

PG Finanças: Plataforma para contribuir com gerenciamento financeiro pessoal

Mateus Oliveira Dias Viana, André Constantino da Silva

Curso Superior de Tecnologia em Análise e Desenvolvimento de Sistemas
Instituto Federal de Ciência e Tecnologia de São Paulo – (IFSP)
Campus Hortolândia – SP – Brasil

mateus.oliveira@aluno.ifsp.edu.br, andre.constantino@ifsp.edu.br

***Abstract.** Recently we have experience ups and downs in relation to the debts of the Brazilian people, which could be mitigated with financial tools to assist in the day by day and thus get around adverse situations and be able to better plan future actions in relation to the use of money. Considering the importance and the current use of the technologies of computers, laptops, tablets and smartphones, developed a web application for financial management with a modern, fast and responsive technology, finding a better experience when using the application that helps with personal finances, ensuring a better control of all the users' financial environment.*

***Resumo.** Recentemente tivemos experiências de altos e baixos em relação às dívidas do brasileiro, o que poderia ser amenizado com ferramentas de finanças para auxiliar no dia a dia e assim conseguir contornar situações adversas e poder planejar melhor as ações futuras em relação ao uso do dinheiro. Considerando a importância e o atual uso das tecnologias de computadores, laptops, tablets e celulares inteligentes, desenvolveu uma aplicação web para o gerenciamento financeiro com uma tecnologia moderna, rápida e responsiva, encontrando uma melhor experiência ao utilizar a aplicação que auxilia com as finanças pessoais, garantindo um melhor controle de todo âmbito financeiro dos usuários.*

1. Introdução

O desemprego no Brasil traz consequências como, por exemplo, as dívidas. Flávia Albuquerque (2018) apresenta um índice de contas atrasadas dos brasileiros com idades entre 18 e 95 anos chegando a 40,5% e, com o passar do tempo, essas pessoas criaram mais dívidas por não conseguirem quitá-las no prazo e, conseqüentemente, ficando com o seu CPF negativado e com dificuldades em obter crédito.

O IBGE (2020) divulga resultados de um estudo realizado pela Pesquisa Nacional por Amostra de Domicílios Contínua (PNAD Contínua), apresentando a taxa média de desocupações (desempregos) no Brasil, onde houve uma queda de 12,3% em 2018 para 11,9% em 2019; segunda queda anual consecutiva. Entretanto, no segundo trimestre de 2020 o IBGE apresenta uma taxa de desocupação de 12,75% dos brasileiros; esse aumento brusco nos índices tendo como grande influência a pandemia de COVID-19, compromete a economia.

A partir de toda situação em relação a dívidas que encontramos atualmente no Brasil, a necessidade de uma ferramenta para auxiliar no planejamento e gerenciamento financeiro se torna uma necessidade. Atualmente, podemos encontrar algumas ferramentas *on-line* para o gerenciamento financeiro como o Guiabolso, Organize e a Mobills (2020). Este trabalho tem o propósito de criar uma solução *web* visando auxiliar pessoas com a gestão financeira, permitindo que consigam alcançar seus objetivos, como controle de gastos e receitas, educação financeira básica para ajudar nos primeiros passos, definição de metas individuais, possibilidade de compartilhamento de objetivos/metast em grupo e que os usuários possam

colaborar inserindo no *board* as estratégias/tarefas e as pequenas conquistas referentes ao objetivo final. Tudo isso sendo exibido por gráficos para que possa ser visualizado por todos a evolução do que foi planejado pelo grupo.

Na Seção 2, apresenta-se conceitos em relação ao desenvolvimento de um sistema web, a saber, a divisão do sistema em duas partes, o *front-end* e o *back-end*, e *frameworks* para desenvolvimento de sistema para esta plataforma. A justificativa para o desenvolvimento deste trabalho é apresentada na Seção 3, sendo o objetivo geral e os objetivos específicos apresentados na Seção 4. O Guiabolso, Organizze e a Mobills, trabalhos correlatos a este, são apresentados na Seção 5. A metodologia escolhida para o desenvolvimento é apresentada na Seção 6, enquanto que o desenvolvimento do trabalho são apresentados nas Seções 7 e 8, esta última enfocando a implementação da aplicação. As conclusões são apresentadas na Seção 9, que encerra este artigo.

2. Fundamentação teórica

Essa Seção apresenta algumas definições de forma progressiva do que está sendo utilizado para desenvolver o produto final deste trabalho. Com isso, se tem mais objetividade e clareza de conceitos e tecnologia que estão sendo discutidos. Iniciamos apresentando a divisão arquitetural de uma aplicação *web* em dois componentes: o *front-end* e o *back-end*. Sabendo de tecnologias específicas para cada um desses componentes, são apresentados alguns *frameworks* empregados no desenvolvimento do software desenvolvido neste trabalho.

2.1. Front-end

Tudo o que se pode visualizar ou até mesmo interagir com o produto final é conhecimento como *front-end*, como diz Robbins (2012, p.9) “Frontend se refere a qualquer aspecto do processo de design que aparece ou se relaciona diretamente ao navegador”. Com isso tivemos uma evolução criando conceitos de UI (*User Interface* – Interface do usuário) para que seja melhor efetiva a interação do humano com a máquina, podendo ir além, como diz Arty (2015) “UI Design é sobretudo como será realizada interação entre o usuário e o meio (que no nosso caso é a interface gráfica), indo muito além da estética/aparência do projeto”.

Para o desenvolvimento dessa aplicação, foram utilizados os conceitos do *Material Design* criado pela Google (2014) em todo o processo de desenvolvimento de *front-end*. Essas técnicas fazem o uso mais liberal de layouts baseados em *grids*, animações e transições responsivas, preenchimentos, e efeitos de profundidade como luzes e sombras.

2.2. Back-end

Huamin Chen e Prasant Mohapatra (2004) também comenta sobre o *back-end* e afirma que ele concentra as regras de negócio no servidor, permitindo que as telas se tornem mais dinâmicas e interativas. O *back-end* possui basicamente uma aplicação que se comunica com o banco de dados, essa aplicação contém todas as regras de negócios e manipulação dos dados, como é dito por Cesar e Maykon (2016) “O *back-end* é um termo utilizado para representar a etapa final de um processo, na Web, ele é conhecido por compor as funcionalidades internas do servidor”.

O *back-end* desta aplicação foi desenvolvido utilizando o Node.js, que de acordo com Lenon (2018) existe muita flexibilidade, leveza e produtividade ao se utilizar o Node.js e também ele possui a mesma sintaxe do *front-end* que é o *javascript*.

2.3. Banco de dados

Um banco de dados é um local estruturado para que se possa armazenar, ler e manipular os dados, como se torna mais funcional a aplicação de uma determinada empresa, onde segundo DATE (2004, p. 10), “Um banco de dados é uma coleção de dados persistentes, usada pelos

sistemas de aplicação de uma determinada empresa”. Para essa aplicação foi utilizado o PostgreSQL, um sistema de gerenciamento de banco de dados (SGBD), onde serão armazenados todos os dados para o melhor funcionamento da aplicação.

2.4. *Frameworks* para desenvolvimento de software

Um *framework* é um compilado de componentes prontos dentro de uma estrutura, que serve como base para o desenvolvimento de novos projetos, nele estão contidas soluções de problemas comumente encontrados durante toda etapa de desenvolvimento e com isso se ganha tempo no processo, dando mais foco em novas funcionalidades. Como é dito por Pagani (2011), Maldonado (2012), Bueno (2017) e Cândido (2017), podemos considerar *framework* como recursos computacionais para o desenvolvimento de aplicações *web*, que disponibilizam ferramentas que auxiliam na solução de problemas de natureza semelhante a experiências prévias.

Como *framework* de *front-end* utilizado neste projeto decidiu-se pela adoção do Angular, onde podemos encontrar diversas funcionalidades importantes para o desenvolvimento de uma aplicação *web* seguindo o conceito de *Single Page Application* (SPA), onde Madhuri A. Jadhav, Balkrishna R. Sawant Anushree Deshmukh (2015), afirma que o SPA é composto de componentes individuais que podem ser substituídos ou atualizados independentemente. Para o *back-end* foi selecionado o *framework* Express, que basicamente é uma estrutura relativamente pequena que fica em cima da funcionalidade do servidor web do Node.js para simplificar suas APIs (*Application Programming Interface* – Interface de Programação de Aplicativos) e adicionar novas funcionalidades úteis (Evan, 2016).

2.5. *Websocket*

Segundo o Pedro (2019) o protocolo *WebSocket* permite a comunicação bidirecional entre uma aplicação cliente com o código em um ambiente controlado e um servidor remoto que tem a possibilidade de optar por comunicações provenientes do cliente, melhorando assim o desempenho e solucionando problemas de comunicação existentes no mercado de aplicações *web*. Para este projeto decidiu-se o uso do Socket.io, uma biblioteca *Javascript* para realizar comunicações utilizando o protocolo *websocket*.

