Autor

De acordo com a figura 2 é explanado o funcionamento lógico da tela de cadastro, onde o usuário preenche seus dados e é verificado os dados, até que sejam validados, caso eles estiverem corretos os dados são salvos no banco de dados, logo após será efetuado o *login* do usuário.

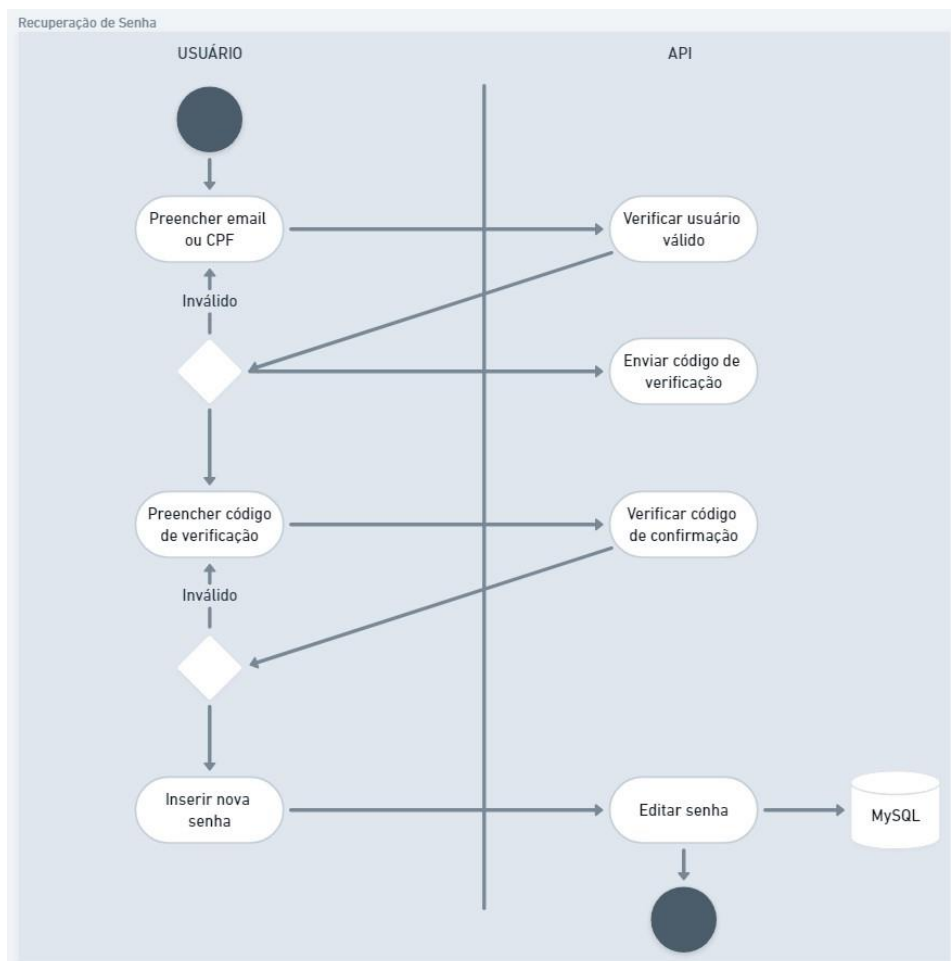
Figura 3 – Atualizar saldo



Fonte: o Autor

Conforme demonstrado na figura 3, é apresentado o esquema da tela de atualização do saldo, onde é possível visualizar os dados, e caso o saldo seja alterado o valor é atualizado e salvo no banco de dados.

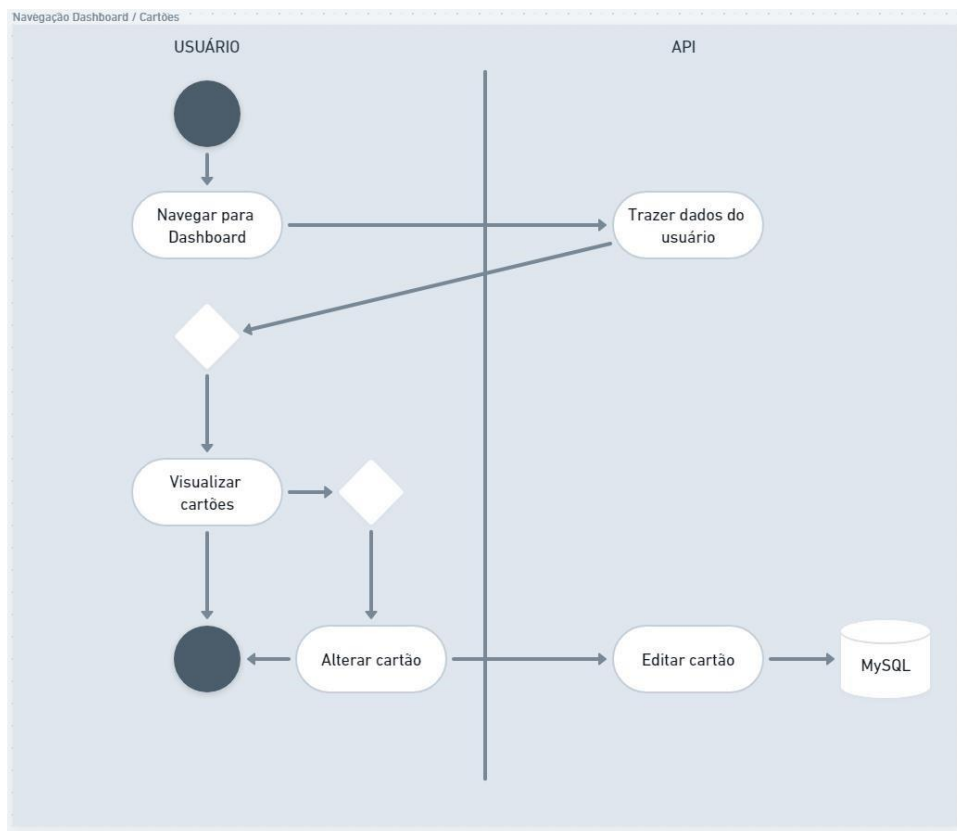
Figura 4 – Recuperação de senha



Fonte: o Autor

Conforme representado na figura 4, caso o usuário esqueça sua senha de acesso, ele poderá redefini-la. Primeiramente, ele irá informar seu e-mail ou CPF, onde a *API* irá validar se ele é um usuário já cadastrado ou ir diretamente para a segunda tela de recuperação de senha, onde irá inserir o código de verificação, caso seja válido ele irá mandar um código de verificação para este e-mail. Com esse código ele poderá então redefinir uma nova senha que será enviado para *API* e salva no banco de dados.

