

GUILHERME CARVALHO  
YAGO VALDEROZA  
PAULO FAGUNDES

# **GERENCIADOR DE FUNDOS DE INVESTIMENTOS**

São Paulo  
2021, v-1.0.0



GUILHERME CARVALHO  
YAGO VALDEROZA  
PAULO FAGUNDES

## **GERENCIADOR DE FUNDOS DE INVESTIMENTOS**

Monografia apresentada ao Centro Universitário Carlos Drummond Andrade para obtenção da nota da disciplina "Projeto de Programação" no 1º semestre de 2021.

Centro Universitário Carlos Drummond de Andrade  
Faculdade de Análise e Desenvolvimento de Sistemas  
Programa de Graduação

Orientador: Eduardo Palhares Júnior

São Paulo  
2021, v-1.0.0

GUILHERME CARVALHO

YAGO VALDEROZA

PAULO FAGUNDES

GERENCIADOR DE FUNDOS DE INVESTIMENTOS/ GUILHERME CARVALHO

YAGO VALDEROZA

PAULO FAGUNDES. – São Paulo, 2021, v-1.0.0-

33p. : il. (algumas color.) ; 30 cm.

Orientador: Eduardo Palhares Júnior

Projeto Integrador – Centro Universitário Carlos Drummond de Andrade

Faculdade de Análise e Desenvolvimento de Sistemas

Programa de Graduação, 2021, v-1.0.0.

1. Palavra-chave1. 2. Palavra-chave2. 2. Palavra-chave3. I. Eduardo Palhares Júnior. II. Centro Universitário Carlos Drummond de Andrade. III. Curso de Análise e Desenvolvimento de Sistemas. IV. Gerenciador de Fundo de Investimentos.

GUILHERME CARVALHO  
YAGO VALDEROZA  
PAULO FAGUNDES

## GERENCIADOR DE FUNDOS DE INVESTIMENTOS

Monografia apresentada ao Centro Universitário Carlos Drummond Andrade para obtenção da nota da disciplina "Projeto de Programação" no 1º semestre de 2021.

Trabalho aprovado. São Paulo, 24 de junho de 2021:

---

**Eduardo Palhares Júnior**  
Orientador

---

**Ricardo de Andrade**  
Convidado 1

São Paulo  
2021, v-1.0.0



# Resumo

Desenvolvimento de um aplicativo web destinado a gerenciar aplicações financeiras em fundos de investimentos. Com o crescimento no setor financeiro agregando valor ao mercado de investimentos, é importante criar uma facilidade para o investidor aplicar dinheiro nos fundos de seu interesse. Nosso projeto do gerenciador de fundos foca em organizar e exibir um resumo dos diversos patrimônios do investidor em cada fundo e a rentabilidade dos montantes investidos em cada fundo da plataforma. Nesse documento será apresentado o processo passo-a-passo do desenvolvimento do programa, detalhando o que foi feito em cada etapa da construção das estruturas do programa.



# Abstract

Development of a web app destined to manage finance applications in investment funds. With the growth of the finance system adding value to the investment market, it's important to create a facility for the investor to apply money in the funds of interest. Our fund manager project focuses on organizing and displaying a resume of the investor's patrimony in each fund and the profitability of the invested amounts in each of the platform's funds. In this document the step-by-step process of the program's development will be presented, detailing what was done in each step of construction of the program's structure.



# Lista de ilustrações

|  |    |
|--|----|
| Figura 1 – Estrutura da tabela no banco de dados . . . . .     | 23 |
| Figura 2 – Fundos de investimentos listados . . . . .          | 24 |
| Figura 3 – Estrutura da tabela fundos . . . . .                | 24 |
| Figura 4 – Estrutura de cálculo fundos . . . . .               | 25 |
| Figura 5 – Classe de requisições . . . . .                     | 25 |
| Figura 6 – Regras e entidades da orientação à objeto . . . . . | 26 |
| Figura 7 – Regras de conversão e armazenamento . . . . .       | 26 |
| Figura 8 – Interação com o banco de dados . . . . .            | 27 |
| Figura 9 – Desenvolvimento dos métodos . . . . .               | 27 |
| Figura 10 – Tratamento de erros . . . . .                      | 28 |
| Figura 11 – Codificação HTML . . . . .                         | 28 |
| Figura 12 – Integração das variáveis . . . . .                 | 29 |
| Figura 13 – Exemplo na interface . . . . .                     | 29 |
| Figura 14 – Estrutura da de cálculo fundos . . . . .           | 30 |



# Lista de quadros



# Lista de tabelas



# Sumário

|            |                                       |           |
|------------|---------------------------------------|-----------|
| <b>1</b>   | <b>INTRODUÇÃO</b>                     | <b>17</b> |
| <b>2</b>   | <b>DESENVOLVIMENTO</b>                | <b>19</b> |
| <b>2.1</b> | <b>Objetivos</b>                      | <b>19</b> |
| 2.1.1      | Objetivo Geral                        | 19        |
| 2.1.2      | Objetivos Específicos                 | 19        |
| <b>2.2</b> | <b>Justificativa</b>                  | <b>19</b> |
| <b>2.3</b> | <b>Fundamentação teórica</b>          | <b>19</b> |
| 2.3.1      | Banco de dados                        | 19        |
| 2.3.2      | Java                                  | 21        |
| 2.3.3      | Angular                               | 21        |
| <b>2.4</b> | <b>Metodologia</b>                    | <b>21</b> |
| 2.4.1      | Limitações de estudo                  | 22        |
| 2.4.2      | Fontes de Pesquisa                    | 22        |
| <b>3</b>   | <b>RESULTADOS</b>                     | <b>23</b> |
| <b>3.1</b> | <b>Protótipo</b>                      | <b>23</b> |
| 3.1.1      | Engenharia reversa do banco de dados  | 23        |
| 3.1.2      | Tabela da aplicação                   | 23        |
| 3.1.3      | Estrutura da tabela de fundos         | 24        |
| 3.1.4      | Estrutura da tabela de Cálculo Fundos | 25        |
| 3.1.5      | Classe java Controller                | 25        |
| 3.1.6      | Classe Java Model                     | 26        |
| 3.1.7      | Classe Java Converter                 | 26        |
| 3.1.8      | Classe Java Repository                | 27        |
| 3.1.9      | Classe Java Service                   | 27        |
| 3.1.10     | Classe de Tratamento de Handler       | 28        |
| 3.1.11     | Classe Html no Angular                | 28        |
| 3.1.12     | Classe Component Angular              | 29        |
| 3.1.13     | Dashboard do aplicativo               | 29        |
| 3.1.14     | Lista de Saldo e Cotas do aplicativo  | 30        |
| <b>4</b>   | <b>CONCLUSÃO</b>                      | <b>31</b> |
|            | <b>REFERÊNCIAS</b>                    | <b>33</b> |



# 1 Introdução

O avanço tecnológico trouxe importantes mudanças no comportamento social e sobretudo o mercado de fundos de investimentos, fundo de investimento é um tipo de aplicação financeira que reúne recursos de um conjunto de investidores (cotistas), com o objetivo de obter lucro com a compra e venda de títulos e valores mobiliários, de cotas de outros fundos ou de bens imobiliários, ações no Brasil e no exterior.