3. Justificativa

De acordo com os dados apresentados anteriormente, compreende-se que uma ferramenta que possibilite ao usuário se organizar financeiramente se torna de grande ajuda, permitindo-o refletir sobre a necessidade de alguns gastos, auxiliando-o a alcançar uma estabilidade financeira e podendo ir além com o planejamento em grupo de algo ou alguma coisa que é de comum acordo de todos os integrantes.

Exemplo de planejamento em grupo:

- **Objetivo geral:** Viagem em família para casa da avó Neusa.
- **Data criação do planejamento:** 30/11/2020
- **Data limite:** 15/03/2021
- **Custo da viagem:** R\$ 10.000,00
- **Custo pré-viagem:** R\$ 3.000,00
- **Reserva emergencial:** R\$ 2.000,00
- **Participantes:** José, Florisvalda, Lucas e Marilda.
- **Objetivos por pessoa:** cada integrante tem como meta acumular R\$ 2.500,00 para a viagem
 - José:
 - **Custo pré-viagem:** Revisão do carro (R\$ 1.500)

- Guardar R\$ 625,00 do seu salário todos os meses, iniciando em 05/12/2020
- Florisvalda:
 - **Custo pré-viagem:** compras no supermercado (R\$ 500,00)
 - Guardar R\$ 625,00 do seu salário todos os meses, iniciando em 05/12/2020
- Lucas:
 - **Custo pré-viagem:** comprar bebidas (R\$ 250,00)
 - Guardar R\$ 300,00 do seu salário todos os meses, iniciando em 05/12/2020
 - Fazer renda extra em transporte por aplicativo, com objetivo de acumular R\$ 433,33 por mês, iniciando no dia 01/12/2020
- Marilda:
 - **Custo pré-viagem:** comprar presentes para a avó Neusa (R\$ 750,00)
 - Guardar R\$ 625,00 do seu salário todos os meses, iniciando em 05/12/2020

Esta será uma forma de planejamento, em que os integrantes do grupo se organizam melhor na distribuição das tarefas.

A aplicação também prevê a disponibilização de um módulo para educação financeira, com simulações de investimentos conservadores, por exemplo: Certificado de Depósito Bancário (CDB), Certificado de Depósito Interbancário (CDI), tesouro SELIC etc. De acordo com Savoia (2007) em “Paradigmas da Educação Financeira no Brasil”, com o passar do tempo, a importância da educação financeira no país tem aumentado cada vez mais, porém ainda possui uma grande deficiência em comparação com outros países, que lidam com a educação financeira abrindo discussões sobre como melhorar os conhecimentos ao utilizar o seu dinheiro.

4. Objetivos

4.1. Objetivos gerais

Este trabalho tem como objetivo desenvolver uma aplicação *web* para auxiliar no controle e gestão financeira de uma pessoa, sendo uma ferramenta para facilitar esse processo.

4.2. Objetivos específicos

A aplicação se propõe em auxiliar no gerenciamento e controle das finanças do usuário, tendo uma estrutura pronta para que se possa personalizar a sua vida financeira, além do auxílio para dar os primeiros passos nas tomadas de decisões.

A plataforma disponibiliza uma estrutura de personalização no momento de criação das receitas e despesas, podendo criar ganhos fixos mensais, semanais e/ou diários, rendas extras, entre outros tipos de receitas. Podem ser criados custos fixos, parcelas de compras, fatura de cartões etc. para as despesas mensais, semanais e/ou diárias.

A criação de objetivos de curto, médio e longo prazo faz com que se tenha uma melhor organização para o futuro. A aplicação irá auxiliar oferecendo dicas de como e onde aplicar o dinheiro e assim alcançar mais rápido esses objetivos, apresentando estratégias para economia do dinheiro e tomadas de decisão para obter uma renda extra.

O planejamento financeiro compartilhado é a possibilidade de criar um objetivo com outras pessoas que possuem acesso ao sistema, assim será possível que um grupo de pessoas contribuam e vejam em tempo real toda a estrutura do planejamento sendo criada. Ao finalizar, estará disponível para todos os contribuintes darem baixa no que já foi cumprido.

A aplicação, além de oferecer um ambiente de controle financeiro, auxilia a se preocupar com suas receitas e, principalmente, com as despesas. Com isso, o foco é auxiliar da melhor forma nas tomadas de decisões.

5. Trabalhos correlatos

Atualmente no mercado podemos encontrar algumas alternativas para o gerenciamento financeiro e também diferentes modelos de abordagens, como as aplicações onde o usuário possui autonomia para alimentar com os dados que julga necessário para o seu controle financeiro, tudo feito de forma manual. Outro modelo são as aplicações que realizam integrações com o seu cartão de crédito/débito e apresentando em uma interface o histórico de movimentações realizadas.

Uma aplicação bastante popular é o Organizze (2020), cuja tela principal é exposta pela Figura 1, onde segue o modelo de controle manual dos dados financeiros. Ele possui uma interface simples e objetiva mostrando informações rápidas em sua tela inicial. Com ele é possível criar receitas, despesas, cadastrar contas e também cartões, permitindo efetuar a gestão de todas as suas finanças. A aplicação conta com relatórios bastante completos onde é possível consultar todas as suas receitas e despesas em diversos períodos, como por exemplo: as receitas e despesas de hoje, desta semana, deste mês, nos últimos 3, 6 ou 12 meses, deste ano e também é possível personalizar um intervalo de tempo. Os dados são apresentados em um gráfico e com uma lista mais detalhada logo abaixo com todas as receitas e despesas. Com o Organizze também é possível estabelecer metas onde é possível estabelecer um valor máximo de despesas e um valor que deseja alcançar com suas receitas.

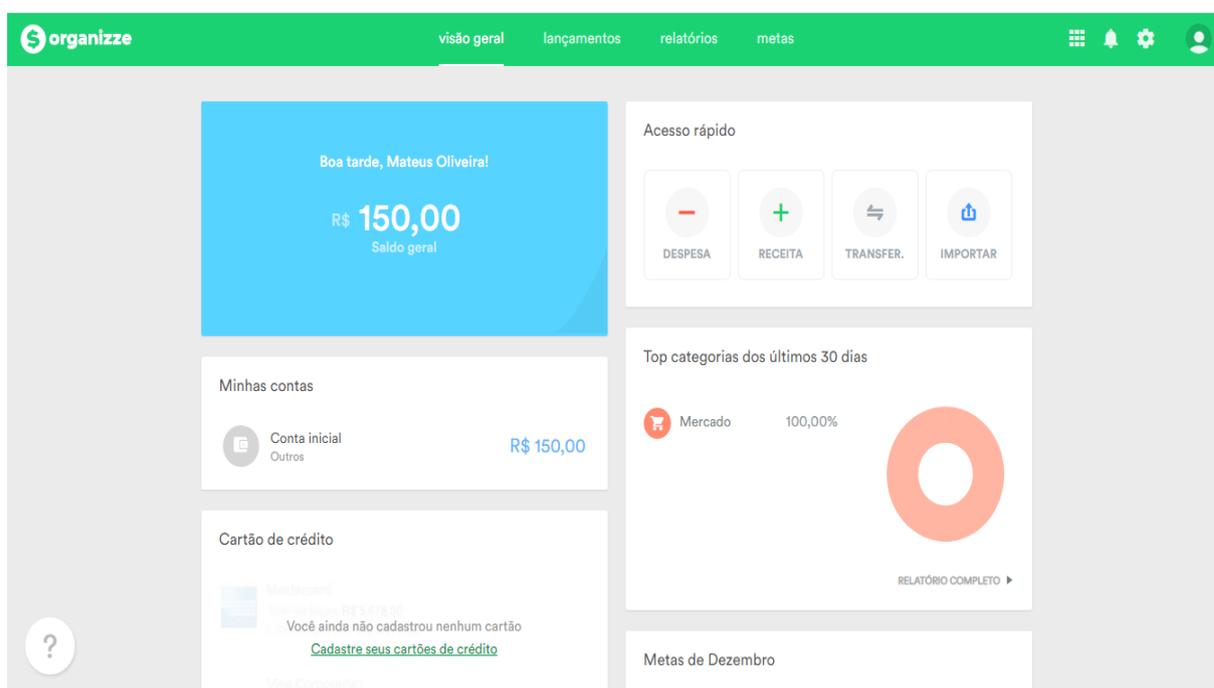


Figura 1. Aplicação web Organizze (2020).