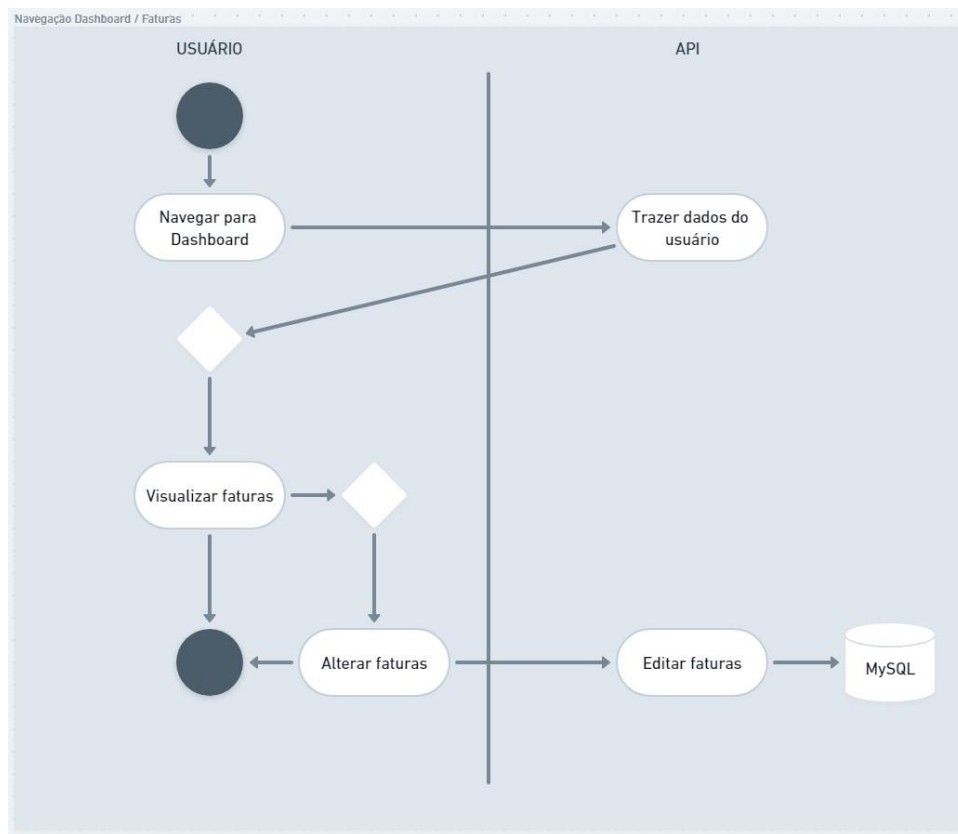
Figura 5 –Cartões



Fonte: o Autor

Conforme a figura 5 o usuário logado poderá visualizar os cartões de crédito e caso seja necessário alterar o cartão de crédito ou editá-lo a ação será salva no banco de dados.

Figura 6 – Faturas



Fonte: o Autor

A figura 6 mostra como será feita a visualização das faturas, e será possível alterar essas faturas, caso elas sejam alteradas ou editadas serão salvas no banco de dados.

Figura 7 – Visualização do calendário de despesas



Fonte: o Autor

Conforme a figura 7, a visualização do calendário onde será possível ver cada despesa prevista para o mês correspondente.

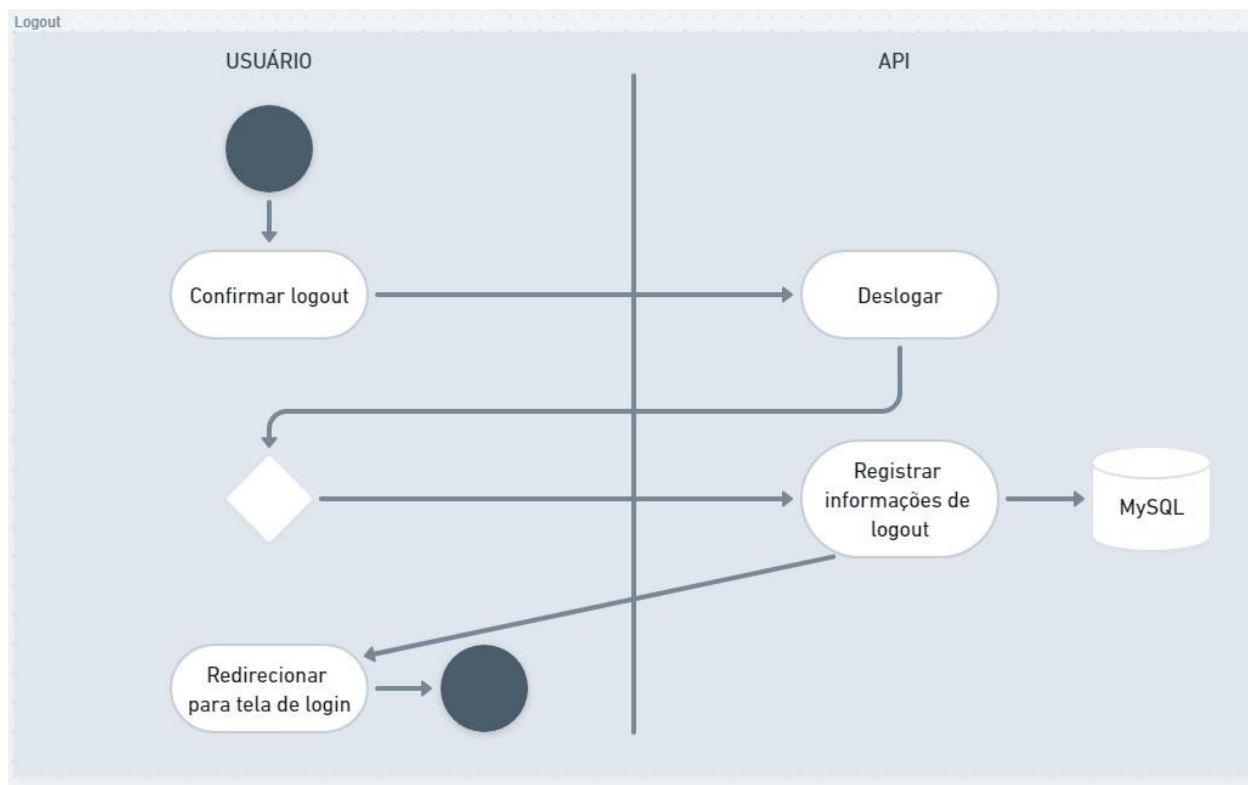
Figura 8 – Visualizar gráfico de despesas