Ao se tratar do mercado de fundos de investimentos, este tornou-se muito mais concorrido. A principal vantagem dos fundos de investimento é possibilitar que diversos investidores concentrem recursos para aumentar seu poder de negociação e diluir os custos de administração.

Escolhemos banco de dados e back-end e front-end para apresentar no desenvolvimento do trabalho, são eles MySQL, Java. e Angular. Faremos um breve relato para demonstrar como estes três recursos funcionam e as vantagens para o gestor que precisa ter em mãos alguns cenários para a rápida tomada de decisão, sabendo que o cliente da atualidade quer respostas imediatas para fundos de negócios, e potenciais compras. Falaremos também sobre a limitação do nosso trabalho que foi a tentativa de criar aplicativo para empresa.



## 2 Desenvolvimento

### 2.1 Objetivos

#### 2.1.1 Objetivo Geral

O objetivo principal do sistema é fornecer aos clientes uma ferramenta simples auxiliar quando investir em fundos financeiros e facilitar o controle e gerenciamento de suas cotas e dinheiro investido, de maneira totalmente digital. O investidor pode ver um panorama geral do seu patrimônio e as suas aplicações organizadas por categorias.

#### 2.1.2 Objetivos Específicos

1. Panorama geral do patrimônio
2. Aplicações organizadas por categoria

### 2.2 Justificativa

Com a recente tendência mundial de pesquisar por empresas na internet, todos os investidores são obrigados a se atualizar sobre os negócios adotando uma nova estratégia que envolve a utilização de sites para alcançar todas as possíveis metas nos seus investimentos.

Atualmente o objetivo é que seja totalmente de fácil acesso e digital ao investidor aplicar o dinheiro em sites com informações das ações da empresas e informações sobre administração do fundo, contendo informações das ações que contém no fundo, a ideia é migrar o brasileiro do investimento da poupança e começar a investir em outras áreas e alcançar sua liberdade financeira ou até mesmo criar sua própria aposentadoria, queremos ajudar o brasileiro a chegar nesse objetivo através dos nosso site.

### 2.3 Fundamentação teórica

#### 2.3.1 Banco de dados

Banco de dados é um sistema de arquivos relacionados que armazena dados sobre pessoas, lugares ou coisas. O banco de dados organiza dados relacionados e os exibe de forma que gerem sentido, transformando os dados em informações úteis ao usuário. Já há muitos anos vêm se utilizando SGBDs para gerenciar e manipular os bancos de dados da

empresa. Os dados de um banco de dados podem ser acessados pelo usuário através de um programa de computador ou um aplicativo de celular associado a ele.

Segundo (SILBERSCHATZ et al., 2006), um banco de dados "é uma coleção de dados inter-relacionados, representando informações sobre um domínio específico", um armazenamento que contém os dados.

SGBDs são Sistemas de Gerenciamento de Banco de Dados. Sua função é fornecer uma interface para que os usuários possam consultar e manipular os dados armazenados sem que a aplicação cliente em sua máquina tenha que gerenciar o acesso de usuários e processar os comandos e informações. Tais sistemas podem ser definidos em três partes básicas: Linguagem de definição de dados, linguagem de manipulação de dados, e dicionário de dados.

A Linguagem de definição de dados é um conjunto de comandos usados para especificar os conteúdos que serão aceitos no banco de dados através da estrutura de base com tabelas, relacionamentos e os elementos dos dados que serão inseridos.

A Linguagem de Manipulação envolve os comandos usados para incluir, remover, recuperar e modificar as informações dos bancos de dados de acordo com as alterações dos registros.

Dicionário de dados é um tipo de legenda que descreve o conteúdo, classificação, tamanho, etc. dos campos de dados dos diagramas entidade-relacionamento do banco de dados.

Durante a criação de um banco de dados, utiliza-se da modelagem de dados para definir qual será a estrutura de organização do banco de dados, como os dados se relacionam, e serão atribuídas identificações às entidades. A modelagem de dados trabalha com três níveis: o modelo conceitual, modelo lógico, e o modelo físico.

O Modelo conceitual é a estrutura base da organização dos dados. Nesse nível são definidos de forma abstrata quais dados serão armazenados, como serão organizados e como estarão relacionados no nível Entidade-Relacionamento.

O modelo lógico estará definindo a estrutura já contando com as limitações do sistema, irá implementar os atributos das entidades, tais como nome, as chaves primárias e estrangeiras, tipo de dados de cada campo, seu tamanho, etc.

O modelo físico é a sequência de comandos executados na interface do banco de dados para criar as tabelas, estruturas e ligações das entidades então criando o banco de dados conforme visualizado no modelo lógico.

As entidades são objetos (pessoas, lugares ou coisas) das quais terão seus dados armazenados e organizados no banco de dados, e se relacionam ou não com outras entidades através do conceito entidade-relacionamento. No conceito de entidade-relacionamento,

temos a cardinalidade dos relacionamentos, que é o número de ocorrências de uma entidade que pode estar associada a uma outra entidade através do relacionamento em mínimo e máximo.

Antigamente todo dado era armazenado em fichas de papel, pastas, eram feitos arquivos para organizar fisicamente o conteúdo. Este modo era muito trabalhoso principalmente para grandes bases. Com o avanço da tecnologia e o desenvolvimento dos microcomputadores, o modo de armazenar os dados também foi aprimorado.

### 2.3.2 Java

Todos os comandos do programa, cálculos, processamento e organização de dados (back-end do aplicativo) necessários para que o programa funcione e execute suas funções designadas foram desenvolvidos na linguagem Java ensinada nas aulas do professor Oswaldo.

### 2.3.3 Angular

O Site de investimentos será desenvolvido com o front-end em Angular. O AngularJS é um framework open-source de desenvolvimento front-end que possibilita o desenvolvimento de aplicações web, com foco em simplificar tanto a codificação quanto o processo de teste. Além disso, é possível integrá-lo com bibliotecas famosas como o Bootstrap e o PrimeNG que foi a biblioteca usada neste projeto, ajudando a acelerar esse tipo de codificação.

O AngularJS também permite aos desenvolvedores web fazer uso da linguagem de marcação HTML para definir associações de dados, validações, além de response handlers para lidar com as ações do usuário em um formato declarativo que também contribui para essa mesma aceleração. Com tudo isso, a maior consequência é, de longe, o crescimento dos aplicativos, cada vez mais ricos em funcionalidades e recursos.

## 2.4 Metodologia

Os procedimentos metodológicos serão baseados nos livros indicados pelos professores e através das aulas, além das notas do conteúdo adquirido durante as aulas expositivas.

Os livros serão a grande fonte de base teórica para todos os assuntos abordados no decorrer do nosso projeto.

A internet nos trará conhecimento mais recentes sobre o tema ora abordado e insight pois apresenta trabalhos sobre as tendências neste segmento.