Outra aplicação que segue o mesmo modelo do Organizze é o Mobills (2020), cuja tela principal é exposta na Figura 2, onde todo controle e gestão das finanças é feito de forma manual e possui algumas funções a mais, como por exemplo:

1. Mobills educação: aqui são apresentados *posts* onde são abordados diversos assuntos na área de economia e, principalmente, educação financeira com dicas, tendências de mercado etc.;

2. Criação de planejamentos: com essa função é possível se organizar melhor nos planejamentos e assim melhorar as estratégias para alcançá-los;

3. Programa de pontos: possibilita ao usuário acumular pontos seguindo algumas regras estabelecidas pelo aplicativo e trocá-los por curso de planejamento financeiro, livros, contra *premium* ou *gift cards* (cartão presente – um cartão pré-pago onde se pode trocar por produtos ou serviços);

4. Download: é possível realizar o *download* das transações em formato Excel ou PDF.

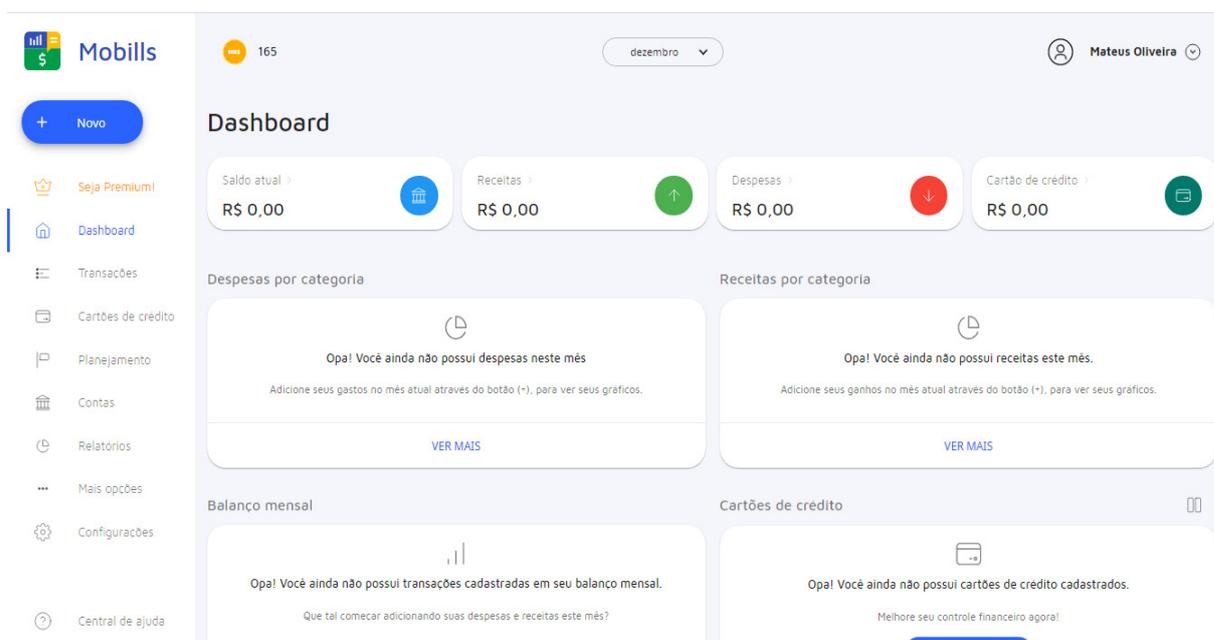


Figura 2. Aplicação web Mobills (2020).

Seguindo o outro modelo, onde ocorrem integrações com outras aplicações, existe o Guia Bolso (2020), cuja tela principal é exposta na Figura 3. Uma aplicação com um intuito de deixar o processo de controle financeiro mais automatizado, sendo possível vinculá-lo com uma conta bancária e, assim, obter o histórico de transações realizadas com o cartão de crédito, débito, transferências efetuadas e recebidas. Essa integração poupa bastante tempo em relação ao Organizze e ao Mobills, entregando alguns serviços adicionais, como empréstimos e aquisição de um cartão.

Todas as aplicações citadas trazem diversos recursos e serviços para agregar no controle financeiro de rotina. A aplicação *web* proposta neste trabalho visa a objetividade com informações relevantes ao perfil do usuário e, diferentemente das aplicações citadas anteriormente, possibilitar a criação de grupos com outros usuários e aprimorar o planejamento coletivo.

6. Metodologia

Para o desenvolvimento do software foi adotado a metodologia Scrum, visando entregas rápidas para agregação de valor ao produto, de acordo com Thiago Bessa e Daniel Dias Banco Arthaud (2018). Unger e Chandler (2010) definem *User eXperience*, como a criação e sincronização de elementos que afetam a experiência do usuário de modo a influenciar suas percepções e comportamento. Com isso, durante o desenvolvimento foram aplicadas boas práticas de *User eXperience* (UX), como: fluidez ao navegar na aplicação, velocidade de resposta, ações rápidas

como criar uma nova receita por exemplo etc. E assim alcançando o público-alvo, sendo eles usuários na faixa etária de 15 a 65 anos.

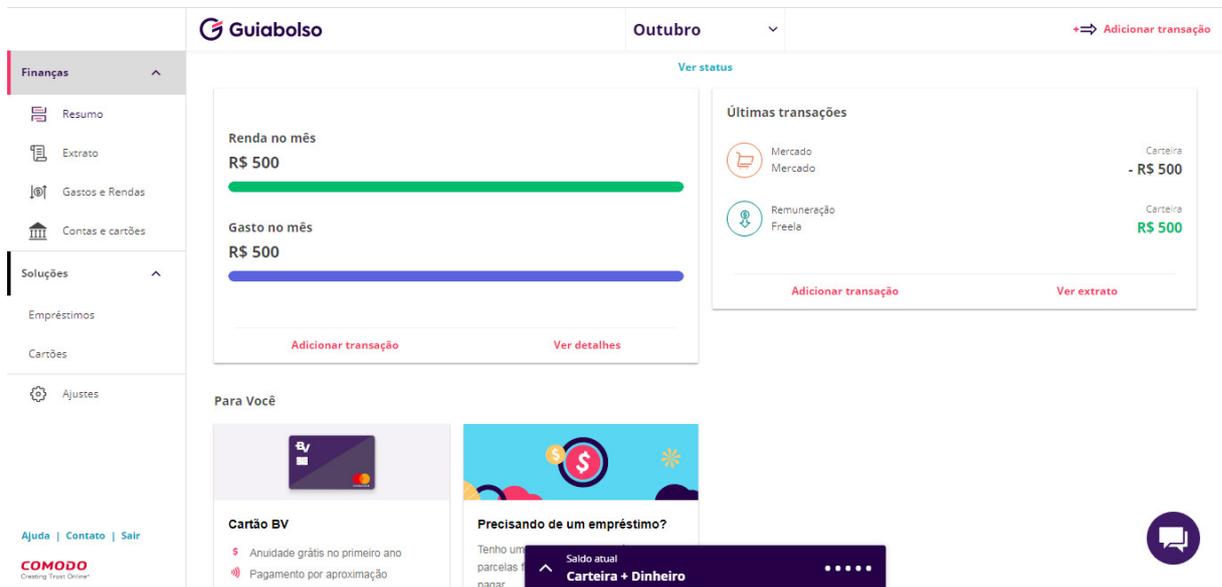


Figura 3. Aplicação web Guia Bolso (2020).

A interface conta com os conceitos do *Material Design*, que é basicamente um *guideline* de estilos criados pela Google, a fim de padronizar todo o visual das páginas *web* do seu ecossistema, e para auxiliar em todo o processo de desenvolvimento será utilizado a biblioteca *Angular Material*.

O desenvolvimento foi dividido em 4 *sprints* com um mês de duração cada. Ao final de cada ciclo foi feita uma análise com o Dr. André Constantino da Silva, orientador deste trabalho e especialista em Interação Humano-Computador, para saber se os requisitos de UX citados anteriormente foram atingidos.

6.1. Backlog do produto

O Backlog é uma lista de requisições para o produto que ainda não foram produzidas. Ele conta com algumas funcionalidades essenciais para o produto final, e com isso foi dividido em *sprints*, sendo elas:

- **Controle de receitas e despesas;**
 - Manter fontes de renda
 - Manter receitas;
 - Manter despesas;
 - Filtrar fontes de rendas, receitas e/ou despesas;
- **Planejamento em grupo *realtime*:**
 - Criação de grupo(s);
 - *Realtime* com Socket.io;
 - Objetivo final do grupo;
 - Objetivos individuais
- **Objetivos:**
 - Manter objetivos;
 - Aviso de que foi alcançado esse objetivo;
 - Filtrar objetivos;
- **Educação financeira:**

- Manter temas;
 - Manter *posts*;
- Filtrar tema e/ou *posts*;

6.2. Materiais e métodos

1. *Back-end*:

- a. Node.js com TypeScript;
- b. Socket.io;

2. *Front-end*:

- a. HTML5;
- b. SCSS;

3. Banco de dados:

- a. PostgreSQL;

4. *Framework/bibliotecas front-end*:

- a. Angular;
- b. Angular Material;
- c. Material Icons;

6.3. Ferramentas

As ferramentas utilizadas para o desenvolvimento do *front-end* e *back-end* foram as descritas a seguir. O Visual Studio Code, onde foram construídas todo o código fonte da aplicação. Para visualizar o banco de dados, foi utilizado o PgAdmin 4 como interface visual do PostgreSQL.

7. Desenvolvimento

Para o desenvolvimento da aplicação foram elaborados alguns diagramas, para um melhor entendimento de todos os requisitos, sendo o diagrama de caso de uso e o diagrama de classes da UML (2006) e o diagrama de entidade e relacionamento (2010).