Fonte: o Autor

A figura 8 mostra o gráfico de despesas onde graficamente será possível ver o nível de gasto do mês respectivo.

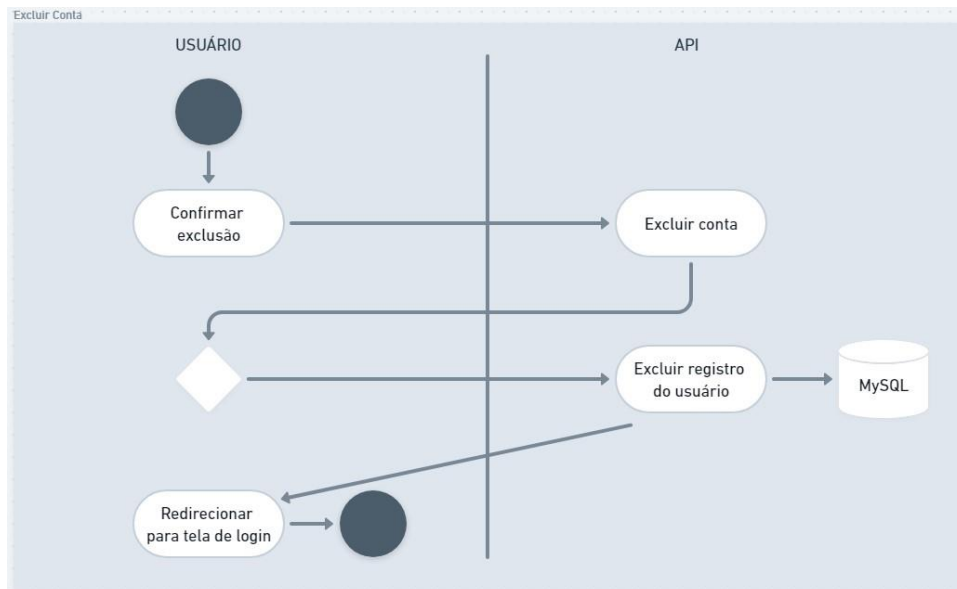
Figura 9 – Logout



Fonte: o Autor

Na figura 9 tem-se o processo de *logout* confirmar o *logout*, registrar as informações de *logout* e salvar no banco de dados, após essa etapa será redirecionada para a tela de login novamente.