O conteúdo de aula, tanto material escrito como as vídeo-aulas ministradas pelos professores serão de grande ajuda por trazer demonstrações práticas de como usar o conteúdo estudado para solucionar problemas do dia-a-dia do funcionário de TI. Dessa forma ficará mais visível o trabalho na totalidade.

### 2.4.1 Limitações de estudo

Apenas alguns dos integrantes do grupo conhecem alguma linguagem de programação mais a fundo, e não estudamos integração de software a fundo, portanto as nossas intenções foram apenas produzir um site simples para os clientes. A alta complexidade das regras de negócio e com um curto período de tempo para o desenvolvimento do projeto foram limitações para o grupo.

### 2.4.2 Fontes de Pesquisa

- Livros
- Google Acadêmico
- Diário de estudo

## 3 Resultados

Ao não aplicarmos o sistema no consultório não podemos obter resultados positivos nem negativos, o que se pode dizer é que os objetivos que queríamos atingir também seriam algo que obviamente seria benéfico para o negócio do consultório, levando em conta a organização, a facilidade de acessar os dados e até mesmo o nível e proteção que comparado a um caderno obviamente é algo superior.

### 3.1 Protótipo

#### 3.1.1 Engenharia reversa do banco de dados

Estrutura da tabela no banco de dados, como neste projeto estamos atuando com apenas uma tabela a engenharia reversa ficou bem enxuta.

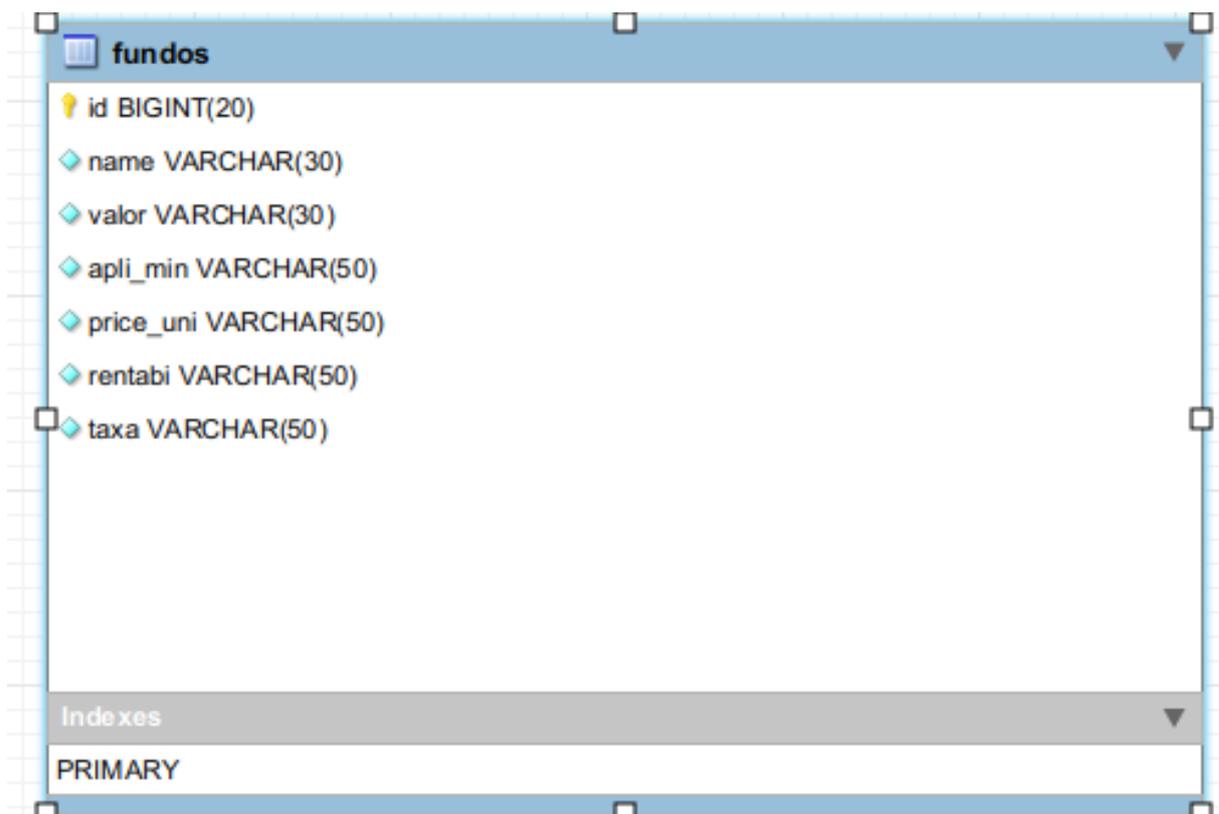


Figura 1 – Estrutura da tabela no banco de dados

#### 3.1.2 Tabela da aplicação

Exemplo de dados na tabela do banco de dados contendo as informações dos fundos de investimentos listados no site.

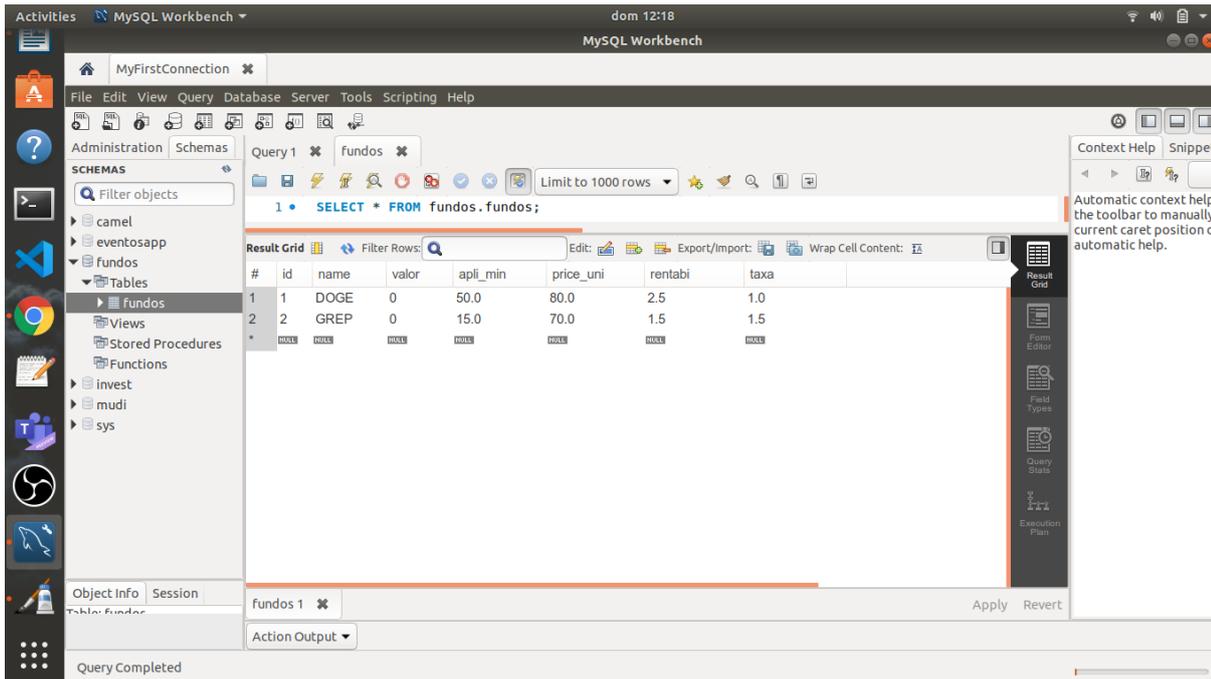


Figura 2 – Fundos de investimentos listados

### 3.1.3 Estrutura da tabela de fundos

Nesta imagem demonstramos a estrutura da tabela fundos no banco de dados.