7.1. Diagrama de caso de uso

Começando com o Diagrama de Casos e Uso (Figura 4) pode-se obter uma visão macro do projeto e o que os atores têm acesso. Primeiramente é importante saber a definição de um caso de uso, onde segundo Melo (2010, p.57) “um caso de uso descreve uma sequência de ações que representam um cenário principal (perfeito) e cenários alternativos, com o objetivo de demonstrar o comportamento de um sistema (ou parte dele), através de interações com atores”. Em outras obras também podemos encontrar mais definições como a de Booch, Rumbaugh e Jacobson (2005, p. 231) “um caso de uso é uma descrição de um conjunto ou sequência de ações, inclusive variantes, que um sistema executa para produzir um resultado de valor observável por um ator.”.

A seguir, apresentamos brevemente os principais casos de uso. O caso de uso “Navegar em educação financeira” acontece quando o usuário acessa o módulo de educação financeira, que contém diversos *posts* sobre diversos assuntos na área de finanças. Não é necessário estar autenticado para acessar esse módulo.

O caso de uso “Manter fonte de renda” é uma ação onde o usuário tem a possibilidade de criar fontes de rendas e dentro delas ter suas estruturas de receitas e despesas, sempre tendo

como resultado principal o valor total desse escopo criado, sendo ele positivo ou negativo. Essa funcionalidade só é permitida para usuários autenticados na aplicação.

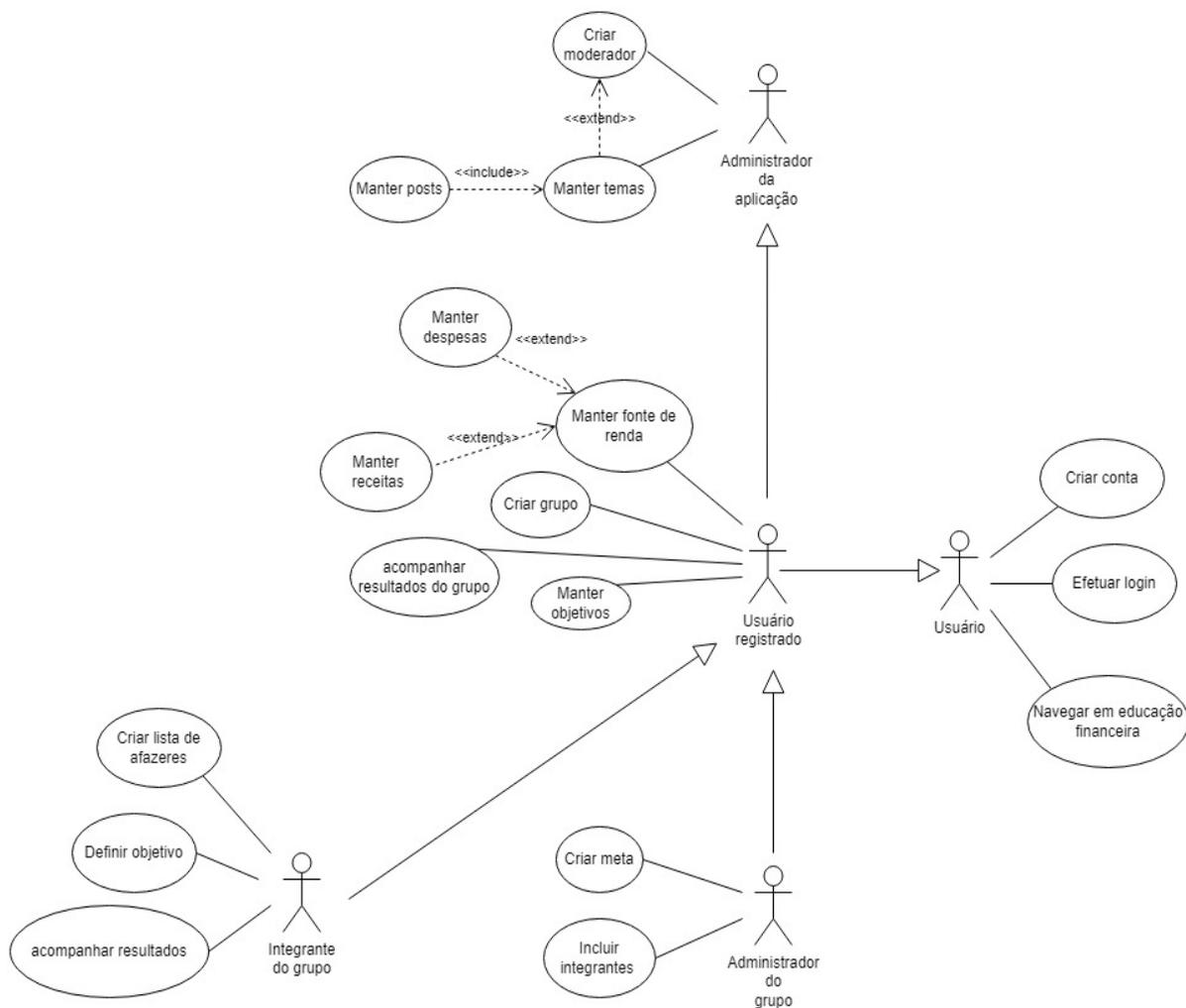


Figura 4. Diagrama de caso de uso para expor ações do usuário na aplicação PG Finanças.

O caso de uso “Criar grupo” é uma ação onde o usuário pode criar um planejamento em grupo com outras pessoas registradas na aplicação e assim ter um ambiente *real time* para ter um gerenciamento melhor de tudo o que for discutido pelo grupo. Somente usuários autenticados na aplicação podem criar um grupo de planejamento.

O caso de uso “Criar lista de afazeres” é uma ação realizada somente por integrantes de um grupo. Cada integrante cria sua lista para sinalizar o que ele pretende fazer para alcançar a sua meta. Para esse tipo de ação é necessário que o usuário esteja autenticado na aplicação e faça parte de um grupo.

O caso de uso “Criar moderador” é uma ação realizada pelo administrador da aplicação onde é criado um moderador para que ele possa alimentar o módulo de educação financeira com novos temas e *posts*. Essa funcionalidade só pode ser executada por um super usuário que esteja autenticado na aplicação.

O caso de uso “Manter objetivos” é uma ação para criação de objetivos futuros na qual o usuário estabelece algo que deseja alcançar e o período para isso. Somente usuários autenticados têm acesso a essa funcionalidade.

7.2. Diagrama Entidade-Relacionamento (DER)

O diagrama de entidade-relacionamento (Figura 5) dessa aplicação apresenta uma visão macro de alto nível para representação do banco de dados. Como citado por Araújo (2008, p.34) “O Modelo de Entidade-Relacionamento é um modelo de alto nível, independente do SGBD (Sistema de Gerenciamento de Bancos de Dados), que representa o problema a ser modelado”.

Pode-se observar no diagrama de entidade-relacionamento (Figura 5) que as entidades receitas e despesas possuem os mesmos campos, pois são especializadas em finanças. A entidade de receitas representa valores adicionados que são acumulados de forma crescente e o mesmo acontece com as despesas, com os dois valores acumulados é feito um cálculo de receitas menos despesas para se obter o valor atual de sua carteira de finanças.