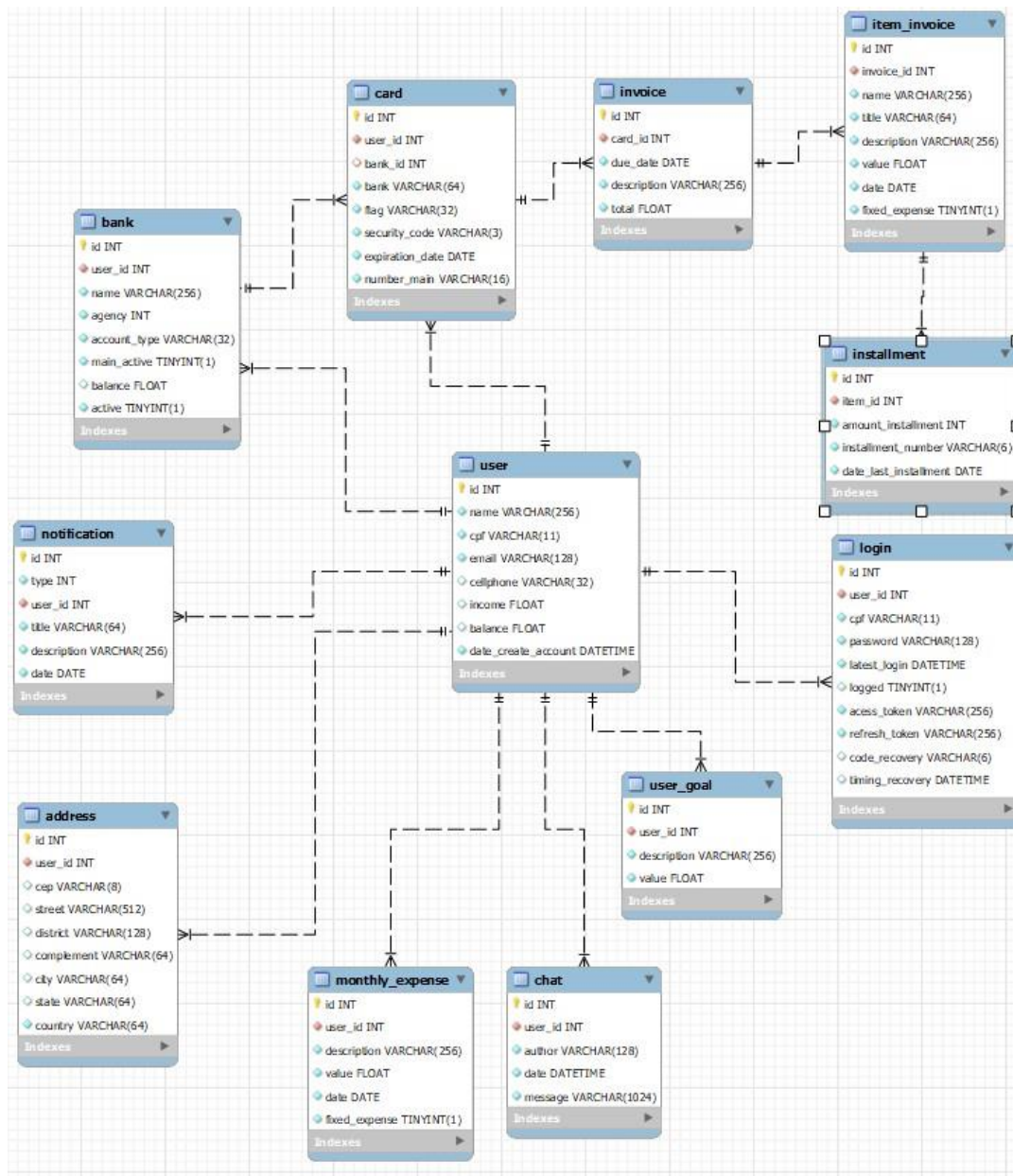
Figura 10– Exclusão da conta



Fonte: o Autor

Conforme a figura 10, a exclusão da conta será feita a partir de uma confirmação do usuário, onde será feita uma requisição para o banco de dados e pôr fim retornar a tela de login.

Figura 11– Diagrama ER



Fonte: o Autor

Conforme a figura 11, as tabelas *card*, *invoice*, *item_invoice*, *installment*, *login*, *chat*, *monthly_expense*, *address* e *notification* se relacionam com a tabela *user* e *user_goal*. A relação de todas as tabelas mante a conexão com a aplicação.

2.2 DESENVOLVIMENTO *FRONT-END*

O desenvolvimento desta parte do sistema, onde se interage diretamente com o usuário, assim como onde também fica toda parte visual da ferramenta, foi realizado em *JAVASCRIPT*. Para tal, foram utilizadas bibliotecas com o intuito de aperfeiçoar e otimizar o aplicativo mencionado nesse projeto. Elas são descritas nas próximas seções (JAVASCRIPT, 2021)

2.2.1 *React Native*

React é uma biblioteca *JavaScript* para construir interfaces de usuário, o *React* facilita a criação de *UIs* interativas. Projete visualizações simples para cada estado em seu aplicativo, e o *React* atualizará e renderizará com eficiência apenas os componentes certos quando seus dados forem alterados. (*REACT*, 2021)

2.2.2 Saas

O *Syntactically Awesome Style Sheets (Sass)*, em português Folhas de Estilo Sintaticamente Incríveis, como o nome sugere, permite a estilização das telas de uma maneira muito enxuta. Ele é uma alternativa ao clássico *Cascading Style Sheets (CSS)*, em português Folhas de estilo em cascata. Arquivos escritos em Sass chegam a ter metade da quantidade de linhas do que um arquivo escrito em CSS. Apesar de não interpretável pelos navegadores, existem várias ferramentas capazes de converter o Sass em CSS. A vantagem de sua utilização existe apenas no desenvolvimento, tornando possível uma produtividade maior ao estilizar as telas do sistema, pois depois, quando o sistema for publicado, o arquivo será processado para uma folha de estilo padrão (SASS, 2019).

2.2.3 *Bootstrap*

Bootstrap, como era conhecido originalmente, é um framework front-end criado em meados de 2010. Sua concepção se deu pela combinação dos esforços do designer Mark Otto e pelo desenvolvedor Jacob Thornton enquanto trabalhavam no Twitter. Segundo Mark, o projeto tinha como premissa melhorar as ferramentas internas da empresa (WAUTERS, 2012), começando com coisas simples de HTML e CSS, até que Jacob começou a desenvolver plugins e ambos decidiram torná-lo código aberto.

Bootstrap adapta seus projetos Web com um simples código, de smartphones para tablets e para desktops através do recurso conhecido como média queries do CSS. Além disso, apresenta uma excelente documentação para elementos HTML, componentes CSS e *HTML* personalizados e plug-ins *jQuery*. Segundo informações presentes em seu site oficial, “o *Bootstrap* usa CSS tradicional, mas seu código fonte utiliza os dois pré-processadores CSS mais populares, Less e Sass” (BOOTSTRAP, 2016)

2.2.3 *FontAwesome*

Font Awesome é um conjunto de ícones vetoriais escaláveis que podem ter personalizados, instantaneamente, o tamanho, cor, sombra, e outras características compatíveis com CSS.

2.2.4 *Gulp*

Gulp.js é uma ferramenta de automação de tarefas em *JavaScript*. Tarefas como minificar, otimizar e compilar arquivos, tão repetitivas e necessárias ao desenvolvimento, podem ser automatizadas com o *Gulp*.

2.2.4 Chart

Chart.js é uma biblioteca em *JavaScript* que auxilia na criação de gráficos utilizando apenas HTML, CSS e JS para renderizar os gráficos na tela do usuário, para isso, utiliza-se o elemento canvas do *HTML5*.

2.2.5 Argon

Argon é um sistema de design amigável, de código aberto e bonito baseado no *Bootstrap 4* usando *React*, *Reactstrap* e *create-react-app*.

2.3 DESENVOLVIMENTO *BACK-END*

Esta seção aborda o âmbito do desenvolvimento do *Back-end* deste sistema, onde se localiza as regras de negócios, segurança, assim como o banco de dados. De maneira a programar e diante tantas linguagens e tecnologias foi escolhido o *Python3* para a *API* juntamente com o *framework Flask*, para a criação do banco de dados o *MYSQL*.

2.3.1 Python

Python é uma linguagem interpretada orientada a objetos de alto nível com semântica dinâmica. A simplicidade da sintaxe dessa linguagem torna mais fácil a depuração e manutenção do código bem como a própria tecnologia estimula o reuso dos códigos e utilização de módulos nativos da linguagem, reduz custos de processamentos das aplicações e por ser multiplataforma é capaz de ser usada à diversos afins. (PYTHON, 2021)

2.3.2 *Flask*

Flask é um pequeno *framework web* escrito em *Python*. É classificado como um *microframework* porque não requer ferramentas ou bibliotecas particulares, mantendo um núcleo simples, porém, extensível. Não possui camada de abstração de banco de dados, validação de formulário ou quaisquer outros componentes onde bibliotecas de terceiros pré-existentes fornecem funções comuns. No entanto, o *Flask* oferece suporte a extensões que podem adicionar recursos do aplicativo como se fossem implementados no próprio *Flask*. Existem extensões para mapeadores objeto-relacional, validação de formulário, manipulação de upload, várias tecnologias de autenticação aberta e várias ferramentas comuns relacionadas ao *framework*. (FLASK, 2021)

2.3.3 *MYSQL*

MYSQL é um sistema gerenciador de banco de dados relacional que se utiliza da linguagem *sql* (*Structre Query Language*). Uma das linguagens mais usadas para gerenciamento de banco de dados.