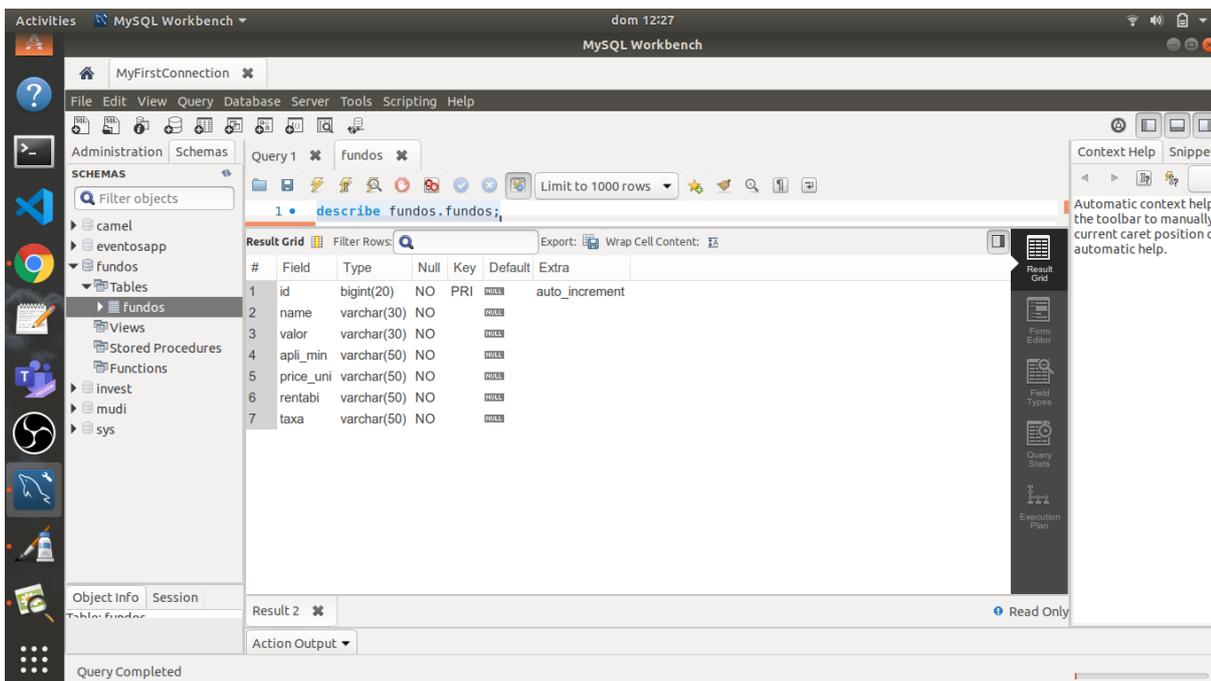


Figura 3 – Estrutura da tabela fundos

### 3.1.4 Estrutura da tabela de Cálculo Fundos

Nesta imagem demonstramos a estrutura da tabela cálculo fundos no banco de dados.

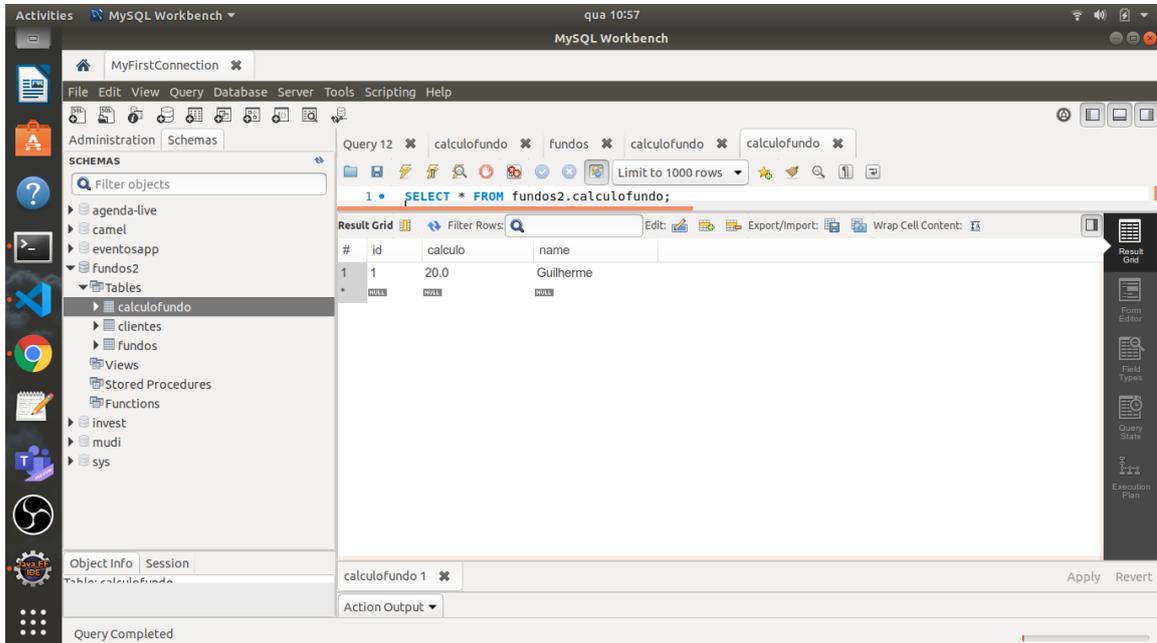


Figura 4 – Estrutura de cálculo fundos

### 3.1.5 Classe java Controller

Essa classe recebe as requisições do programa.