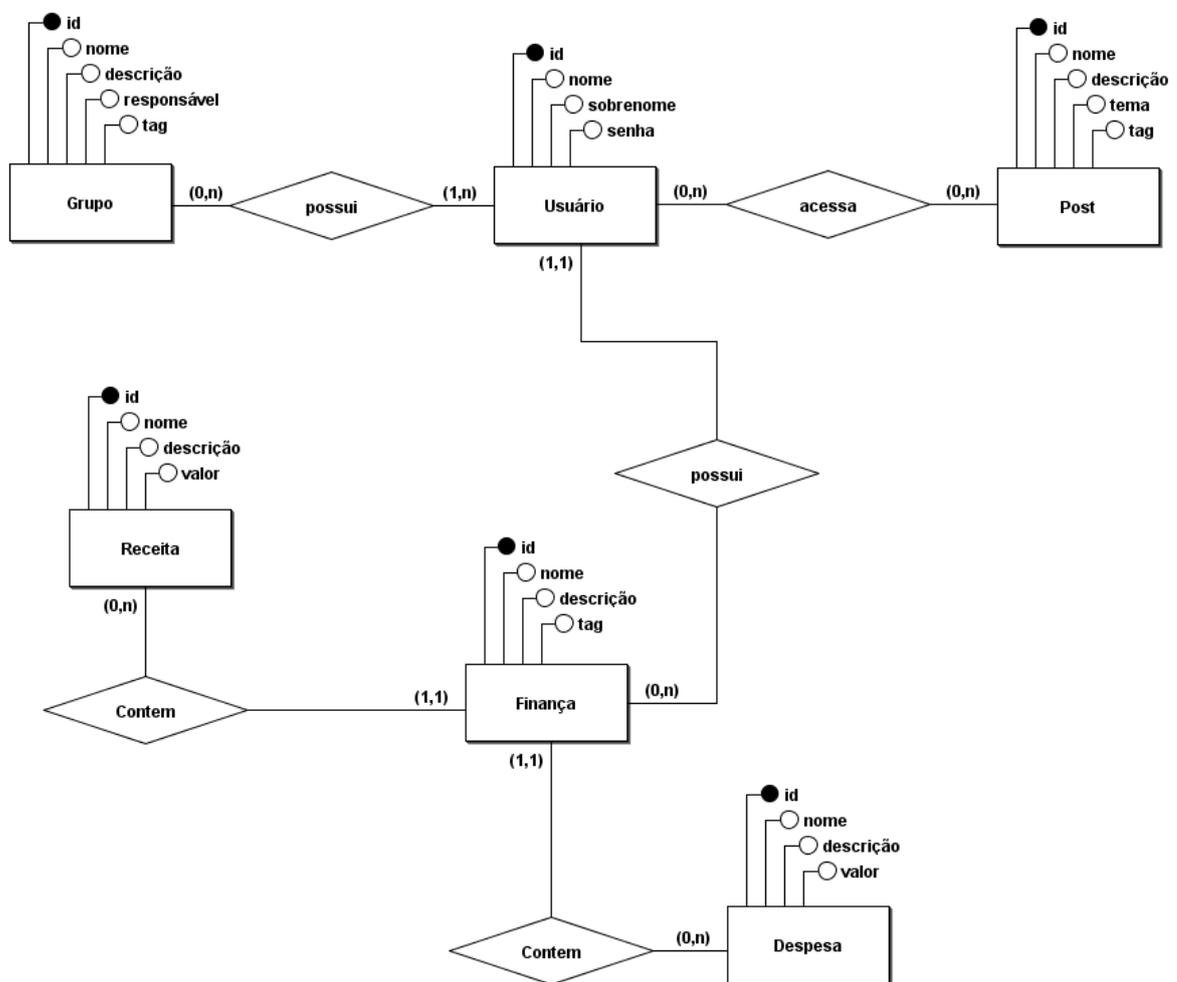


Figura 5. Diagrama Entidade-Relacionamento elaborado para o sistema PG Finanças.

8. Desenvolvimento da aplicação

Visando atender aos requisitos, a aplicação *web* de gerenciamento financeiro foi implementada abordando, de forma prática e descomplicada, uma gestão básica de finanças

peçoais. Com isso temos um *dashboard* para uma visão geral de como está a situação financeira do usuário no período corrente e áreas para a gestão financeira de fato, criação de objetivos, planejamentos e uma área para ampliar seus conhecimentos sobre gestão financeira e investimentos.

Com essa abordagem pretende-se proporcionar ao usuário uma melhor uma melhor experiência na hora de se organizar financeiramente, controlando melhor todos os gastos e assim o progresso de suas finanças ao longo do tempo.

8.1. Arquitetura

A arquitetura da aplicação (Figura 6) é baseada no padrão cliente-servidor, e a comunicação é via formato JSON, que de acordo com Ivan (2015) é um tipo de formato mais leve de dados para troca de informações entre aplicações, formato é estruturado em chave-valor.

O cliente envia dados via JSON para o servidor e ele processa esses dados interagindo com o banco de dados, fazendo as operações SQL. O Servidor, responde para o cliente (em formato JSON) para que ele possa exibir esses dados.

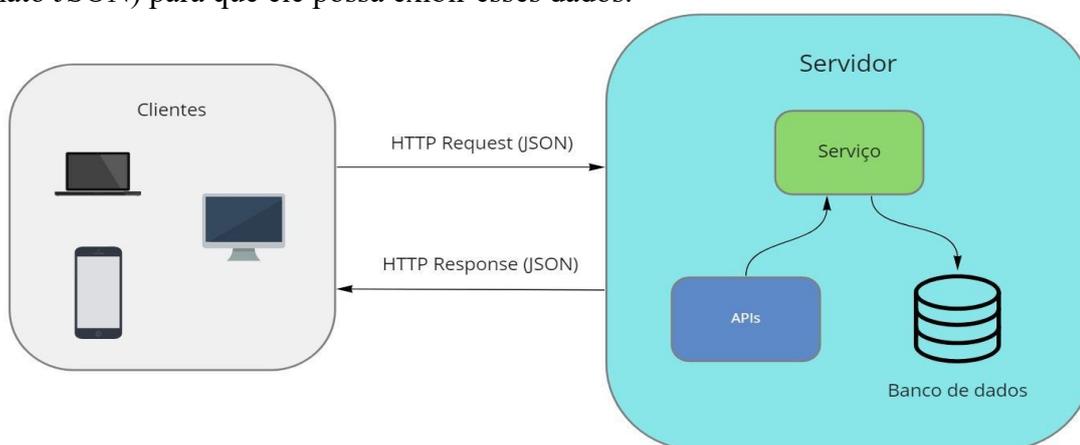


Figura 6. Arquitetura proposta para a aplicação PG Finanças.

8.2. Desenvolvimento do front-end

A *dashboard* (Figura 7) é a tela principal da aplicação, nela é possível ver o saldo total que o usuário possui e também a somatória das receitas e despesas do mês corrente. Essa tela conta com uma visão simplificada da situação corrente do usuário, na parte superior tem *cards* com indicadores de valores para o usuário, como:

1. **Saldo:** Esse indicador mostra todo o saldo que ele tem acumulado até o momento. Todo dinheiro que sobra mês a mês é incrementado nesse saldo.
2. **Receitas:** Esse indicador mostra as receitas do mês corrente, então todas as entradas que o usuário fizer durante o mês serão incrementadas nesse indicador.
3. **Despesas:** Ao contrário das receitas, esse indicador mostra todos os gastos registrados pelo usuário do mês corrente, então todo novo registro será incrementado o valor nesse indicador.

O gráfico apresentado na *dashboard* mostra uma visão do progresso que ocorreu no último ano do usuário, assim ele poderá fazer uma análise de como foi o seu percurso usando a aplicação e se efetivamente conseguiu se organizar melhor com suas finanças.

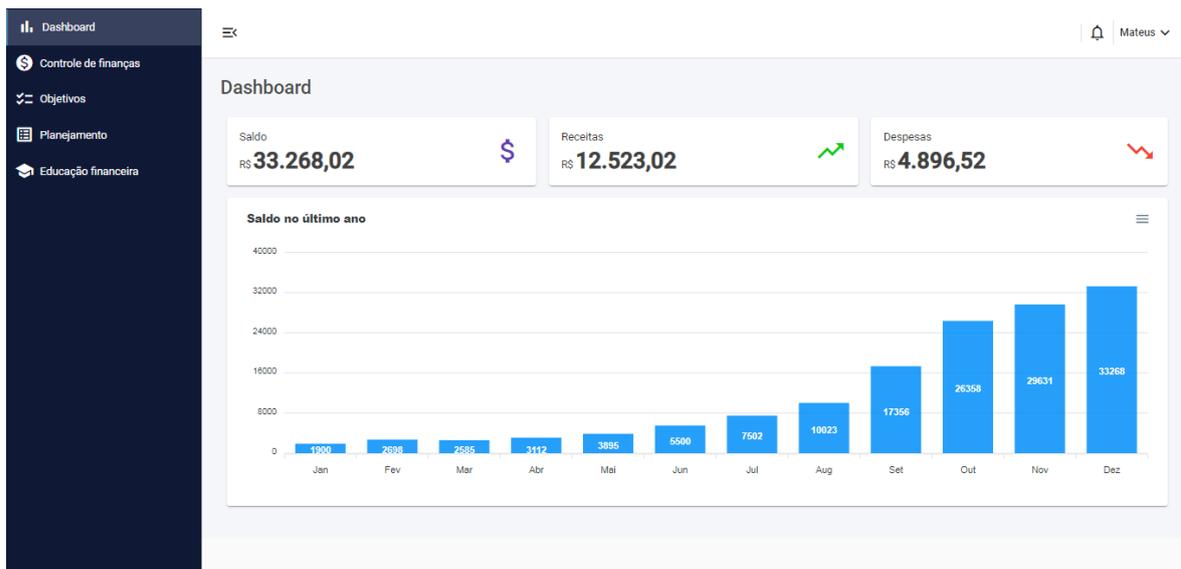


Figura 7. *Dashboard* da aplicação PG Finanças.

O Controle de finanças (Figura 8) é uma tela onde o usuário irá fazer todas as registros de receitas e despesas do mês corrente, contendo os mesmos indicadores encontrados na *dashboard*, além de um gráfico onde mostra todo o progresso do último ano, apresentando o total de receitas e despesas em todos os meses. Ao lado do gráfico encontra-se uma tabela onde são listadas todas as receitas e despesas registradas do mês corrente.