Pelo fato de ser um sistema comum na comunidade dos desenvolvedores e ser relacional, foi escolhido para cruzar as informações, como relacionar dados tais como identificação de alunos e inscrições, vagas e empresas etc. (MYSQL, 2021)

3. DEMONSTRAÇÃO DA APLICAÇÃO

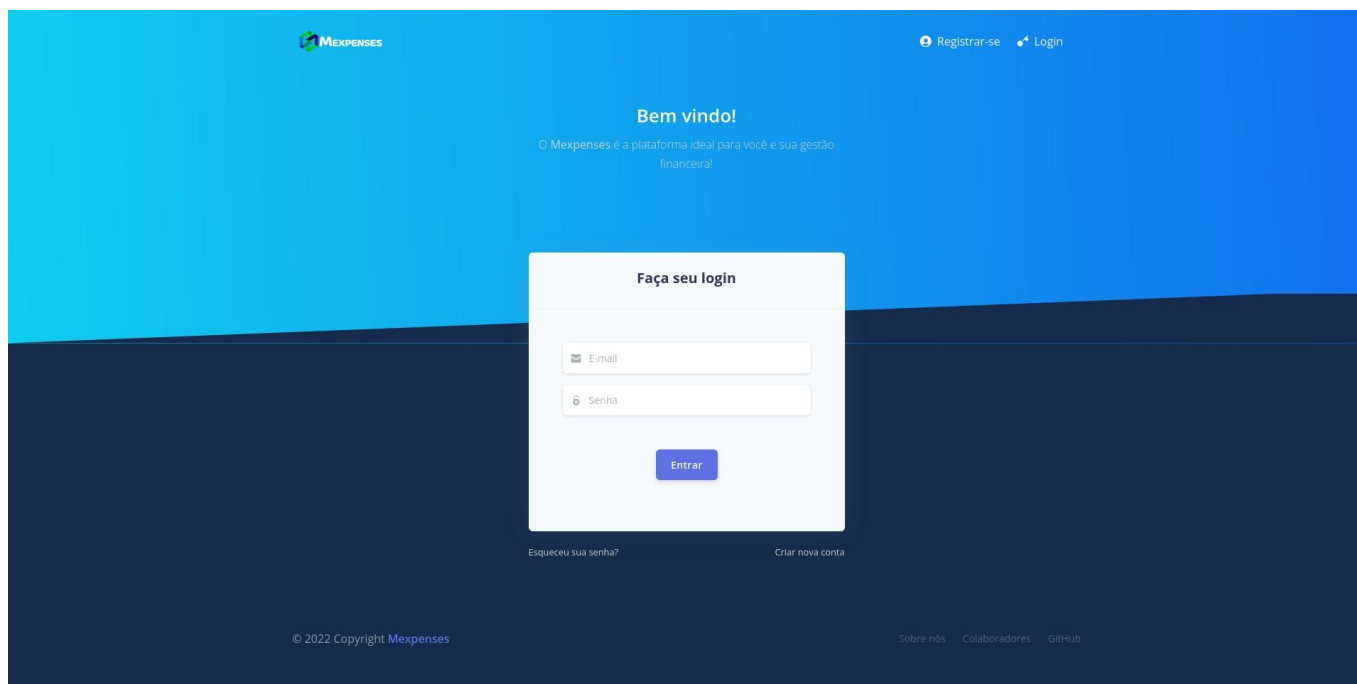
Figura 12– Tela Inicial

A imagem mostra a interface de usuário da aplicação Mexpenses. No topo, há um cabeçalho azul com o logo 'MEXPENSES' à esquerda e os links 'Registrar-se' e 'Login' à direita. O corpo da página tem um fundo azul escuro com uma onda branca no topo. No centro, há um formulário branco com o título 'Registrar-se'. O formulário contém os seguintes campos: 'Nome', 'E-mail', 'Telefone celular', 'CPF', 'CEP', 'Senha' (com uma dica de senha: 'Senha somente composta por números com 6 caracteres'), 'Saldo' e 'Renda'. Abaixo dos campos, há uma caixa de seleção para 'Eu concordo com os termos de Política e Privacidade.' e um botão azul 'Create account'. No rodapé, há o copyright '© 2022 Copyright Mexpenses' e links para 'Sobre nós', 'Colaboradores' e 'GitHub'.

Fonte: o Autor

Conforme a figura 12, a tela *inicial* possui os campos de requisição dos dados pessoais do usuário para que ele possa se cadastrar, após os dados preenchidos ao clicar no botão (Create account) os dados serão enviados a *API* que realizará o cadastro do novo cliente. A mesma tela possui a opção de login caso o usuário já seja cadastrado e deseje acessar a conta, assim sendo redirecionado para a tela de *login*.

Figura 13- Tela de login



Fonte: o Autor

Conforme a figura 13, a tela de *login* tem o campo de e-mail e senha onde o usuário poderá preencher, após preenchido os campos a *API*, a mesma tela possui também as opções de recuperar a senha e criar uma conta na aplicação.

Figura 14- Tela de perfil

MEU PERFIL

Olá Maycon

Esta é sua página de perfil. Você pode ver suas informações gerais sobre você e editá-las quando achar melhor.