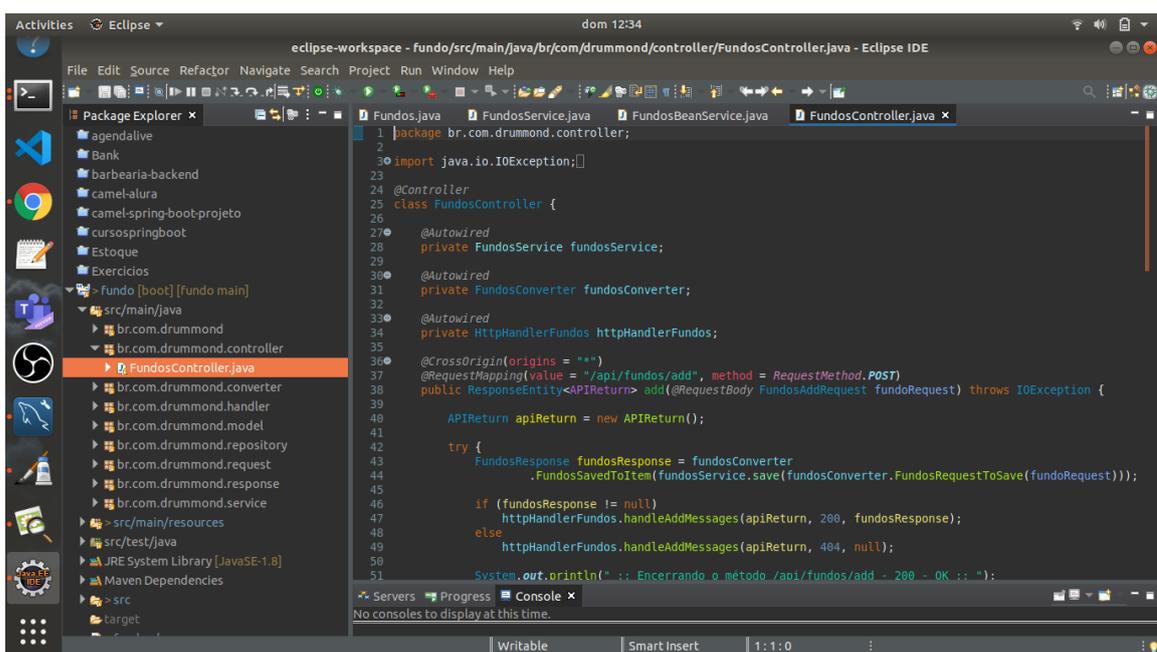


Figura 5 – Classe de requisições

### 3.1.6 Classe Java Model

Aqui se encontram as regras de negócio e entidades do negócio usando Orientação a Objeto, conhecimento adquirido nas aulas de Java da universidade.

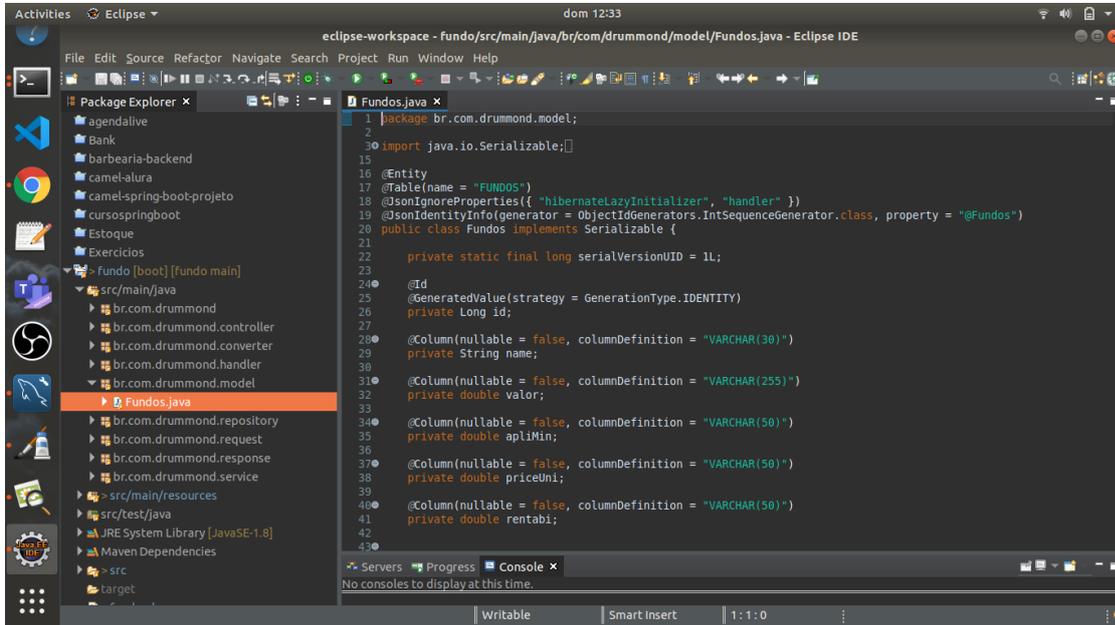


Figura 6 – Regras e entidades da orientação à objeto

### 3.1.7 Classe Java Converter

Essa classe contém os métodos para converter o valor(string) recebido do front-end e com isso podemos salvar as informações no banco de dados. Essa classe contém métodos para salvar as informações no banco de dados e salvar em uma lista os dados.

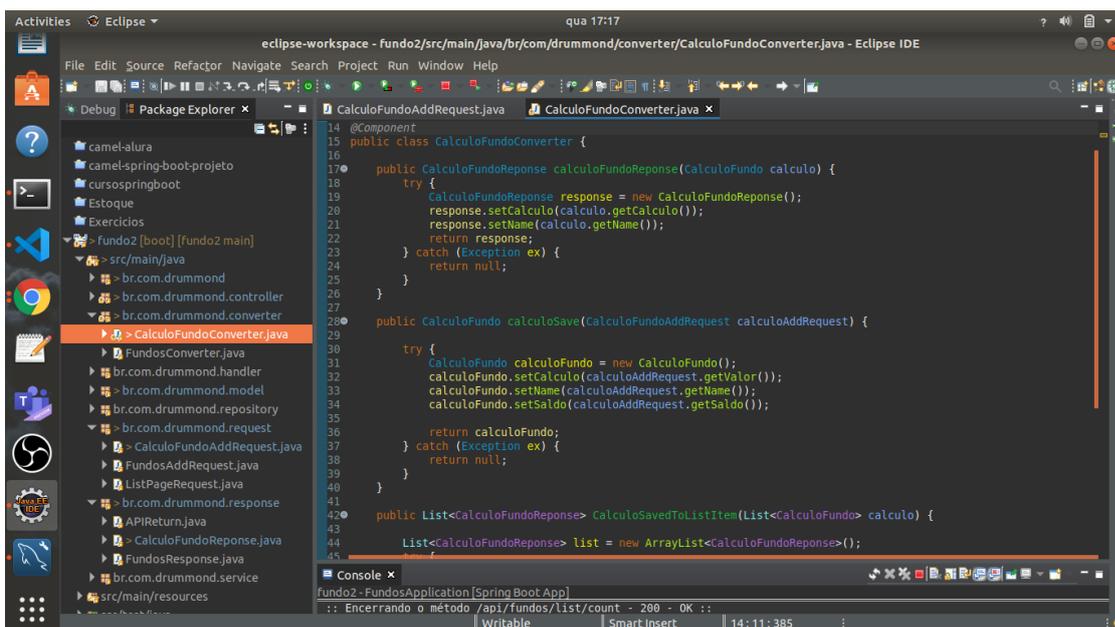


Figura 7 – Regras de conversão e armazenamento

### 3.1.8 Classe Java Repository

Classe que faz consultas, alterações e salva os dados no banco de dados através de Querys.