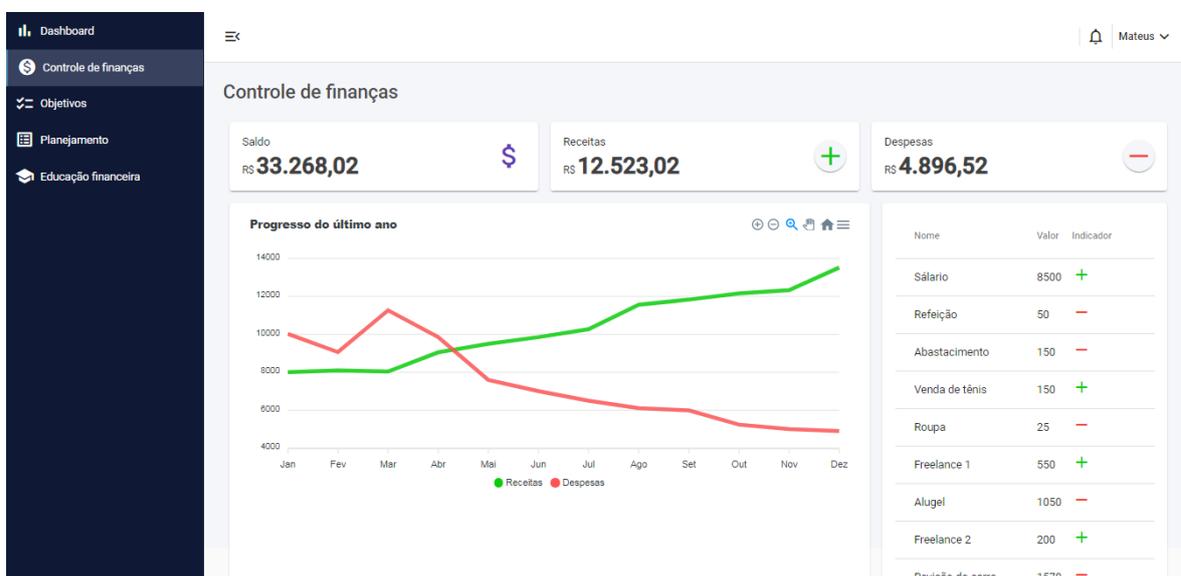


Figura 8. Tela de controle de finanças do PG Finanças.

Os *cards* de receitas e despesas possuem um botão para o registro de um novo indicador. Ao pressionar o botão é aberto um *pop-up* (Figura 9) com poucos campos para efetuar de forma rápida o registro. Ao clicar em qualquer um dos botões será exibido o mesmo *pop-up*, diferenciando somente o título e o processamento ao salvar.

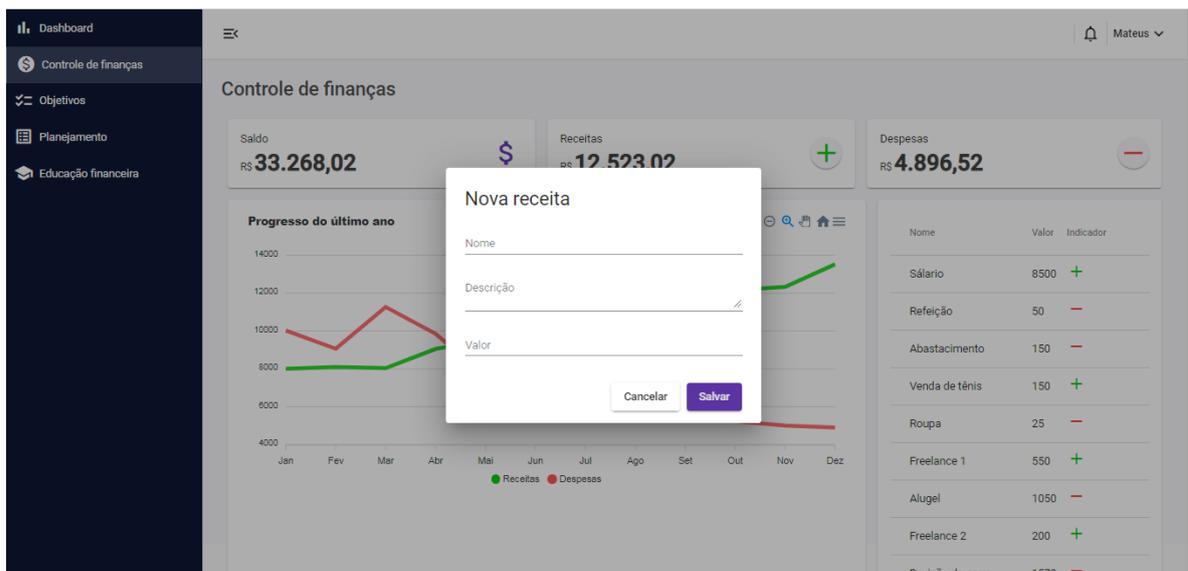


Figura 9. Pop-up para registro de nova receita.

Com a tela de objetivos (Figura 10) o usuário irá conseguir organizar todos os seus objetivos futuros tendo uma visão melhor do valor necessário mensal que ele terá que guardar para que possa alcançar o objetivo estipulado. Por exemplo, o usuário cria um objetivo de que daqui 1 ano ele quer comprar um celular novo e para isso ele precisa economizar R\$2.000,00 (dois mil reais) até lá. Ao registrar esses dados a aplicação irá exibir uma projeção futura para que se possa alcançar esse objetivo, com um gráfico mostrando o quando foi economizado até o período corrente e também a data final desse objetivo em questão. O usuário poderá realizar novas entradas todos os meses quantas vezes julgar necessário, até chegar ao objetivo final.

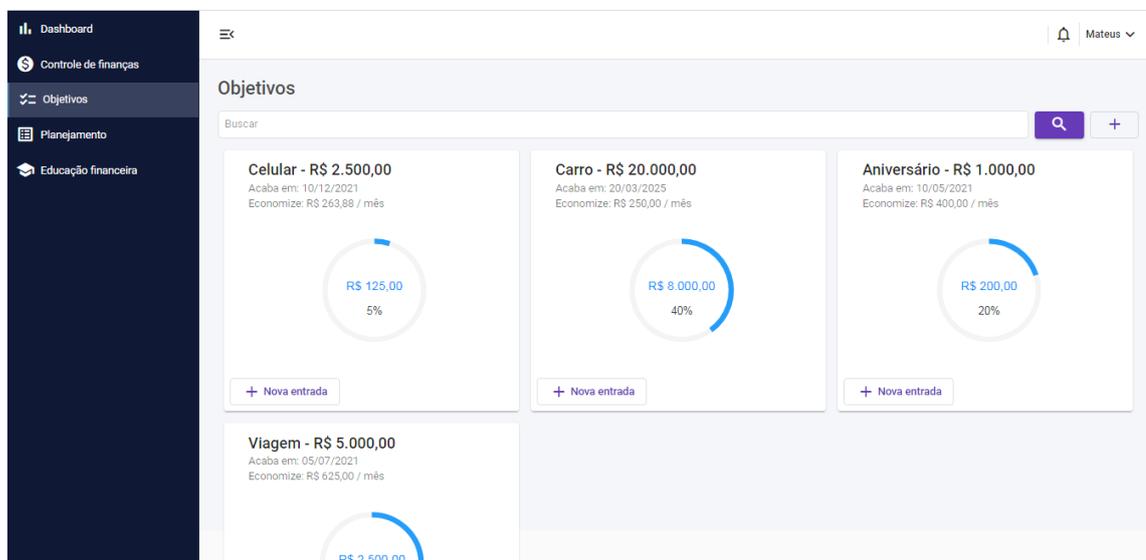


Figura 10. Tela para acompanhamento de objetivos da aplicação PG Finanças.

Com a tela de planejamento (Figura 11) o usuário pode planejar melhor todos os gastos mensais, criando limites para os mesmos, sendo possível ter de forma visual toda a projeção de gastos. Dessa forma se tem um melhor controle de todas as despesas que estão agrupadas em categorias, como por exemplo: alimentação, estudos, custos fixos, combustível etc.

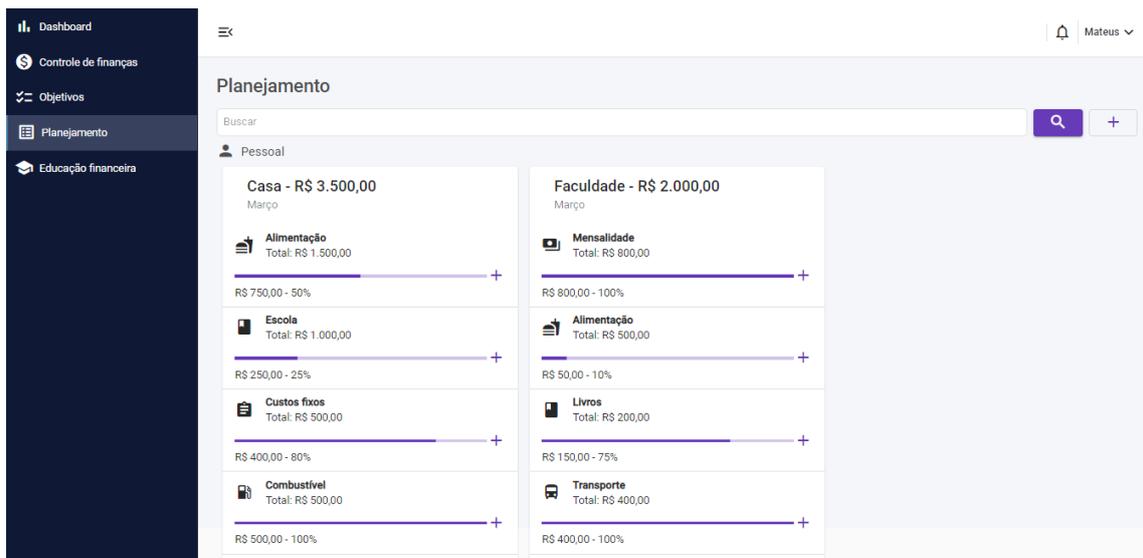


Figura 11. Tela de planejamento financeiro da aplica o PG Finan as.