[Excluir minha conta](#)

Minha conta [Salvar](#)

INFORMAÇÕES DE USUÁRIO

| | |
|----------------------|----------------------------|
| Nome | E-mail |
| Maycon Moreira de Sá | moreira.maycon98@gmail.com |
| Senha | Confirmar senha |
| 6 | 6 |

INFORMAÇÕES DE LOCALIDADE E CONTATO

| | | |
|------------------|------------------------------|-------------|
| Telefone celular | Endereço | Complemento |
| 14991876429 | Doutor Miguel Bruno Ferreira | Casa |
| Cidade | Estado | CEP |
| Marília | São Paulo | 00000-000 |

4 Bancos **4** Cartões **22** Transações

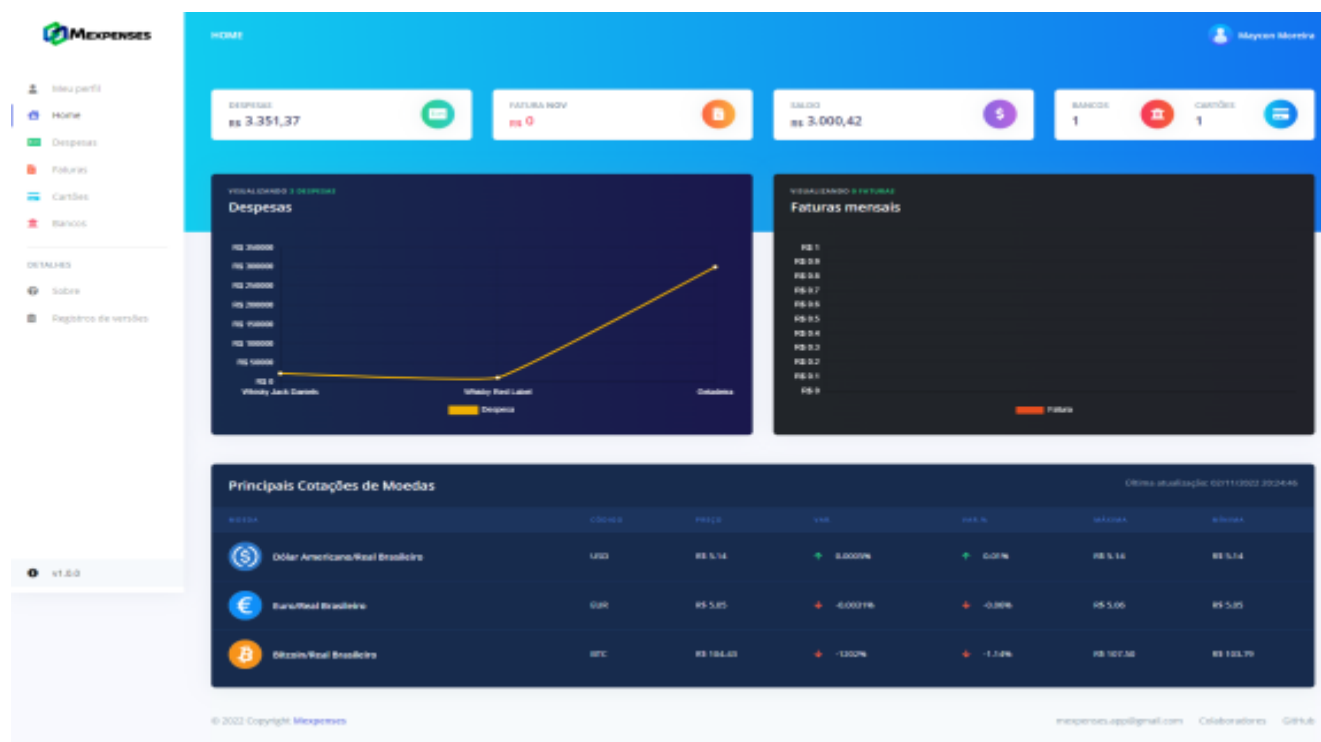
Maycon Moreira de Sá
000.000.000-00
moreira.maycon98@gmail.com

v1.0.0 © 2022 Copyright Mexpenses Sobre nós Colaboradores Github

Fonte: o Autor

Conforme a figura 14, a tela possui todas as informações pessoais do usuário, além de informar a quantidade bancos e cartões que o cliente possui. Ao lado esquerdo na mesma tela é possível ter acesso a tela inicial da aplicação e as suas demais funcionalidades.