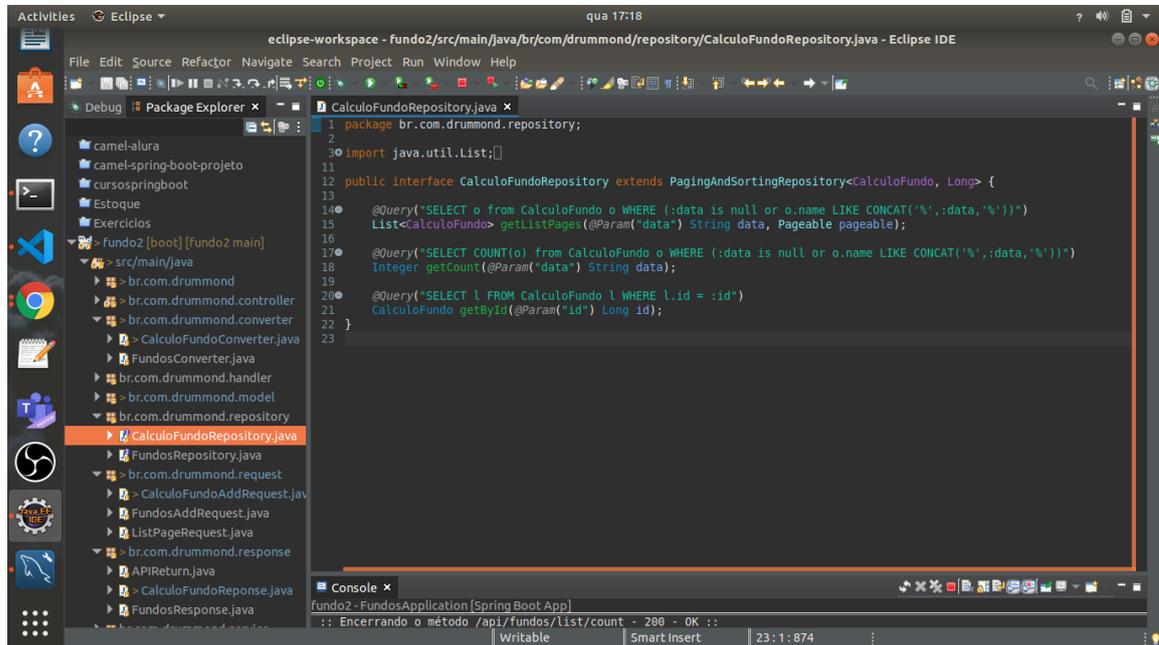


Figura 8 – Interação com o banco de dados

### 3.1.9 Classe Java Service

Observa-se o desenvolvimento dos métodos usados na aplicação.

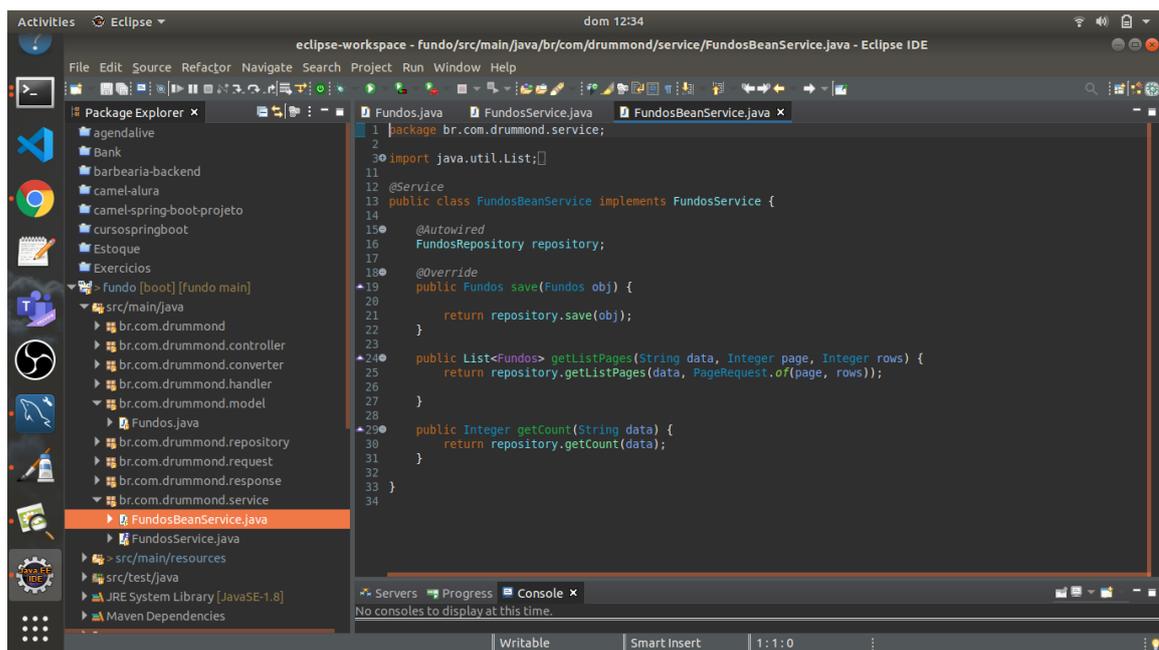


Figura 9 – Desenvolvimento dos métodos

### 3.1.10 Classe de Tratamento de Handler

Classe que se aplica ao tratamento de erros da aplicação.

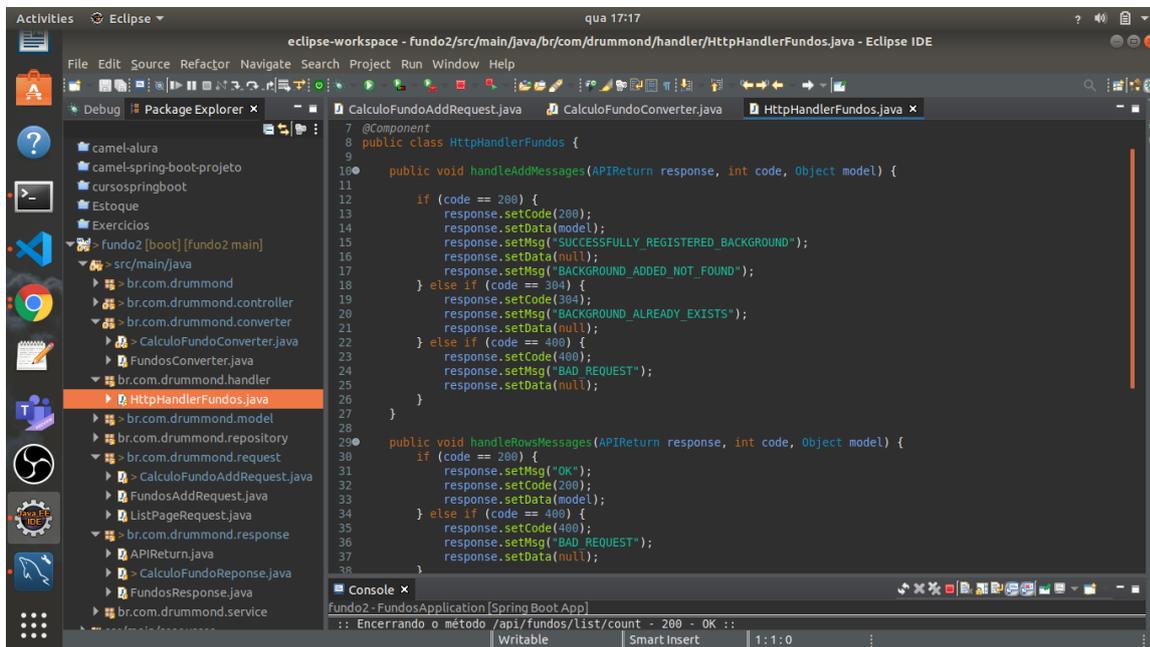


Figura 10 – Tratamento de erros

### 3.1.11 Classe Html no Angular

Classe de codificação Html na tecnologia Angular, está classe é feita a parte de escrita no frontend.