A tela   dividida em duas  reas, sendo o planejamento pessoal e o planejamento em grupo. No planejamento pessoal, est o todos os planejamentos individuais do usu rio, com todas as categorias de despesas que ele possui. J  no planejamento em grupo (Figura 12) n o tem as categorias e sim os participantes do grupo e quanto cada integrante ir  contribuir, para que o objetivo geral do plano seja alcan ado. Todos os dados exibidos nos planejamentos em grupo ser o atualizados em tempo real para melhor controle do que foi acordado com o grupo.

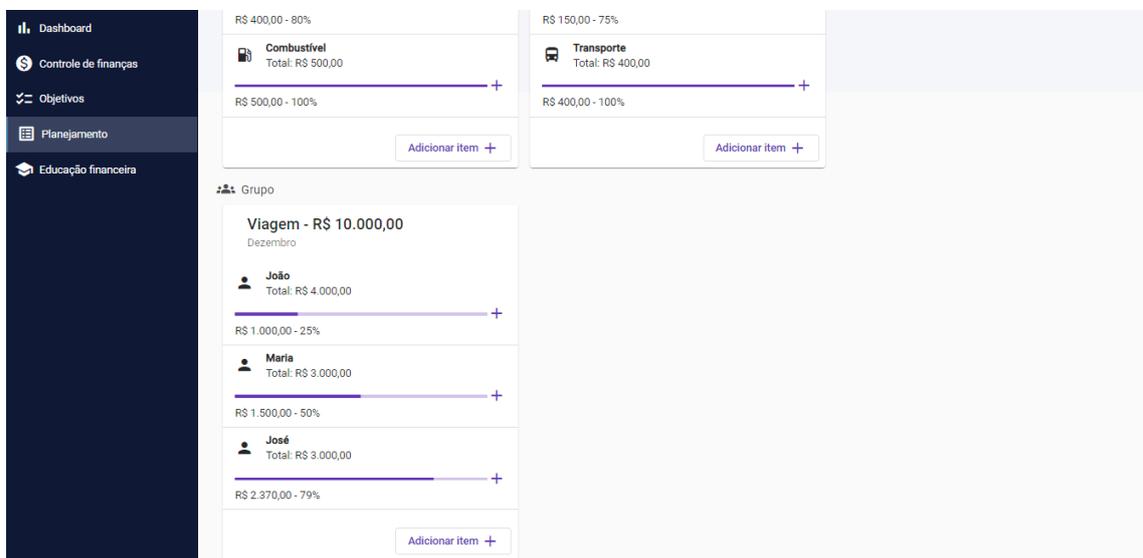


Figura 12. Tela de planejamento financeiro *real-time* em grupo do PG Finan as.

Com a tela de Educa o financeira (Figura 13) o usu rio poder  encontrar *posts* relacionados ao mundo financeiro e tamb m encontrar dicas e recomenda es sobre como administrar melhor o seu dinheiro, bem como formas de gerar renda extra e investimentos, aumentando assim o seu patrim nio. Essa  rea da aplica o   para que a pessoa tenha no es b sicas de como pode usar da melhor forma o seu dinheiro.

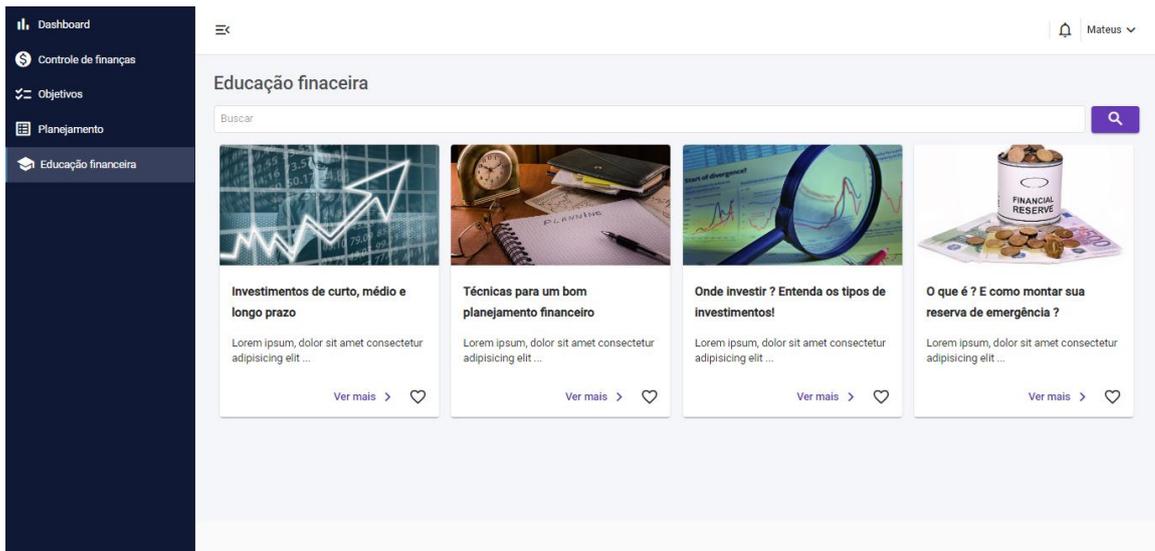


Figura 13. Tela de educação financeira da aplicação PG Finanças.

Pensando em uma melhor experiência para o usuário e que ele possa ter mais flexibilidade ao utilizar a aplicação, todas as telas estão responsivas, onde são reajustadas de acordo com o tamanho da tela, como mostrado na Figura 14.

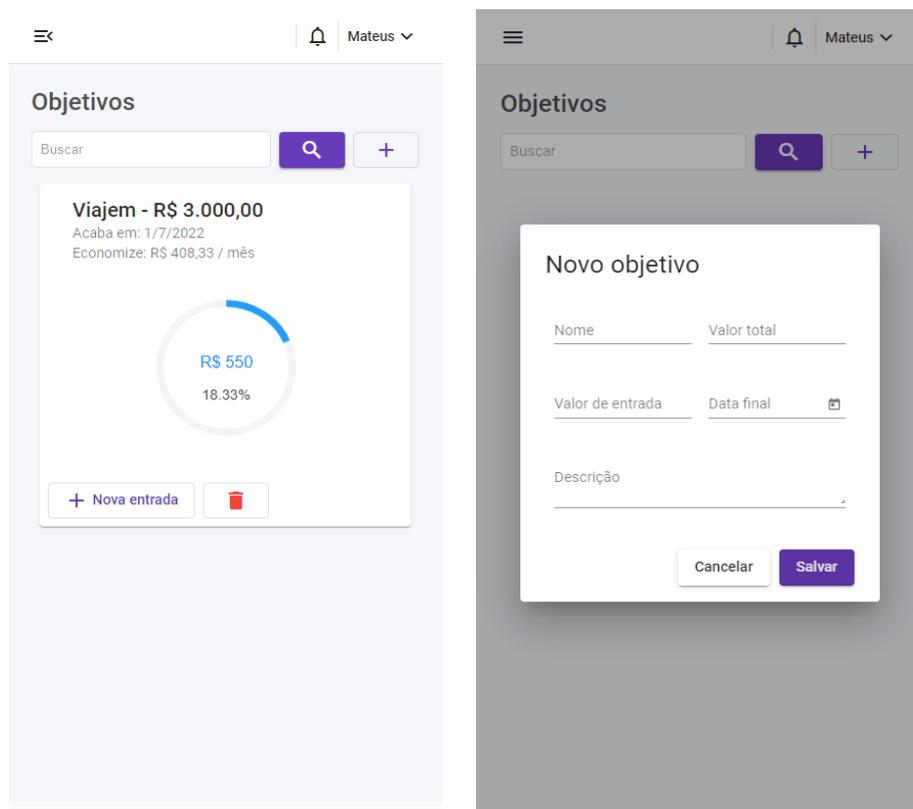


Figura 14. Tela de Objetivos da aplicação PG Finanças visualizada em um dispositivo móvel.

9. Conclusão

Esta aplicação *web* para gerenciamento financeiro visa entregar um melhor conforto e velocidade na gestão de finanças pessoais, de forma descomplicada e efetiva. Além de contar com uma área para estudos e entendimentos sobre assuntos relacionados a finanças.

Ao decorrer do desenvolvimento, houveram diversas limitações, como construir dinamicamente a estrutura JSON para os gráficos e a tipagem dos dados. Em relação aos testes de desenvolvimento foram usadas muitas horas para validar as APIs e garantir a consistência dos dados.

Conhecimentos adquiridos nas disciplinas de Linguagem de Programação, Banco de Dados, Estrutura de dados, Arquitetura de Software e Desenvolvimento de Sistemas Web auxiliaram em todos os processos de desenvolvimento da aplicação.