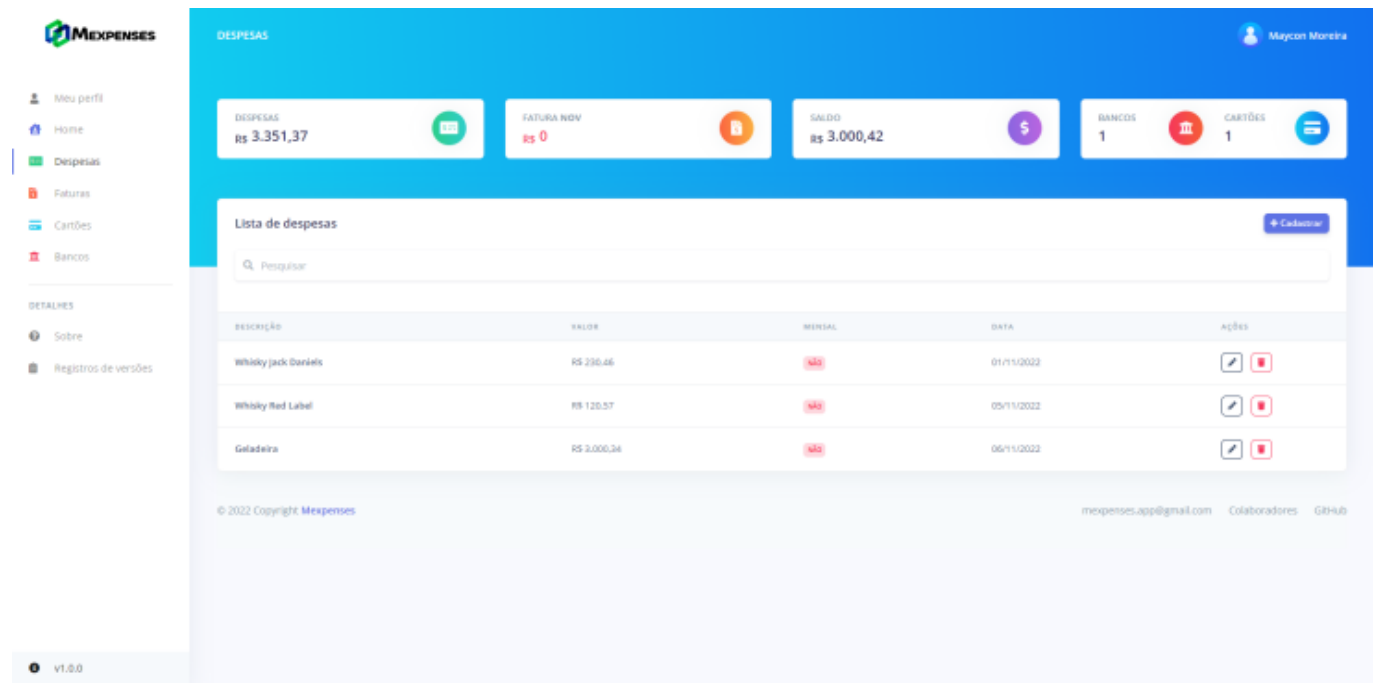
Figura 15- Home



Fonte: O autor

Conforme a figura 15, a tela possui as principais informações bancárias do usuário como o saldo, valor total de despesas, bancos e cartões cadastrados, tais informações são visualizadas em todas as telas listadas no menu inferior a esquerda. Além de gráficos que possuem o valor de suas despesas e os meses comprometidos de suas finanças, na tela também é possível visualizar quais são e o valor das principais cotações de moedas.

Figura 16- Despesas



Fonte: O autor

Conforme a figura 16, a tela lista todas as despesas cadastradas pelo usuário a mesma permite a edição de cada uma delas como também a exclusão, ao cadastrar uma nova despesa cada uma delas terá a sua descrição, valor, mensal (se aquela devida despesa ocorrerá mensalmente, ou apenas naquele mês) e a data em que foi realizada. A mesma também possui um campo de pesquisa onde é possível filtrar ou procurar despesas específicas.

Figura 17- Faturas

The screenshot shows the 'Faturas' (Invoices) screen in the Mexempresas application. The interface is divided into a sidebar and a main content area.

Sidebar:

- Meu perfil
- Faturas
- Relatório de vendas

Main Content Area:

Summary Bar:

- Saldo:** R\$ 3.351,37
- Fatura Nova:** R\$ 0
- Saldo:** R\$ 3.000,42
- Saldo:** 1
- Saldo:** 1

Form Fields:

- Cartão:**
 - Mastercard - Banco do Brasil SA
 - Youtank
 - 01/11/2020
- Quanto do mês:**
 - Local de sua compra
 - Descrição (descrição)
 - Quantidade de parcelas (parcelas)
 - Valor
- Botões:**
 - Adicionar gasto
 - Calcular

Lista de faturas:

| Nome | Quantidade | Valor | Saldo de seu depósito | Aplicar |
|--------------------------------------|------------|-------|-----------------------|---------|
| Nenhuma informação para ser exibida. | | | | |

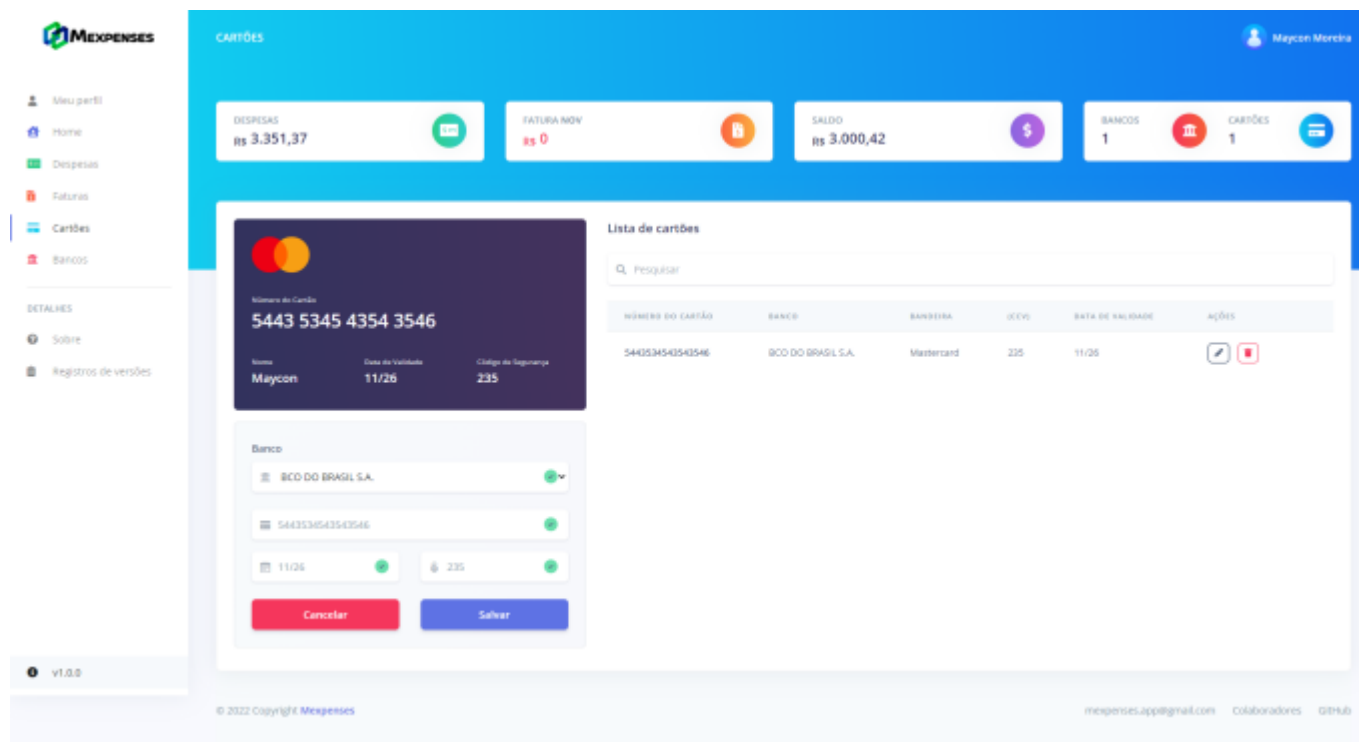
Transactions List:

- 06/12/2022**
 - Nubank**
 - Valor: R\$ 2.448,48
 - Manual**
 - Compra do mês**
 - Compra para casa: R\$ 2.313
 - Manual**
 - Recurso Lanchete**
 - R\$ 84,90
 - Manual**
 - Recurso**
 - R\$ 55,68

Fonte: O autor

Conforme a figura 17, a tela possui campos onde é possível cadastrar um cartão e a suas faturas do mesmo, onde os gastos serão registrados com local de compra, descrição, quantidade de parcelas, e o seu valor, logo ao lado será possível visualizar todas as despesas naquela determinada fatura com o seu valor total gasto. A tela possui também um campo de pesquisa onde é possível filtrar uma determinada fatura.