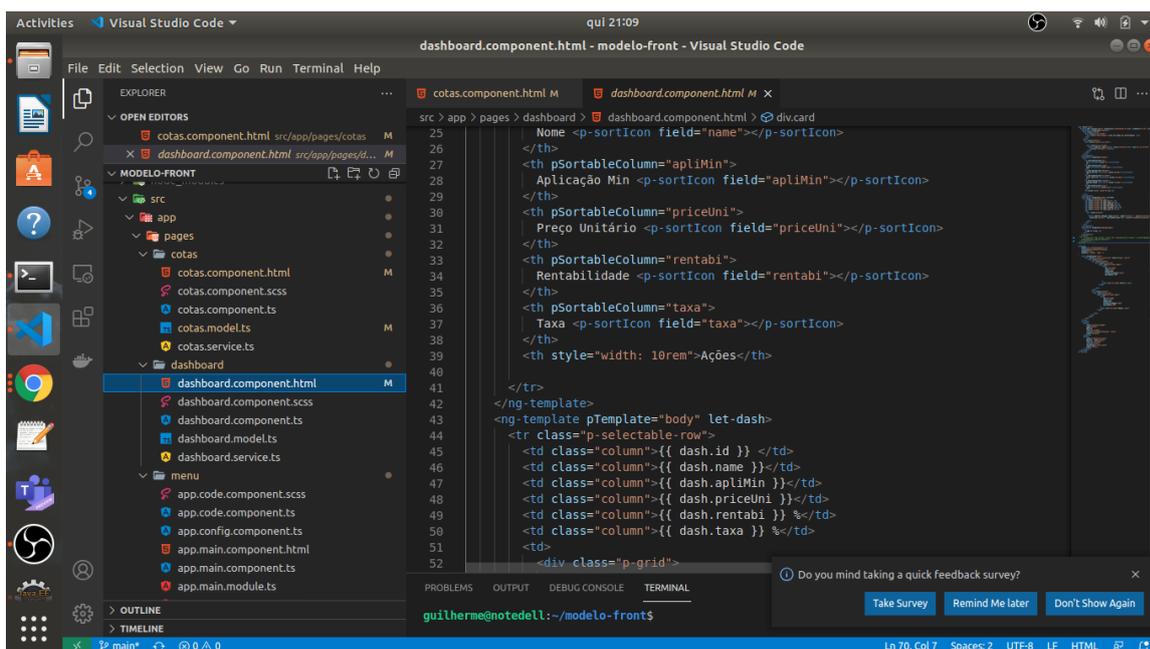
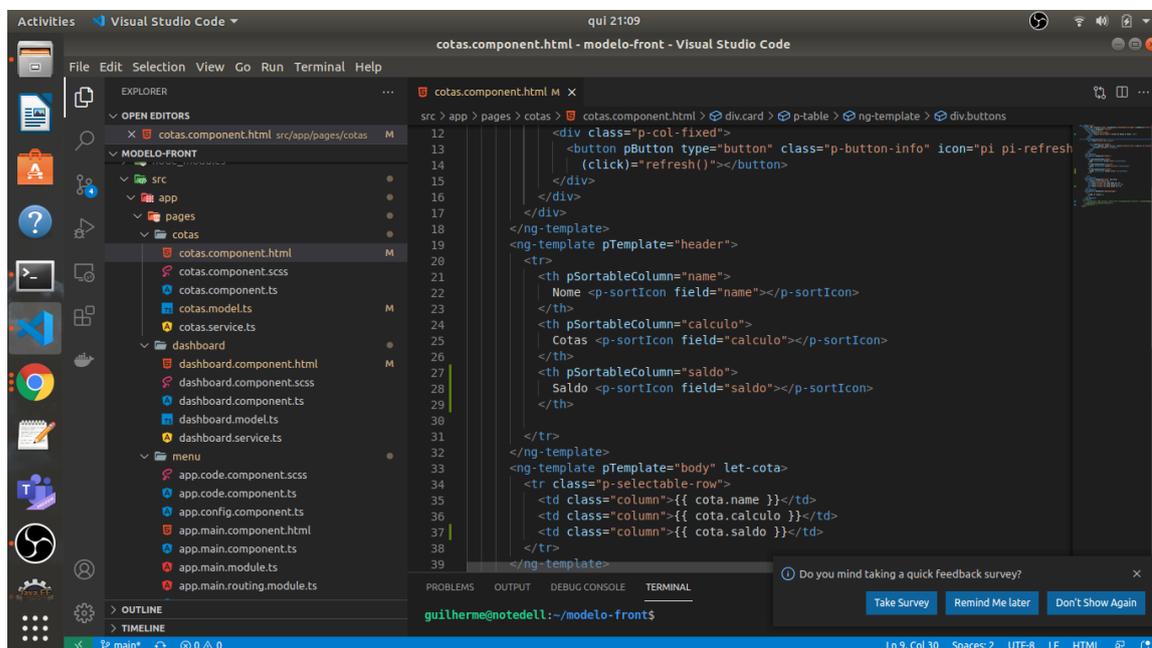


Figura 11 – Codificação HTML

### 3.1.12 Classe Component Angular

Nessa classe é feito a integração das variáveis do backend para serem expostas no frontend com os resultados.



```

src > app > pages > cotas > cotas.component.html > div.card > p-table > ng-template > div.buttons
12 <div class="p-col-fixed">
13 <button pButton type="button" class="p-button-info" icon="pi pi-refresh
14 (click)="refresh()"></button>
15 </div>
16 </ng-template>
17 </div>
18 <ng-template pTemplate="header">
19 <tr>
20 <th pSortableColumn="name">
21 Nome <p-sortIcon field="name"></p-sortIcon>
22 </th>
23 <th pSortableColumn="calculo">
24 Cotas <p-sortIcon field="calculo"></p-sortIcon>
25 </th>
26 <th pSortableColumn="saldo">
27 Saldo <p-sortIcon field="saldo"></p-sortIcon>
28 </th>
29 </tr>
30 </ng-template>
31 <ng-template pTemplate="body" let-cota>
32 <tr class="p-selectable-row">
33 <td class="column">{{ cota.name }}</td>
34 <td class="column">{{ cota.calculo }}</td>
35 <td class="column">{{ cota.saldo }}</td>
36 </tr>
37 </ng-template>
38 </div>
39 </div>

```

Figura 12 – Integração das variáveis

### 3.1.13 Dashboard do aplicativo

Nesta imagem pode-se observar um exemplo da interface de listas de fundos do aplicativo em uso no navegador Google Chrome.