Acredito no potencial para um crescimento futuro podendo alcançar um número grande de usuários e se tornando uma plataforma acessível e descomplicada para todos. E pensando nisso, para realização de trabalhos futuros seria de extrema importância realizar testes com usuário reais, outro ponto seria melhorar ainda mais a performance para suportar uma grande quantidade de usuários simultâneos e também desenvolver o que foi idealizado sobre o planejamento em grupo.

Referências

- ALBUQUERQUE, F. Brasileiros com nome em cadastro de devedores chegam a 40,5% da população. Agência Brasil, Mar. 2018. Disponível em: <<https://agenciabrasil.ebc.com.br/economia/noticia/2018-03/brasileiros-com-nome-em-cadastro-de-devedores-chegam-405-da-populacao>>. Acesso em: maio 2020.
- ARAÚJO, M. A. P. Modelagem de dados – Teoria e prática. Saber Digital: Revista Eletrônica do CESVA, Valença; Volume 1, 2008.
- ARTY, D. UI Design - O que é User Interface Design (UI Design)?. Chief do Design, Ago. 2018. Disponível em: <<https://www.chiefofdesign.com.br/ui-design/>>. Acesso em: 20 maio 2021.
- BESSA, T.; ARTHAUD, D. D. B. Metodologias Ágeis para o Desenvolvimento de Software. **Revista Ciência e sustentabilidade**, Juazeiro do Norte, v. 4, n. 2, p. 173-213, jul./dez. 2018. Disponível em: <<https://periodicos.ufca.edu.br/ojs/index.php/cienciasustentabilidade/article/download/314/308/>>. Acessado em: dez. 2020.
- BOOCH, G.; RUMBAUGH, J.; JACOBSON, I. UML: guia do usuário. 6º Reimpressão. Rio de Janeiro, Elsevier, 2005.
- BUENO, D. C. HyMobWeb: uma abordagem para a adaptação híbrida de interfaces Web móveis sensíveis ao contexto e com suporte à multimodalidade. 2018. 173 f. Dissertação (Mestrado) - Curso de Programa de Pós-graduação em Ciência da Computação, Ciências Exatas e da Terra::Ciência da Computação, Universidade Federal de São Carlos, Sorocaba, 2017. Disponível em: <<https://repositorio.ufscar.br/handle/ufscar/9157>>. Acessado em: dez. 2020.
- CÂNDIDO, P. H. V. Problemas de interação na adaptação de interface Web móvel em Frameworks Front-end. Itapetininga, 2017. 51 f. Trabalho Final de Curso Técnico (Técnico em Manutenção e Suporte em Informática) -- Instituto Federal de Educação, Ciência e Tecnologia de São Paulo, Câmpus Itapetininga, Itapetininga, 2017. Disponível em: <<https://drive.ifsp.edu.br/s/CoQpCEY3zC00jw6>>. Acessado em: dez. 2020.

- CAMPOS, J. C., RIBEIRO, A. N. UML no desenvolvimento de sistemas interactivos. In: Actas da 2ª Conferência Nacional em Interação Pessoa-Máquina, 2., 2006, Braga, 2006, p. 77-80. Disponível em: <<https://haslab.uminho.pt/jccampos/publications/uml-no-desenvolvimento-o-de-sistemas-interactivos>>. Acessado em: dez. 2020.
- CHEN, H.; MOHAPATRA, P. Using service brokers for accessing backend servers for web applications. In: Proceeding of 23rd International Conference on Distributed Computing Systems Workshops, 23., 2003. DOI: <<https://doi.org/10.1109/ICDCSW.2003.1203670>>.
- DATE, C. J. Introdução a Sistemas de Bancos de Dados. 8. ed. Rio de Janeiro: Elsevier, 2003.
- GOOGLE. Material Design. Google. Disponível em: <<https://material.io/design>>. Acessado em: dez. 2020.
- IBGE – Instituto Brasileiro de Geografia e Estatística. Desemprego | IBGE, IBGE, Set. 2020. Disponível em: <<https://www.ibge.gov.br/explica/desemprego.php>>. Acessado em: mar. 2020.
- IBGE – Instituto Brasileiro de Geografia e Estatística, Desemprego cai para 11,9% na média de 2019; informalidade é a maior em 4 anos | IBGE, IBGE, Jan. 2020. Disponível em: <<https://agenciadenoticias.ibge.gov.br/agencia-noticias/2012-agencia-de-noticias/noticias/26741-desemprego-cai-para-11-9-na-media-de-2019-informalidade-e-a-maior-em-4-anos>>. Acessado em: mar. 2020.
- EXPRESS. Express.js. OpenJs Foundation. Fev/2021. Disponível em <<https://expressjs.com/>>. Acessado em: fev. 2021.
- HAHN, E. M. Express in Action Writing, building, and testing Node.js applications. Manning Publications Co., 2016. Disponível em: <https://paranbits.com/home/zak/dbox/copyfiles/2019-01-04/258_Node.js-Express-in-Action.pdf> Acessado em: dez. 2020.
- JADHAV, M. A.; SAWANT, B. R.; DESHMUKH, A. Single Page Application using AngularJS, International Journal of Computer Science and Information Technologies (IJCSIT), Vol. 6, 2015. Disponível em: <<https://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.736.4771&rep=rep1&type=pdf>>. Acessado em: dez. 2020.
- LABS, M. Mobills. Mobills Labs Soluções em Tecnologia LTDA, Nov. 2020. Disponível em: <<https://www.mobills.com.br/>>. Acessado em: abr. 2020.
- LENON. Node.js – O que é, como funciona e quais as vantagens, Opus Software, Set. 2018. Disponível em: <<https://www.opus-software.com.br/node-js/>>. Acessado em: dez. 2020.
- GUIABOLSO F. C. B. S. LTDA. Guiabolso. Guiabolso Finanças Correspondente Bancário e Serviços LTDA. Nov. 2020. Disponível em: <<https://www.guiabolso.com.br/>>. Acessado em: abr. 2020.
- MALDONADO, J. C.; BRAGA, R. T. V.; GERMANO, F. S. R.; MASIERO, P. C. (ed). Padrões e frameworks de software. Notas Didáticas, Instituto de Ciências Matemáticas e de Computação da Universidade de São Paulo, ICMC/USP, São Paulo, SP, Brasil, 2002. Disponível em: <<https://sites.icmc.usp.br/rtvb/apostila.pdf>>. Acessado em: jul. 2020.
- MELO, A. C. Desenvolvendo aplicações com UML 2.2 do conceito à implementação. 3º ed. Rio de Janeiro; Brasport, 2010.
- NODE. Node.js. OpenJS Foundation. Disponível em: <<https://nodejs.org/en/>> Acessado em: dez. 2020.
- ORGANIZZE, Organizze. Organizze, Nov/2020. Disponível em: <<https://www.organizze.com.br/>>. Acessado em: dez. 2020.

- PAGANI, T. Quando utilizar (ou não) frameworks CSS. 2011. Disponível em: <<https://tableless.com.br/quando-utilizar-ou-nao-frameworks-css/>>. Acessado em: dez. 2020.
- POSTGRESQL. PostgreSQL. The PostgreSQL Global Development Group. Disponível em: <<https://www.postgresql.org/>>. Acessado em: jan. 2021.
- ROBBINS, J. N. Learning web design: A beginner's guide to HTML, CSS, JavaScript, and web graphics. Cambridge, UK: O'Reilly Media, 2012.
- RAUCH, G. Socket.io. Automattic. Disponível em: <<https://socket.io/>>. Acessado em: ago. 2021.
- SAVOIA, J. R. F.; SAITO, A. T.; SANTANA F. de A. Paradigmas da educação financeira no Brasil. **Revista de Administração Pública**, vol. 41, n. 6, Scielo Brazil: Nov/dez. 2007, p. 1121-1141. Disponível em: <<https://www.scielo.br/pdf/rap/v41n6/06.pdf>>. Acessado em: abr. 2020.
- SIEBRA, S. A. Banco de dados. Vol. 2, Universidade Federal Rural de Pernambuco (UFRPE), 2010. Disponível em: <<https://pt.slideshare.net/CLEAN13/livro-banco-dedadosvolume02>>. Acessado em: ago. 2020.

Documento Digitalizado Público

Anexo I - artigo - TCC

Assunto: Anexo I - artigo - TCC
Assinado por: Andre Constantino
Tipo do Documento: Relatório Externo
Situação: Finalizado
Nível de Acesso: Público
Tipo do Conferência: Documento Digital

Documento assinado eletronicamente por:

- **Andre Constantino da Silva, PROFESSOR ENS BASICO TECN TECNOLOGICO**, em 21/03/2022 13:36:30.

Este documento foi armazenado no SUAP em 21/03/2022. Para comprovar sua integridade, faça a leitura do QRCode ao lado ou acesse <https://suap.ifsp.edu.br/verificar-documento-externo/> e forneça os dados abaixo:

Código Verificador: 921430

Código de Autenticação: 5040c2a0ce