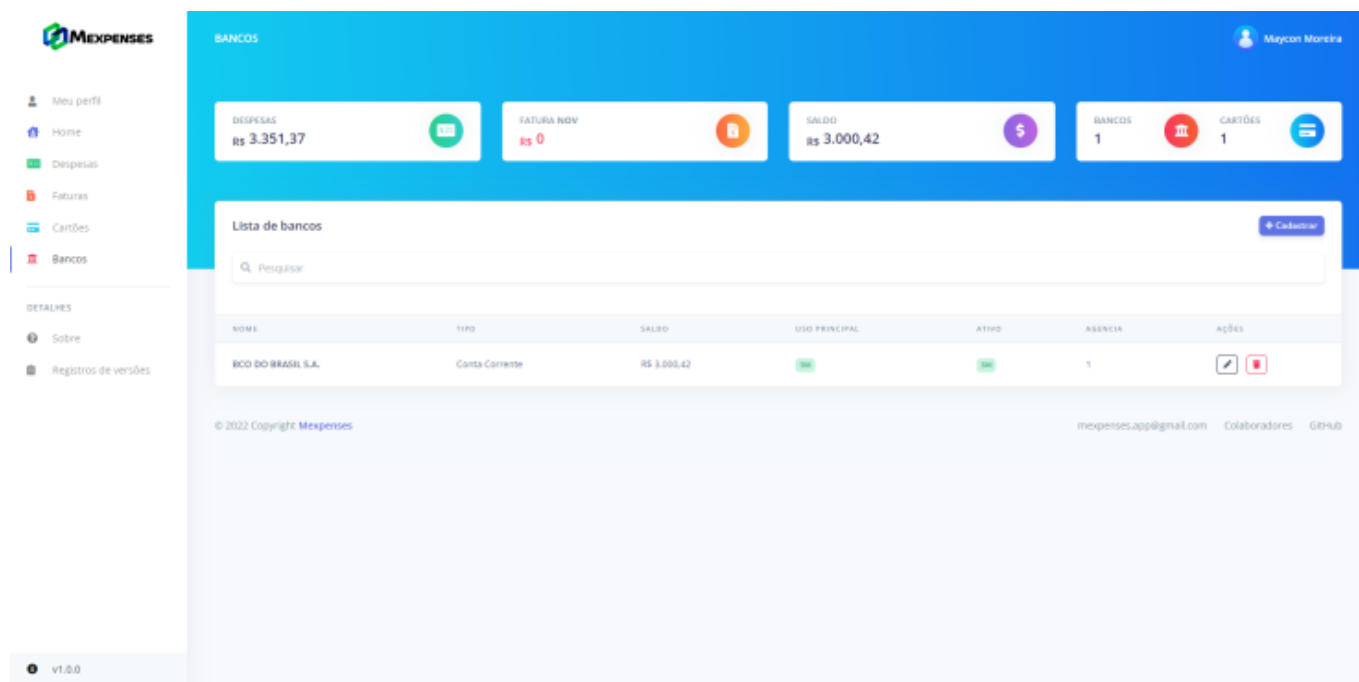
Figura 18- Cartões



Fonte: O autor

Conforme a figura 18, a tela possui os cartões cadastrados pelo cliente com um campo de pesquisa onde é possível filtrar um determinado cartão. Também é possível cadastrar com as suas respectivas informações como nome do banco, número do cartão, data de vencimento e código de segurança.

Figura 19- Bancos



Fonte: O autor

Conforme a figura 19, a tela possui um campo de pesquisa onde é possível filtrar uma determinada instituição financeira pelo seu nome, além de ser possível cadastrar um novo banco com seus respectivos campos como nome, tipo de conta, saldo disponível em conta, se esse banco é de uso principal, e a agência

4. REFERÊNCIAS

FLASK - **Flask Documentation (2.0.x)**, 2022. Disponível em < <https://flask.palletsprojects.com/en/2.0.x/>>. Acesso em 01 de abr. de 2022.

JAVASCRIPT - **MDN Web Docs**, 2022. Disponível em < <https://developer.mozilla.org/pt-BR/>>. Acesso em 01 de abr. de 2022.

MYSQL – **Database Service**, 2022. Disponível em < <https://www.mysql.com/>>. Acesso em 01 de abr. de 2022.

PYTHON – **Welcome to Python**, 2022. Disponível em < <https://www.python.org/>>. Acesso em 01 de abr. de 2022.

REACT - **Learn once, write anywhere**, 2022. Disponível em < <https://reactjs.org/>>. Acesso em 01 de abr. de 2022.

SASS- **SASS, S. A. S. S.** , 2002. Disponível em <<https://sass-lang.com>>. Acesso em 19 de mai. de 2022

BOOTSTRAP- **Bootstrap Documentation**, 2022. Disponível em < <https://getbootstrap.com/>>. Acesso em 19 de mai. de 2022.

FONTAWESOME- **FontAwesome Documentation**, 2022. Disponível em < <https://fontawesome.com/>>. Acesso em 19 de mai. de 2022.

GULP- **Gulp Documentation**, 2022. Disponível em <<https://gulpjs.com/>>. Acesso em 19 de mai. de 2022.

CHART- **Chart Documentation**, 2022. Disponível em <<https://www.chartjs.org/>>. Acesso em 19 de mai. de 2022.

ARGON-**Argon Documentation**, 2022. Disponível em < <https://demos.creativetim.com/argon-dashboard-react/#/documentation/quick-start>>. Acesso em 19 de mai. de 2022.