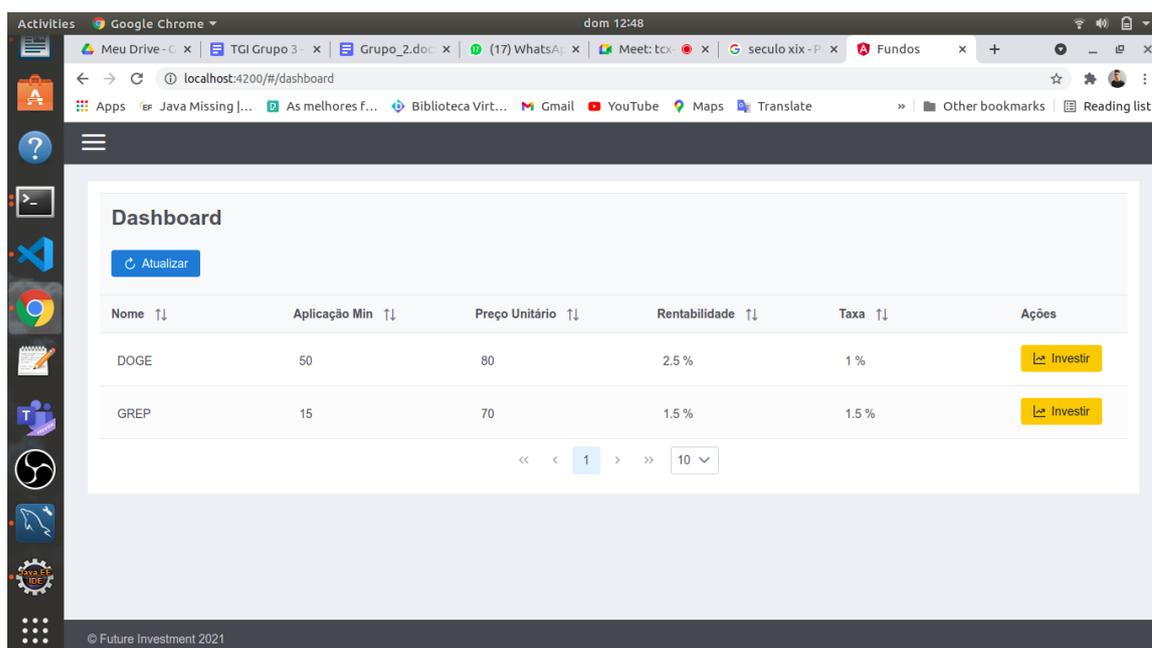


Figura 13 – Exemplo na interface

### 3.1.14 Lista de Saldo e Cotas do aplicativo

Nesta imagem pode-se observar um exemplo da interface de lista de saldo e cotas compradas pelo usuário do aplicativo em uso no navegador Google Chrome.

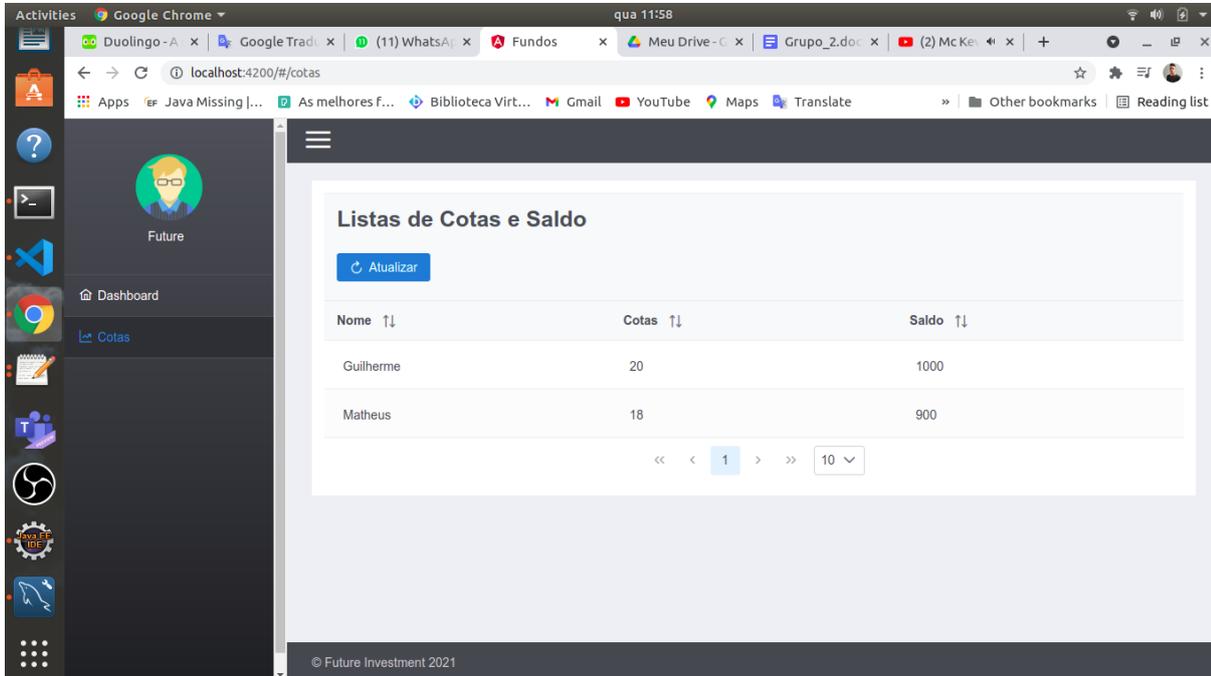


Figura 14 – Estrutura da de cálculo fundos

## 4 Conclusão

Em suma, este trabalho demonstra a necessidade de um aplicativo gerenciador de investimentos para que um investidor tenha maior controle sobre suas aplicações, lucros, taxas e impostos. Tal aplicativo substitui a necessidade de controlar tais investimentos manualmente com arquivos físicos ou mesmo aplicativos menos eficientes que exigem a entrada de dados diretamente do usuário, tal como o Excel, entre outros.

A facilidade proporcionada por esse aplicativo traz grande economia de tempo aos investidores que costumam consultar seus ganhos com frequência. Também facilitará o trabalho da contabilidade, que terá fácil acesso aos dados necessários aos cálculos de renda para então calcular os impostos devidos do investidor.



## Referências

SILBERSCHATZ, A. et al. *Sistema de banco de dados*. Elsevier, 2006. ISBN 9788535211078. Disponível em: <[https://books.google.com.br/books?id=Bn7\\\_GAAACAAJ](https://books.google.com.br/books?id=Bn7\_GAAACAAJ)>. Citado na página 20.